THE QFABRIC ARCHITECTURE

Implementing a Flat Data Center Network

Table of Contents

Executive Summary

Any data center built more than a few years ago is facing one or more of the following challenges:

- The legacy multitier switching architecture cannot cope with the need to provide applications and users with predictable latency and uniform bandwidth. This problem is further exacerbated in a virtualized world, where the performance of virtual machines depends on the physical location of the servers on which those virtual machines reside.
- The power consumed by networking gear represents a significant proportion of the overall power consumed in the data center. This is particularly important today, when escalating energy costs are putting additional pressure on budgets.
- Outages related to misconfigurations and the legacy behavior of Spanning Tree Protocol (STP) result in lost revenue and unhappy customers.
- Siloed Layer 2 domains, built by operators to pare down network complexity, have driven OpEx costs higher by creating multiple standalone systems and devices to manage.
- The increasing performance and densities of modern CPUs has led to an increase in network traffic. The network is often not equipped to deal with the large bandwidth demands and increased number of media access control (MAC) and IP addresses on each network port.
- Separate networks for Ethernet data and storage traffic must be maintained, adding to the training and management budget.

Given all of these outstanding issues, data center operators are seeking new ways of networking within the data center. Juniper Networks® QFabric™ technology offers a solution.

Introduction to QFabric Technology

QFabric technology is Juniper Networks' next-generation data center switching architecture, intended to radically transform the economics and performance of networking in the modern data center. Introduced in early 2011, the QFabric family of products is designed to address the emerging requirements of modern data centers ranging in size from small high-performance computing (HPC) clusters, to large-scale virtualized and converged access enterprise data centers, to cloud-based mega data centers.

QFabric technology exploits the homogeneity and regular layout of a rack-based server infrastructure, as well as the structured cabling already installed in most buildings with future-proofed 40 Gbps and 100 Gbps intra-data center connectivity in mind. A single QFabric system converged switch/router can scale to more than 6,000 10GbE ports with end-to-end latency of five microseconds under typical loads. The lossless QFabric architecture, which is suitable for storage traffic, delivers non-blocking any-to-any connectivity, an STP-free environment for L2 deployments, and a scale-out architecture for both L2 and L3 use cases—all managed as a single device. QFabric architecture—the result of nearly three years of extensive R&D investments—is the networking industry's first true network fabric (read the "Defining Characteristics of QFabric" white paper), producing more than 100 different patent filings across multiple areas of innovation.

This whitepaper describes Juniper's motivation for building the QFabric family of products, and it outlines the design principles and technology options that drove specific architectural choices. This paper also discusses the multiple innovations in the data, control, and management planes that make the QFabric architecture a compelling choice for networking in today's data center. Over the next several years, Juniper believes that competitive pressures to lower the cost and improve the quality of networking will require most modern data centers to deploy a fabric-based architecture.

QFabric Technology at a Glance

QFabric technology has the unique ability to support an entire data center—more than 6,000 10GbE ports—with *a single converged Ethernet switch*. The best way to understand the QFabric architecture is to start with the architecture of a standalone modular switch chassis and map each of its components—line cards, switch fabric, and route engines—to their QFabric technology equivalent. Disaggregating a traditional modular switch and distributing its various components while preserving the operational experience of a single switch is the key architectural innovation of the QFabric architecture.

