Communication within Clouds: Open Standards and Proprietary Protocols for Data Center Networking

Carolyn J. Sher DeCusatis and Aparico Carranza, New York City College of Technology
Casimer M. DeCusatis, IBM Corporation

ABSTRACT

Cloud computing and other highly virtualized data center applications have placed many new and unique requirements on the data center network infrastructure. Conventional network protocols and architectures such as Spanning Tree Protocol and multichassis link aggregation can limit the scale, latency, throughput, and virtual machine mobility for large cloud networks. This has led to a multitude of new networking protocols and architectures. We present a tutorial on some of the key requirements for cloud computing networks and the various approaches that have been proposed to implement them. These include industry standards (e.g., TRILL, SPB, software-defined networking, and OpenFlow), best practices for standards-based data center networking (e.g., the open datacenter interoperable network), as well as vendor proprietary approaches (e.g., FabricPath, VCS, and Qfabric).

INTRODUCTION

Cloud computing is a method of delivering computing services from a large, highly virtualized data center to many independent end users, using shared applications and pooled resources. While there are many different definitions for cloud computing [1], it is typically distinguished by the following attributes: on-demand self-service, broad network access, resource pooling, rapid and elastic resource provisioning, and metered service at various quality levels.

Implementation of these attributes as part of a large enterprise-class cloud computing service that provides continuous availability to a large number of users typically requires significantly more server, networking, and storage resources than conventional data centers (up to an order of magnitude more in many cases). This is only achievable through extensive use of virtualization. While server virtualization has existed since the 1960s, when it was first implemented on IBM mainframes, it has only become widely available on affordable commodity x86 servers within the last decade or so. In recent years, many equipment vendors have contributed to the hardware and software infrastructure, which has made enterprise-class virtualization widely available. This, in turn, enables new designs for cloud computing, including hosting multiple independent tenants on a shared infrastructure, rapid and dynamic provisioning of new features, and implementing advanced load balancing, security, and business continuity functions (including multisite transaction mirroring). This has brought about profound changes in many aspects of data center design, including new requirements for the data center network.

Modern cloud data centers employ resource pooling to make more efficient use of data center appliances and to enable dynamic reprovisioning in response to changing application needs. Examples of this include elastic workloads where application components are added, removed, or resized based on the traffic load; mobile applications relocating to different hosts based on distance from the host or hardware availability; and proactive disaster recovery solutions, which relocate applications in response to a planned site shutdown or a natural disaster. It has been shown [2] that highly virtualized servers place unique requirements on the data center network. Cloud data center networks must contend with huge numbers of attached devices (both physical and virtual), large numbers of isolated independent subnetworks, multitenancy (application components belonging to different tenants are collocated on a single host), and automated creation, deletion, and migration of virtual machines (facilitated by large layer 2 network domains). Furthermore, many cloud data centers now contain clusters or pods of servers, storage, and networking, configured so that the vast majority of traffic (80–90 percent in some cases) flows between adjacent servers within a pod (so-called east-west traffic). This is a very different traffic pattern from conventional data center networks, which supported higher levels of traffic between server racks or pods (so-called north-south traffic).