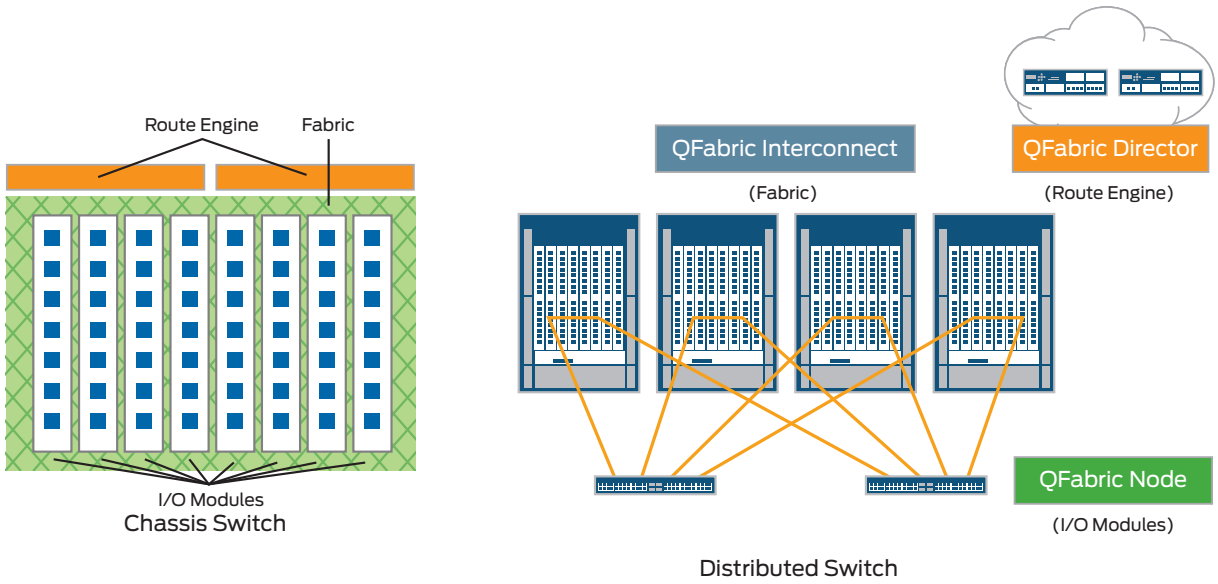


Figure 1: Disaggregating a chassis switch into a distributed switch

The QFabric architecture is composed of three separate components—QFabric Node, QFabric Interconnect, and QFabric Director. QFabric Node—the line card component of a QFabric system—acts as the entry and exit into the fabric. QFabric Interconnect is the high-speed transport device for interconnecting QFabric Nodes. And QFabric Director provides control and management services, delivering a common window for managing all components as a single device.

QFabric Node

QFabric Node is an ultralow latency, high port density, fixed configuration 1 U top-of-rack device that provides access into and out of the network fabric. To draw parallels with a traditional chassis switch, QFabric Node is equivalent to the line card, performing L2/L3 packet forwarding, quality of service (QoS) and access control list management, and other tasks. Packets exiting the uplinks of a QFabric Node enter the QFabric Interconnect, a line rate non-blocking device. All oversubscription is built into the ingress QFabric Node alone, allowing the QFabric Node to provide varying levels of oversubscription all the way down to 1:1.

In addition to its role as the edge of the QFabric architecture, QFabric Node can also serve as a high-performance standalone converged access switch. In fact, it has been shipping in this form as the Juniper Networks QFX3500 Switch since March 2011. A simple software update allows it to move from a standalone switch to a QFabric Node.

QFabric Interconnect

QFabric Interconnect is an ultralow latency, 21 U eight slot chassis with sixteen 40GbE QFabric Node-facing ports per slot. QFabric Nodes exchange data traffic with each other via the QFabric Interconnect by forming a full mesh topology using standard high-speed 40 Gbps optics (100 Gbps in the future). To draw parallels with a traditional chassis switch, the QFabric Interconnect represents the backplane; there are no direct connections between QFabric Interconnects in a QFabric architecture.

While the QFabric Interconnect might look like a standard Ethernet switch on the outside, it is actually quite different. The QFabric Interconnect does not participate in any of the protocols that run within the QFabric architecture; instead, it directs traffic between QFabric Nodes by performing simple tag lookups rather than the traditional and laborious full Ethernet lookup. Simple tag lookups require fewer ASICs and less memory in the Packet Forwarding Engine (PFE) complex, leading to a smaller PFE footprint (less space and greater density) and lower power requirements for running and cooling the hardware, making it more energy efficient.

QFabric Director

QFabric Director is a 2 U server based on the x86 architecture and featuring 36 gigabytes of memory and 4 terabytes

of local disk storage. Multiple QFabric Directors form a compute cluster, which is connected to QFabric Nodes and QFabric Interconnects through a dedicated 1GbE network. To draw parallels with a traditional chassis-based switch, the QFabric Director is equivalent to the supervisor module and Routing Engine.

A Single Logical Switch

With all components working together, the QFabric architecture behaves as a single logical switch that seamlessly integrates into the existing data center infrastructure. Figure 2 shows how QFabric Nodes, QFabric Interconnects, and QFabric Directors are connected to form a QFabric solution. The left side shows the physical connectivity while the right side shows its iconized logical form.

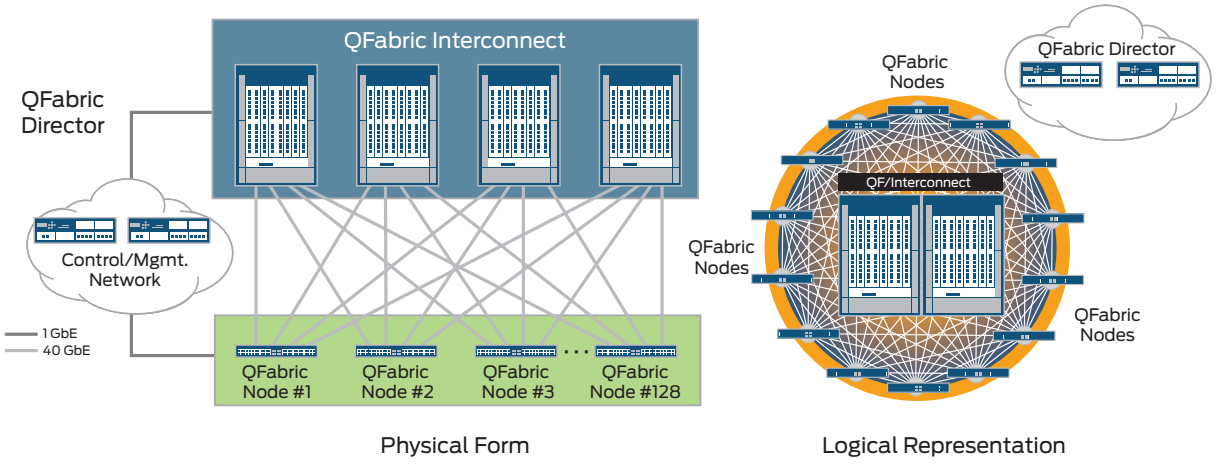


Figure 2: Physical topology and logical icon for QFabric architecture

All interfaces into the QFabric architecture are based on the Ethernet, IP. and Fibre Channel (FC) protocols, which means that they are open and standards-based for seamlessly connecting to storage, servers, routers, firewalls, and other data center devices.

The following diagram shows how the QFabric architecture connects the servers, storage, routers, and service appliances in a data center-wide network topology.