0163-6804/12/$25.00 © 2012 IEEE

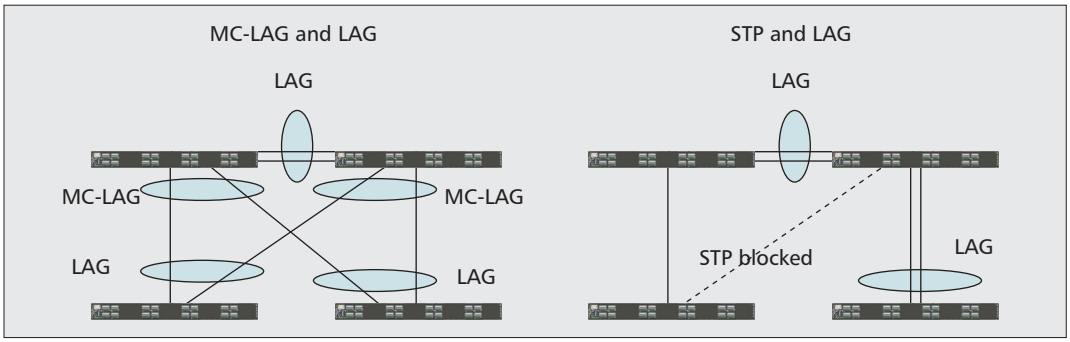

Figure 1. *MC-LAG configuration without STP (left) and with STP (right).*

To cope with these problems, many attempts have been made to develop best practices for networking design and management. Several new industry standards and proprietary network architectures have recently been proposed. Many network designers, users, and administrators have long expressed a desire for standardization and multivendor interoperability in the data center, to simplify management, improve reliability, and avoid being locked into one particular vendor's proprietary product offerings and pricing structure. These conclusions were supported by a recent analyst report [3], which determined that multisourcing of network equipment is not only practical, but can reduce total cost of ownership by 15–25 percent. Furthermore, a recent survey of 468 business technology professionals on their data networking purchasing preferences [4] showed that adherence to industry standards was their second highest requirement, behind virtualization support. Standardization also encourages future-proofing of the network and helps promote buying confidence. Despite these advantages, many large network equipment providers have advocated for proprietary network protocols. A recent study [5] showed that five out of the six largest network equipment manufacturers include proprietary features in their products, and only three of the six claimed interoperability with other vendors' access layer switches.

In this article, we present a tutorial on cloud networking design practices, including both industry standard and vendor proprietary alternatives. It should be noted that although modern data centers will almost certainly require some version of these new protocols, many of these approaches are far less mature than conventional network designs. Early adopters should use caution when evaluating the best choices for their data center needs.

Spanning Tree Protocol and Multichassis Link Aggregation

Spanning Tree Protocol (STP) is a layer 2 switching protocol used by classical Ethernet that ensures loop-free network topologies by always creating a single path tree structure through the network. In the event of a link failure or reconfiguration, the network halts all traffic while the spanning tree algorithm recalculates the allowed loop-free paths through the network. (STP creates a loop-free topology using Multi Chassis EtherChannel [MCEC], also referred to as Virtual Port Channels [vPC] for Cisco switches.) The changing requirements of cloud data center networks are forcing designers to reexamine the role of STP. One of the drawbacks of a spanning tree protocol is that in blocking redundant ports and paths, a spanning tree reduces the aggregate available network bandwidth significantly. Additionally, STP can result in circuitous and suboptimal communication paths through the network, adding latency and degrading application performance. A spanning tree cannot easily be segregated into smaller domains to provide better scalability, fault isolation, or multitenancy. Finally, the time taken to recompute the spanning tree and propagate the changes in the event of a failure can vary widely, and sometimes becomes quite large (seconds to minutes). This is highly disruptive for elastic applications and virtual machine migrations, and can lead to cascaded system-level failures.

To help overcome the limitations of STP, several enhancements have been standardized. These include Multiple STP (MSTP), which configures a separate spanning tree for each virtual local area network (VLAN) group and blocks all but one of the possible alternate paths within each spanning tree. Also, the link aggregation group (LAG) standard (IEEE 802.3ad) allows two or more physical links to be bonded into a single logical link, either between two switches or between a server and a switch. Since a LAG introduces a loop in the network, STP has to be disabled on network ports using LAGs. It is possible for one end of the link aggregated port group to be dual-homed into two different devices to provide device-level redundancy. The other end of the group is still single-homed and continues to run normal LAG. This extension to the LAG specification is called multichassis link aggregation (MC-LAG), and is standardized as IEEE 802.1ax (2008). As shown in Fig. 1, MC-LAG can be used to create a loop-free topology without relying on STP; because STP views the LAG as a single link, it will not exclude redundant links within the LAG. For example, it is possible for a pair of network interface cards (NICs) to be dual-homed into a pair of access switches (using NIC teaming), and then use MC-LAG to interconnect the access switches with a pair of core switches.

Most MC-LAG systems allow dual homing across only two paths; in practice, MC-LAG systems are limited to dual core switches because it is extremely difficult to maintain a coherent

To help overcome the limitations of STP, several enhancements have been standardized. These include Multiple STP, which configures a separate spanning tree for each virtual local area network group and blocks all but one of the possible alternate paths within each spanning tree.

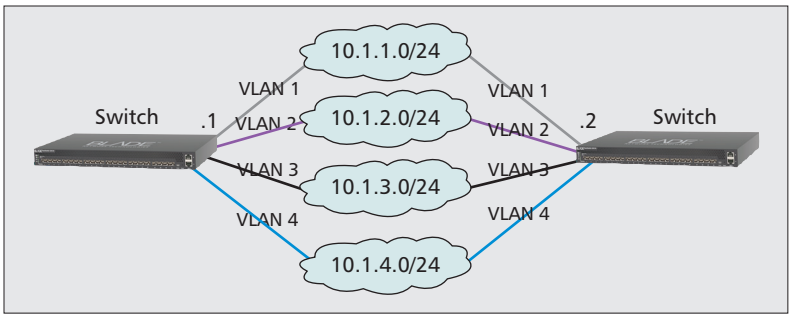

Figure 2. *Example of a four-way layer 3 ECMP design.*

state between more than two devices with submicrosecond refresh times. Unfortunately, the hashing algorithms that are associated with MC-LAG are not standardized; care needs to be taken to ensure that the two switches on the same tier of the network are from the same vendor (switches from different vendors can be used on different tiers of the network). As a relatively mature standard, MC-LAG has been deployed extensively, does not require new forms of data encapsulation, and works with existing network management systems and multicast protocols.

Layer 3 vs. Layer 2 Designs for Cloud Computing

Many conventional data center networks are based on well established, proven approaches such as a layer 3 "fat tree" design (or Clos network) using equal cost multipathing (ECMP). These approaches can be adapted to cloud computing environments. As shown in Fig. 2, a layer 3 ECMP design creates multiple load balanced paths between nodes. The number of paths is variable, and bandwidth can be adjusted by adding or removing paths up to the maximum allowed number of links. Unlike a layer 2 STP network, no links are blocked with this approach. Broadcast loops are avoided by using different VLANs, and the network can route around link failures. Typically, all attached servers are dual-homed (each server has two connections to the first network switch using active-active NIC teaming). This approach is known as a *spine and leaf* architecture, where the switches closest to the server are leaf switches that interconnect with a set of spine switches. Using a two-tier design with a reasonably sized (48-port) leaf and spine switch and relatively low oversubscription (3:1) as illustrated in Fig. 3, it is possible to scale this network up to around 1000–2000 or more physical ports.

If devices attached to the network support Link Aggregation Control Protocol (LACP), it becomes possible to logically aggregate multiple connections to the same device under a common virtual link aggregation group (VLAG). It is also possible to use VLAG interswitch links (ISLs) combined with Virtual Router Redundancy Protocol (VRRP) to interconnect switches at the same tier of the network. VRRP supports IP forwarding between subnets, and protocols such as Open Shortest Path First (OSPF) or Border Gateway Protocol (BGP) can be used to route around link failures. Virtual machine migration is limited to servers within a VLAG subnetwork.

Layer 3 ECMP designs offer several advantages. They are based on proven standardized technology that leverages smaller, less expensive rack or blade chassis switches (virtual soft switches typically do not provide layer 3 functions and would not participate in an ECMP network). The control plane is distributed, and smaller isolated domains may be created.

There are also some trade-offs when using a layer 3 ECMP design. The native layer 2 domains are relatively small, which limits the ability to perform live virtual machine (VM) migrations from any server to any other server. Each individual domain must be managed as a separate entity. Such designs can be fairly complex, requiring expertise in IP routing to set up and manage the network, and presenting complications with multicast domains. Scaling is affected by the control plane, which can become unstable under some conditions (e.g., if all the servers attached to a leaf switch boot up at once, the switch's ability to process Address Resolution Protocol [ARP] and Dynamic Host Configuration Protocol [DHCP] relay requests will be a bottleneck in overall performance). In a layer 3 design, the size of the ARP table supported by the switches can become a limiting factor in scaling the design, even if the medium access control (MAC) address tables are quite large. Finally, complications may result from the use of different hashing algorithms on the spine and leaf switches.

New protocols are being proposed to address the limitations on live VM migration presented by a layer 3 ECMP design, while at the same time overcoming the limitations of layer 2 designs based on STP or MC-LAG. All of these approaches involve some implementation of multipath routing, which allows for a more flexible network topology than STP [5] (Fig. 4). In the following sections, we discuss two recent standards, TRILL and SPB, both of which are essentially layer 2 ECMP designs with multipath routing.

TRILL

Transparent Interconnection of Lots of Links (TRILL) is an Internet Engineering Task Force (IETF) industry standard protocol originally proposed by Radia Perlman, who also invented STP. TRILL runs a link state protocol between devices called *routing bridges* (RBridges). Specifically, TRILL uses a form of layer 2 link state protocol called intermediate system to intermediate system (IS-IS) to identify the shortest paths between switches on a hop-by-hop basis, and load balance across those paths. In other words, connectivity information is broadcast across the network so that each RBridge knows about all the other RBridges and the connections between them. This gives RBridges enough information to compute pair-wise optimal paths for unicast traffic. For multicast/broadcast groups or delivery to unknown destinations, TRILL uses distribution trees and an Rbridge as the root for forwarding. Each node of the network recalculates the TRILL header and performs other