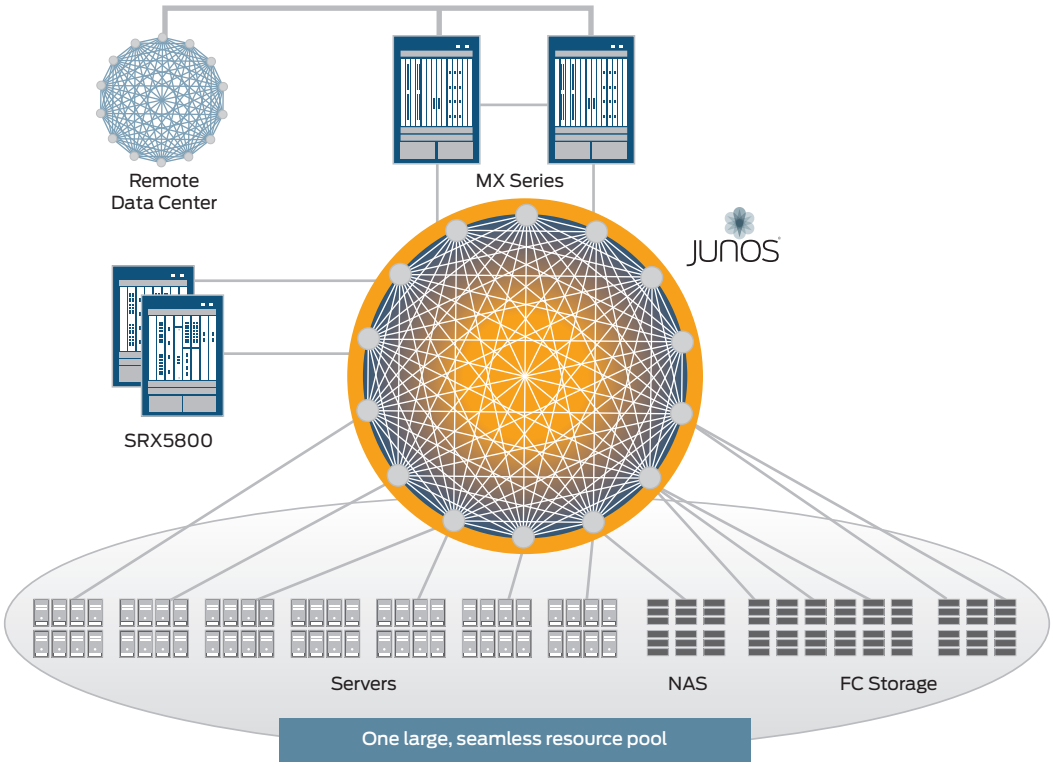


Figure 3: QFabric architecture connecting servers, storage, security appliances, and routers in a data center

QFabric Architecture Benefits

The QFabric architecture offers a wide range of benefits that include:

- A single device abstraction for both L2 and L3 networking to the entire data center fabric, eliminating the management complexity and high OpEx associated with managing multiple network elements individually
- Lower power and space footprints compared to traditional two-tier architectures, saving millions of dollars in operating expenses over the life of the infrastructure
- Constant cross-sectional bandwidth and predictably low latency and jitter between server ports to address any mix of traffic types for superior application performance and user experience
- Eliminates the use of STP for L2 networking within the fabric (STP is well known as the cause of certain failures in several data center operations)
- Seamless mobility of network configuration profiles (i.e., any VLAN on any server port at any time) for greater agility in deploying traditional enterprise IT applications and modern cost saving consolidation practices such as server virtualization
- Optimization of the 5-7 year investments in next-generation structured cabling infrastructure that can support 40 Gbps and 100 Gbps speeds
- Unifying multiple data center networks into a single network infrastructure, reducing the myriad technologies that must be simultaneously maintained
- Scale-out design for a server hall of 100-150 racks, which translates into a linear "build-as-you-grow" cost model that matches the pay-as-you-grow model of cloud computing
- A single, logical switch abstraction that can scale to an extremely large size. (virtualization works best when resource pools are large)
- A rich set of networking constructs such as VLANs, virtual routers, etc. to carve up the physical infrastructure for efficient utilization across multiple users and tenants

QFabric Technology Innovations

QFabric technology introduces a number of innovations to the traditional network architecture, including:

- Smart access, simple transport data plane
- Separation of data and control
- Single switch abstraction
- Distributed control plane

Each of these innovations is described in the sections below.

Smart Access, Simple Transport Data Plane

How traditional data center network systems were built

Traditionally, the blueprint for the data center network has been a multitier switching architecture based on a "simple access, smart aggregation" paradigm. In this model, most of the rich feature set—and cost—is embedded in a pair of modular end-of-row switches that aggregate simple, low featured top-of-rack switches to form the access layer. The low performance oversubscribed aggregation layer, along with poor multipathing and long convergence times of STP, have resulted in small-scale networks. In response, network architects have built out multiple siloed networks, and this has caused a linear increase in access ports accompanied by an exponential increase in cost. Most network operators agree that this model is obsolete.