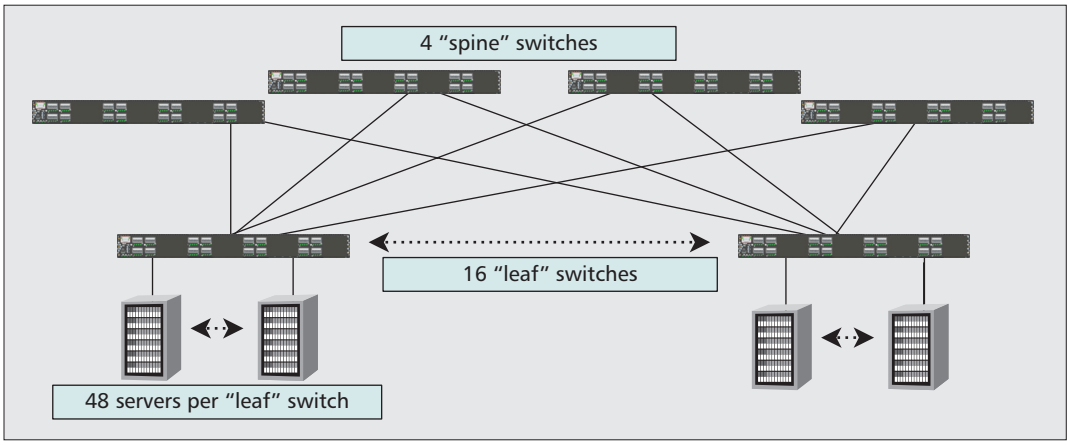

Figure 3. *Example of an L3 ECMP leaf-spine design.*

functions such as MAC address swapping. STP is not required in a TRILL network.

There are many potential benefits of TRILL, including enhanced scalability. TRILL allows the construction of loop-free multipath topologies without the complexity of MCEC, which reduces the need for synchronization between switches and eliminates possible failure conditions that would result from this complexity. TRILL should also help alleviate issues associated with excessively large MAC address tables (approaching 20,000 entries) that must be discovered and updated in conventional Ethernet networks. Furthermore, the protocol can be extended by defining new TLV (type-length-value) data elements for carrying TRILL information (some network equipment vendors are expected to implement proprietary TLV extensions in addition to industry standard TRILL.)

For example, consider the topologies shown in Figs. 5a and 5b. In a classic STP network with bridged domains, a single multicast tree is available. However, in a TRILL fabric, there are several possible multicast trees based on IS-IS. This allows multidestination frames to be efficiently distributed over the entire network. There are also several possible active paths in a TRILL fabric, which makes more efficient use of bandwidth than does STP.

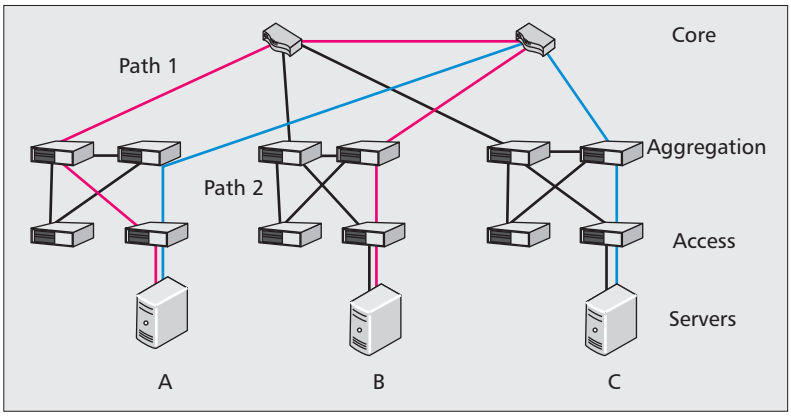

Figure 4. *Example of multipath routing between three servers A, B, and C.*

Shortest Path Bridging

Shortest path bridging (SPB) is a layer 2 standard (IEEE 802.1aq) addressing the same basic problems as TRILL, although using a slightly different approach. SPB was originally introduced to the IEEE as provider link state bridging (PLSB), a technology developed for the telecommunications carrier market, which was itself an evolution of the IEEE 802.1ah standard (provider backbone bridging or PBB). The SPB standard reuses the PBB 802.1ah data path, and therefore fully supports the IEEE 802.1ag-based operations, administration, and management (OA&M) functionality. The 802.1ah frame format provides a service identifier (I-SID) that is completely separate from the backbone MAC addresses and the VLAN IDs. This enables simplified data center virtualization by separating the connectivity services layer from the physical network infrastructure. The I-SID abstracts the service from the network. By mapping a VLAN or multiple VLANs to an I-SID at the service access point, SPB automatically builds a shortest path through the network. The I-SID also provides a mechanism for granular traffic control by mapping services (applications) into specific I-SIDs.

When a new device is attached to the SPB network and wishes to establish communication with an existing device, there is an exchange (enabled by the IS-IS protocol) to identify the requesting device and learn its immediate neighboring nodes. Learning is restricted to the edge of the network, which is reachable by the I-SID. The shortest bidirectional paths from the requesting device to the destination are then computed using link metrics such as ECMP. The same approach is used for both unicast and broadcast/multicast packets, which differs from TRILL. Once the entire multicast tree has been developed, the tree is then pruned, and traffic is assigned to a preferred path. The endpoints thus learn how to reach one another by transmitting on a specific output address port, and this path remains in place until there is a configuration change in the network. In this manner, the endpoints are fully aware of the entire traffic path, which was not the case for TRILL. The route packets take through the network can be determined from the source address, destination

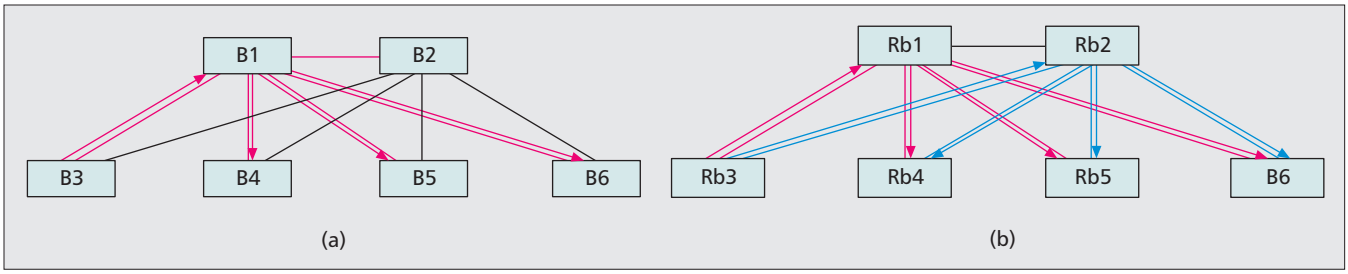

Figure 5. *a) STP multicast with bridged domains; b) TRILL multicast domain with Rbridges.*

address, and VLAN ID. Traffic experiences the same latency in both directions on the resulting paths through the network, also known as congruent pathing.

Within SPB there are two models for multipath bridging: shortest path bridging VLAN (SPBV) and SPB Mac-in-Mac (SPBM). Both variants use IS-IS as the link state topology protocol, and both compute shortest path trees between nodes. SPBV uses a shortest path VLAN ID (SPVID) to designate nodal reachability. SPBM uses a backbone MAC (BMAC) and backbone VLAN ID (BVID) combination to designate nodal reachability. Both SPBV and SPBM provide interoperability with STP. For data center applications, SPBM is the preferred technology. There are several other proposed enhancements and vendor proprietary extensions to SPB that we do not discuss here; interested readers should refer to the latest working drafts of proposed SPB features and extensions.

TRILL or SPB may be used in the following cases:

1 When the number of access port needs exceeds the MC-LAG capacity (typically a few thousand ports) and the network needs an additional core switch that does not participate in the MC-LAG
2 When implementing a multivendor network with one vendor's switches at the access layer and another vendor's switch in the core
3 When implementing different switch product lines from a single vendor, and one product cannot participate in the other product's fabric

Since both TRILL and SPB were developed to address the same underlying problems, comparisons between the two are inevitable. The main difference between the two approaches is that TRILL attempts to optimize the route between hops in the network, while SPB makes the entire path known to the edge nodes. TRILL uses a new form of MAC-in-MAC encapsulation and OA&M, while SPB has variants for both existing MAC-in-MAC as well as queue-in-queue encapsulation, each with their associated OA&M features. Only SPB currently supports standard 802.1 OA&M interfaces. TRILL uses different paths for unicast and multicast traffic than those used by SPB, which likely will not make a difference for the vast majority of IP traffic applications. There are also variations in the loop prevention mechanisms, number of supported virtualization instances, lookup and forwarding, handling of multicast traffic, and other features between these two standard protocols. The real impact of these differences on the end user or network administrator remains to be seen.

As of this writing, several companies have announced their intention to support SPB (including Avaya and Alcatel-Lucent) or to support TRILL (including IBM, Huawei, and Extreme Networks). Other companies have announced proprietary protocols which essentially address the same basic problems but claim to offer other advantages. The following sections briefly discuss three alternatives from some of the largest networking companies (Cisco, Brocade, and Juniper Networks); other proprietary approaches exist, but we do not review them here.