What other vendors are offering today: "Smart Access, Smart Aggregation"

The emergence of Fibre Channel over Ethernet (FCoE) and server virtualization require the access/edge of the data center network to offer a rich set of features such as QoS, access control lists, large address table sizes, and faster learning rates, driving up the cost of the access layer. At the same time, the need to add more ports in a multitier switching architecture has put additional pressure on the end-of-row modular switches to scale, further increasing the cost burden. While this combination of factors has addressed scale to a limited extent, it has done nothing to solve the problems of escalating network CapEx and OpEx (space, energy, and management). Worse yet, in an effort to contain costs, some products have been used to extend the reach of the aggregation layer by adding additional tiers of simple multiplexor/demultiplexor components without rich QoS or forwarding features, effectively driving the "smart access, smart aggregation" approach back to the "simple access, smart aggregation" model.

The QFabric Technology Approach—Smart Access, Simple Transport

The QFabric architecture can be viewed as a federation of several smart access elements—for example, QFabric Nodes exchanging data traffic over a simple transport interconnect such as the QFabric Interconnect. The QFabric architecture was inspired by the Internet architectures of smart provider edge (PE) routers and simple provider (P) core routers, deployed over multiple years in the harshest environments of large network service provider IP/MPLS networks. The "smart access, simple transport" dictum is eloquently articulated in the near decade old RFC3439 standard titled, "Some Internet Architectural Guidelines and Philosophy," which categorically contends that "end-to-end protocol design should not rely on the maintenance of state inside the network…the complexity of the Internet belongs at the edges…"

This simple transport data plane plays a key role in the high performance, lower cost, and lower power and space benefits of QFabric architecture.