FabricPath

Cisco produces switches that use either a version of TRILL or a proprietary multipath layer 2 encapsulation protocol called FabricPath, which is based on TRILL but has several important differences. For example, FabricPath frames do not include TRILL's next-hop header [6], and FabricPath has a different MAC learning technique than TRILL, which according to Cisco may be more efficient. Conversational learning has each FabricPath switch learn the MAC address it needs based on conversations, rather than learning all MAC addresses in the domain. According to Cisco, this may have advantages in cases where a large number of VMs creates far more MAC addresses than a switch could normally handle. FabricPath also supports multiple topologies based on VLANs, and allows for the creation of separate layer 2 domains. FabricPath uses vPC+, Cisco's proprietary multichassis link aggregation protocol, to connect to non-FabricPath switches. According to Cisco, FabricPath has the ability to scale beyond the limits of TRILL or SPB in the network core, supporting up to eight core switches on the same tier in a single layer 2 domain. Edge devices such as the Cisco Nexus 5000 series support up to 32,000 MAC addresses; core devices such as the Cisco Nexus 7000 series support up to 16,000 MAC addresses and 128,000 IP addresses. This approach can likely scale to a few thousand ports or more with reasonable levels of oversubscription (around 4:1). Theoretically, this approach may scale to even larger networks, although in practice scale may be limited by other factors, such as the acceptable size of a single failure domain within the network. Scale can also be extended with the addition of a third

or fourth tier in the network switch hierarchy if additional latency is not a concern. In June 2012 Cisco announced a collection of networking products and solutions collectively known as Cisco ONE, which includes FabricPath and other features. According to Cisco, this approach is different from the standard definition of software-defined networking, and may include both proprietary and industry standard protocols.

Virtual Cluster Switching

Brocade produces switches using a proprietary multipath layer 2 encapsulation protocol called Virtual Cluster Switching (VCS). This approach is based on TRILL (e.g., the data plane forwarding is compliant with TRILL and uses the TRILL frame format [7], and edge ports in VCS support standard LAG and LACP), but it is not necessarily compatible with other TRILL implementations because it does not use an IS-IS core. Brocade's core uses Fabric Shortest Path First (FSPF), the standard path selection protocol in Fibre Channel storage area networks. VCS also uses a proprietary method to discover compatible neighboring switches and connect to them appropriately. According to Brocade, up to 32,000 MAC addresses are synchronized across a VCS fabric. The currently available switches from Brocade can likely scale to around 600 physical ports in a single fabric, although in practice the upper limit may be reduced (e.g., if some ports are configured for Fibre Channel connectivity or interswitch links).

QFabric

Juniper Networks produces switches using a proprietary multipath layer 2/3 encapsulation protocol called QFabric [8]. According to Juniper, Qfabric allows multiple distributed physical devices in the network to share a common control plane and a separate common management plane, thereby behaving as if they were a single large switch entity. For this reason, Qfabric devices are not referred to as edge or core switches, and the overall approach is called a fabric rather than a network. According to Juniper, Qfabric provides equal latency between any two ports on the fabric, up to the current theoretical maximum supported scale of 6144 physical ports with 3:1 oversubscription at the edge of the fabric (the maximum scale fabric requires 128 edge devices and 4 core interconnects). As of March 2012, the largest publicly released independent Qfabric testing involves 1536 ports of 10G Ethernet, or approximately 25 percent of Qfabric's theoretical maximum capacity [8]. According to Juniper, Qfabric provides up to 96,000 MAC addresses and up to 24,000 IP addresses. In May 2012 Juniper announced the QFX3000-M product line (commonly known as micro-fabric), which enables a version of Qfabric optimized for smaller-scale networks. Qfabric is managed by the Junos operating system, and requires a separate fabric controller and an outband management network (often provided by traditional Ethernet switches, e.g., the Juniper EX4200).

Software-Defined Networking and OpenFlow: Emerging Next Generation Protocols

Some of the network traffic routing concerns discussed previously may also be addressed by a new industry standard approach known as software-defined networking (SDN), which advocates virtualizing the data center network with a software overlay and network controller that allows attached servers to control features such as packet flows, topology changes, and network management. There have been several different industry standard software overlay proposals, including the IETF proposed standard distributed overlay virtual Ethernet (DOVE) [2]. SDN is used to simplify network control and management, automate network virtualization services, and provide a platform from which to build agile network services. SDN leverages both industry standard network virtualization overlays and the emerging OpenFlow industry standard protocol. The OpenFlow standard moves the network control plane into software running on an attached server or network controller. The flow of network traffic can then be controlled dynamically, without the need to rewire the data center network. Some of the benefits of this approach include better scalability, larger layer 2 domains and virtual devices, and faster network convergence. These technologies can form the basis for networking as a service (NaaS) in modern cloud data centers.

The OpenFlow specification has been under development as an academic initiative for several years [9]. In early 2011, it was formalized under the control of a non-profit industry consortium called the Open Networking Foundation (ONF). The ONF is led by a board of directors consisting of some of the largest network operators in the world, including Google, Facebook, Microsoft, Verizon, NTT, Deutsche Telekom, and Yahoo. It is noteworthy that all of these companies are end users of networking technology, rather than network equipment manufacturers; this indicates a strong market interest in OpenFlow technology. With this motivation, over 70 equipment manufacturers have joined the ONF (including Cisco, Brocade, Juniper Networks, IBM, and others), which released the latest version of its specification, OpenFlow 1.3, in June 2012 [9]. Since SDN and OpenFlow are still evolving, care must be taken to implement only the approved industry standard version of these protocols to maximize interoperability in a multivendor network and fully realize the benefits intended by the ONF.

OpenFlow takes advantage of the fact that most modern Ethernet switches and routers contain flow tables, which run at line rate and are used to implement functions such as quality of service (QoS), security firewalls, and statistical analysis of data streams. OpenFlow standardizes a common set of functions that operate on these flows and will be extended in the future as the standard evolves. An OpenFlow switch consists of three parts: flow tables in the switch, a remote controller on a server, and a secure communication channel between them. The OpenFlow pro-

The OpenFlow standard moves the network control plane into software running on an attached server or network controller. The flow of network traffic can then be controlled dynamically, without the need to rewire the data center network.

In cloud computing environments, OpenFlow improves scalability and enables multi-tenancy and resources pooling, and will likely co-exist with other Layer 2/3 protocols and network overlays for some time.

tocol allows entries in the flow table to be defined by an external server. For example, a flow could be a TCP connection, all the packets from a particular MAC or IP address, or all packets with the same VLAN tag. Each flow table entry has a specific action associated with a particular flow, such as forwarding the flow to a given switch port (at line rate), encapsulating and forwarding the flow to a controller for processing, or dropping a flow's packets (for example, to help prevent denial of service attacks).

There are many potential applications for OpenFlow in modern networks. For example, a network administrator could create on-demand 'express lanes' for voice and data traffic. Software could also be used to combine several links into a larger virtual pipe to temporarily handle bursts of traffic; afterward, the channels would automatically separate again. In cloud computing environments, OpenFlow improves scalability, enables multitenancy and resource pooling, and will likely coexist with other layer 2/3 protocols and network overlays for some time.

The Open Datacenter Interoperable Network

In May 2012, IBM released a series of five technical briefs known as the Open Datacenter Interoperable Network (ODIN) [10], intended to describe best practices for designing a data center network based on open industry standards. It is noteworthy that these documents do not refer to specific products or service offerings from IBM or other companies. According to IBM, specific implementations of ODIN, which may use products from IBM and other companies, will be described in additional technical documentation (e.g., in June 2012 a solution for storage stretch volume clusters using components from IBM, Brocade, Adva, and Ciena was presented at the Storage Edge conference.) The ODIN documents describe the evolution from traditional enterprise data networks into a multivendor environment optimized for cloud computing and other highly virtualized applications. ODIN deals with various industry standards and best practices, including layer 2/3 ECMP spine-leaf designs, TRILL, lossless Ethernet, SDN, OpenFlow, wide area networking, and ultra-low-latency networks. As of this writing, ODIN has been publicly endorsed by eight other companies and one college [10], including Brocade, Juniper, Huawei, NEC, Extreme Networks, BigSwitch, Adva, Ciena, and Marist College.

Conclusions

In this article, we have presented an overview of layer 2 and 3 protocols that may be used to address the unique requirements of a highly virtualized cloud data center. While existing layer 3 "fat tree" networks provide a proven approach to these issues, there are several industry standards that enhance features of a flattened layer 2 network (TRILL, SPB) or have the potential to enhance future systems (SDN and OpenFlow). Many of these standards are described in the ODIN reference architecture, which has been endorsed by numerous networking equipment companies. Some major networking equipment manufacturers have also developed vendor proprietary protocols to address the same issues (FabricPath, VCS, and Qfabric), each with different features for scalability, latency, oversubscription, and management. None of these proposed solutions has yet reached the same level of maturity as STP and MC-LAG, but there are many promising evaluations underway by early adopters. Within the next year or so, compliance testing of network equipment using common use cases for cloud computing should begin to emerge. Unfortunately, many of the recently developed protocols do not interoperate, and this lack of consensus represents a serious concern for the networking industry, coming at a critical inflection point in the development of next-generation cloud-enabled networks. Some designers may claim that radical architectural innovation in the cloud network requires them to outpace industry standards; this approach is only acceptable if the resulting system either opens its architecture to others after a reasonable time period, or agrees to support any developing industry standards as a free, nondisruptive upgrade in addition to any proprietary alternatives. The same approach holds true for pre-standard implementations, which should provide a free upgrade path to the eventual finalized form of the industry standard. Finally, we recognize that the choice of networking equipment involves trade-offs between many factors in addition to those described here, including total cost of ownership, energy consumption, and support for features such as convergence of Fibre Channel over Ethernet (FCoE), IPv6, or remote direct memory access (RDMA) over converged Ethernet (RoCE), as well as layer 4–7 functionality. These topics will be examined in more detail as part of ongoing research in this area.