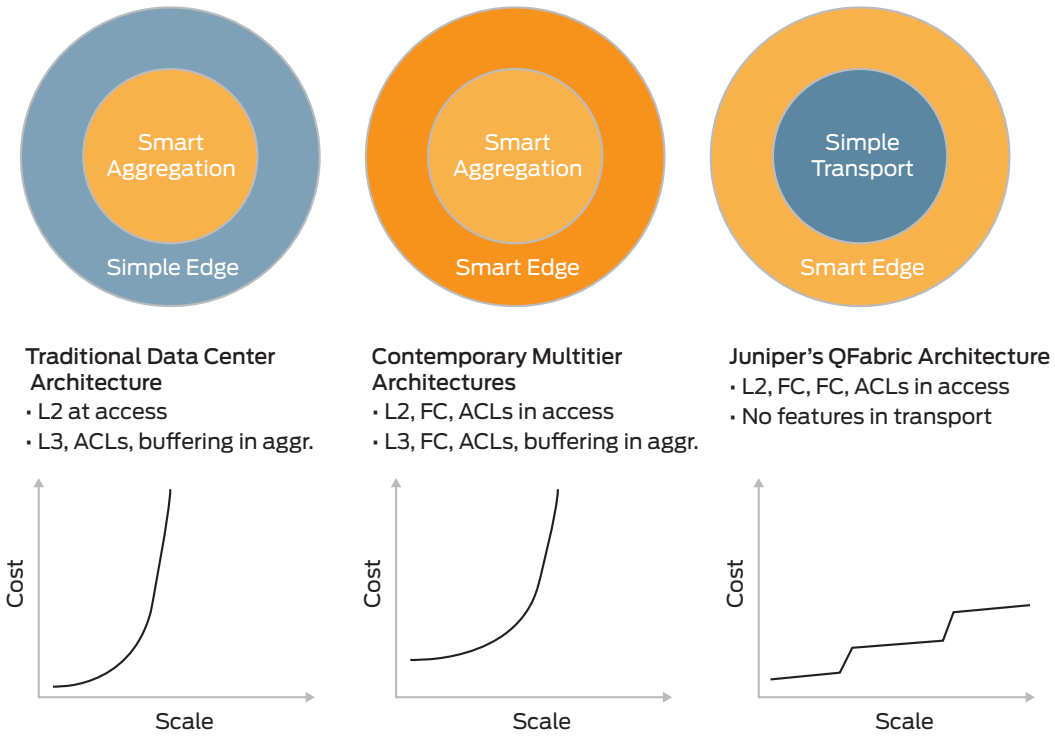


Figure 4: Smart edge, simple transport scales best

Separation of Data and Control

More than 10 years ago, Juniper pioneered the separation of the data and control planes within a single switch/router chassis. However, in a network of individual switches, control messages share the same links as regular data traffic traveling between network elements. Sharing links for both control and data traffic requires careful configuration of QoS parameters to ensure the appropriate isolation of the right traffic types. This puts an extra burden on operators to correctly specify the behavior for packet classification, buffer management, and scheduler priorities and weights. The QFabric architecture extends the separation of data and control traffic by using physically separate networks across the entire data center.

The QFabric architecture has taken the disaggregation of a chassis-based switch's basic components to its proper logical conclusion. With the QFabric architecture, the control network that connects supervisor modules to CPUs on line cards, and which typically lies hidden within the confines of a switch's sheet metal, is now externalized to form a single physical control plane network.

To make sure that this approach does not create management complexity for the operator, QFabric architecture comes fully configured from the factory. Interfaces are enabled with the proper authentication to prevent malicious or inadvertent tampering. In addition, the entire control plane Ethernet network has full redundancy built in to eliminate any single point of failure.

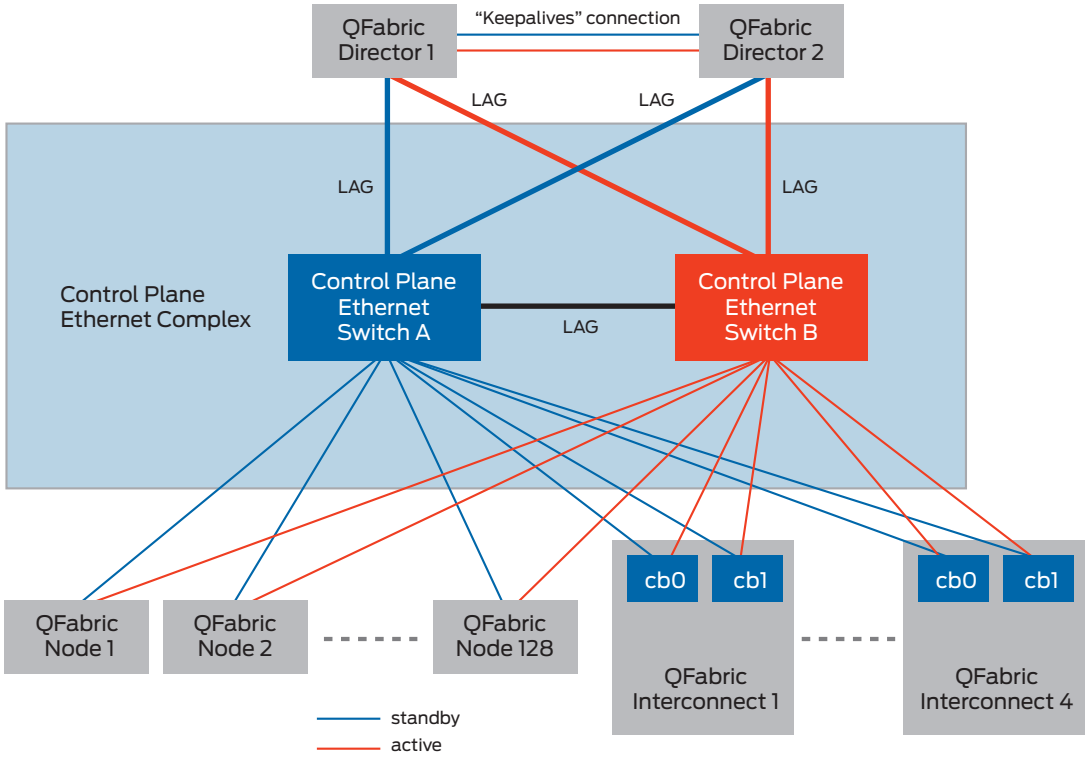


Figure 5: A highly available and resilient control plan Ethernet network

Single Switch Abstraction

The QFabric architecture addresses the network management challenge at its root. Currently, it is extremely hard for element managers to understand the full semantics of an interconnected network. In an attempt to abstract the network and hide its inherent complexity, these managers often reduce the amount of useful information available to the operator. The QFabric architecture avoids these pitfalls by presenting a single switch view of the entire fabric.

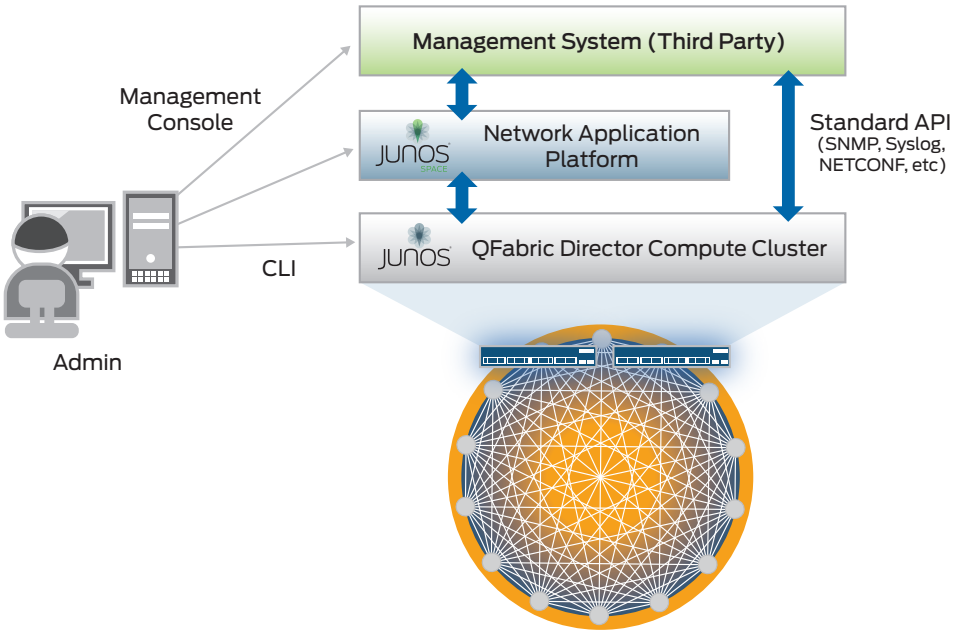


Figure 6: Managing a QFabric architecture

Distributed Control Plane

The QFabric architecture subscribes to the "centralize what you can, distribute what you must" philosophy by implementing a distributed control plane in order to build scale-out networks. In designing QFabric technology, architects distributed whatever had to be distributed while attempting to centralize as much as possible. During the design phase, every aspect of the system was carefully analyzed with this design philosophy in mind.

For instance, QFabric technology had a design goal to preserve the traditional user experience of interacting with a single Ethernet switch. This required that all management and configuration functionality be centralized. It's important to note that "centralized" does not imply a single point of failure; in a QFabric architecture, all centralized functions are deployed in high availability (HA) pairs.

Conversely, while modern CPUs are highly advanced, a single CPU still cannot perform the Routing Engine function on server-facing protocols for thousands of 10GbE endpoints. Therefore, the load required to run these protocols has to be distributed, precluding the single CPU model of traditional switch/routers.

Similar to other distributed systems, the QFabric architecture uses transparent internal processes to share state across all autonomous members of the distributed system. By virtue of this architecture, QFabric technology naturally inherits all useful properties characteristic of a distributed system, including the ability to operate in a *scale-out* fashion. In contrast to traditional monolithic switch/routers, a QFabric architecture grows not by replacing existing components in a scale-up fashion or through a wholesale replication of building blocks that are then interconnected, but by preserving the fabric as is, incrementally adding ports to support additional racks. This method ensures that the marginal cost of the entire switch becomes the cost of connecting one additional rack, which is the definition of a scale-out system.