Acknowledgements

All company names and product names are registered trademarks of their respective companies. The names Cisco, FabricPath, vPC+, ONE, and Nexus are registered trademarks of Cisco Systems Inc. The names Brocade, Virtual Cluster Switching, and VCS are registered trademarks of Brocade Communications Systems Inc. The names Juniper, Qfabric, Junos, QFX3000-M, and EX4200 are registered trademarks of Juniper Networks Inc.

References

[1] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," 07 Oct. 2009: http://csrc.nist.gov/groups/SNS/cloud-computing/, accessed: 29-Jan-2011.

[2] K. Barabesh *et al.*, "A Case for Overlays in DCN Virtualization," *Proc. 2011 IEEE DC CAVES Wksp., Collocated with ITC 22*.

[3] M. Fabbi and D. Curtis, "Debunking the Myth of the Single-Vendor Network," Gartner Research, Gartner Research Note G00208758, Nov. 2010.

[4] A. Wittmann, "IT Pro Ranking: Data Center Networking," *InformationWeek*, Sept. 2010.

[5] C. J. Sher-DeCusatis, *The Effects of Cloud Computing on Vendor's Data Center Network Component Design*, doctoral thesis, Pace University, White Plains, New York, May 2011.

[6] Cisco white paper C45-605626-00, "Cisco FabricPath At-A-Glance," http://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps9402/at_a_glance_c45-605626.pdf, Accessed: 27-Dec-2011.
[7] Brocade Technical Brief GA-TB-372-01,"Brocade VCS Fabric Technical Architecture," http://www.brocade.com/downloads/documents/technical_briefs/vcs-technical-architecture-tb.pdf, Accessed: 27-Dec-2011.
[8] Juniper white paper 2000380-003-EN (Dec. 2011), "Revolutionizing Network Design: Flattening the Data Center Network with the Qfabric Architecture," http://www.juniper.net/us/en/local/pdf/whitepapers/2000380-en.pdf, accessed 27-Dec-2011, see also independent Qfabric test results, available: http://newsroom.juniper.net/press-releases/juniper-networks-qfabric-sets-new-standard-for-net-nyse-jnpr-0859594, accessed 30 June 2012.
[9] N. McKeown *et al.*, "OpenFlow: Enabling Innovation in Campus Networks," *ACM SIGCOMM Comp. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, 2008; see also http://www.openflow.org, accessed 30 June 2012.
[10] C. M. DeCusatis, "Towards an Open Datacenter with an Interoperable Network (ODIN)," vols. 1–5, http://www.03.ibm.com/systems/networking/solutions/odin.html, accessed 30 June 2012 all ODIN endorsements available https://www-304.ibm.com/connections/blogs/DCN, accessed 30 June 2012.

BIOGRAPHIES

CAROLYN J. SHER DECUSATIS (decusatis@optonline.net) is an adjunct assistant professor in the Computer Engineering Technology Department of New York City College of Technology, Brooklyn. She holds a B.A. from Columbia College and an M.A. from Stony Brook University (formerly State University of New York at Stony Brook) in physics, and a Doctor of Professional Studies in computing from Pace University. She is co-author of the handbook *Fiber Optic Essentials*.

APARICO CARRANZA (acarranza@citytech.cuny.edu) is an associate professor and chair of the Computer Engineering Technology Department of New York City College of Technology, Brooklyn. He earned his Associate's degree (summa cum laude) in electronics circuits and systems from Technical Career Institutes, New York, New York; his Bachelor's degree (summa cum laude) and Master's in electrical engineering from City College/University of New York (CC/CUNY); and his Ph.D. degree in electrical engineering from the Graduate School and University Center, CUNY. He also worked for IBM (RS6000 parallel computers; and S/390 mainframe computers) for several years.

CASIMER M. DECUSATIS [F] (decusat@us.ibm.com) is an IBM Distinguished Engineer and CTO for System Networking Strategic Alliances, based in Poughkeepsie, New York. He is an IBM Master Inventor with over 110 patents, editor of the *Handbook of Fiber Optic Data Communication*, and recipient of several industry awards. He holds an M.S. and Ph.D. from Rensselaer Polytechnic Institute, and a B.S. (magna cum laude) in the Engineering Science Honors Program from Pennsylvania State University. He is a Fellow of the OSA and SPIE, a Distinguished Member of Sigma Xi, and a member of the IBM Academy of Technology. His blog on Data Center Networking is https://www-304.ibm.com/connections/blogs/DCN.