With QFabric architecture, this loosely coupled federation of autonomous components interacting as a single system using transparent transaction protocols also provides superior resiliency for the overall system. Physical or logical failures in one component are isolated from the rest of the system and do not trigger failures in the rest of the system. Recovery can also happen in an isolated fashion, letting the rest of the system continue unaffected.

A Deeper Dive into QFabric Architecture

This section provides a more detailed look at the QFabric architecture management and control planes.

Management Plane

QFabric Director compute clusters are composed of two compute nodes, each running identical software stacks. (Although the current implementation supports a maximum of two compute nodes per cluster, the architecture can theoretically support multiple servers.) Two of the compute nodes have a disk subsystem directly attached to them; the remaining compute nodes are diskless and therefore essentially stateless. Each disk subsystem consists of two 2 TB disks arranged in a RAID1 mirror configuration, and the contents across these subsystems are synchronously block replicated for redundancy.

A single global file system is layered on top of this highly available storage subsystem and is exported via Network File System (NFS) to the remaining compute nodes, ensuring that data can only be checked out and modified by one person/process at a time. The disk partitions contain the boot images as well as the Juniper Networks Junos® operating system software images, miscellaneous management infrastructure components, and a relational database for storing configuration data that can be queried. The contents of the disk subsystems are stored in a persistent fashion across reboots (see Figure 7).

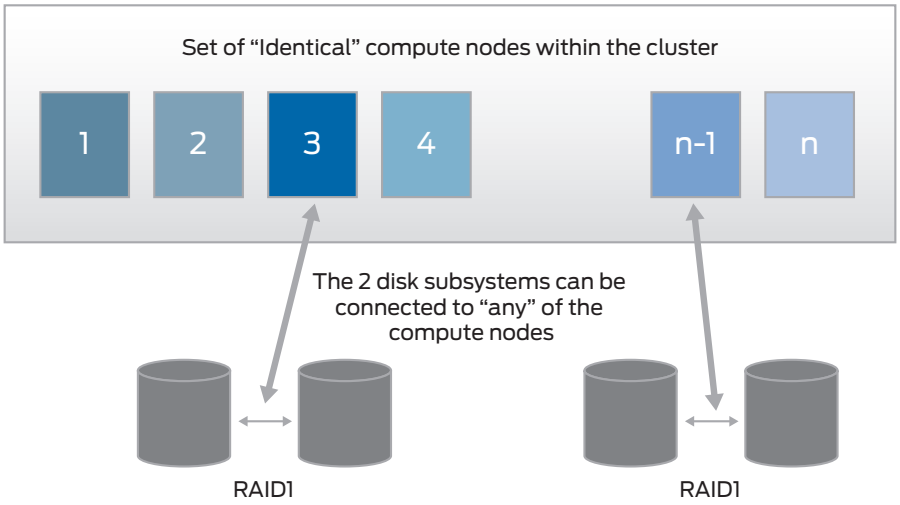


Figure 7: Scale-out compute cluster

The cluster of compute nodes that comprise the QFabric Directors "self assembles" at system boot-up from the images stored on the disk—without requiring user intervention.

Like the supervisor modules of a traditional chassis, the clustered QFabric Directors provide the standard Junos OS command-line interface (CLI), system log, and SNMP to the entire QFabric switch for configuration, management, monitoring, and provisioning. This single switch abstraction reduces by a hundredfold or more the number of network elements that have to be individually managed. When operators log into the QFX Series system, they can see all of the interfaces, as well as their state and metrics, from a single Junos OS CLI interface, following the same naming hierarchy used in traditional modular Ethernet chassis.

Multiple software modules reside across this compute cluster. The following subsections describe the three main clustered applications that present a resilient, single, Junos OS CLI to manage the entire distributed QFabric system.

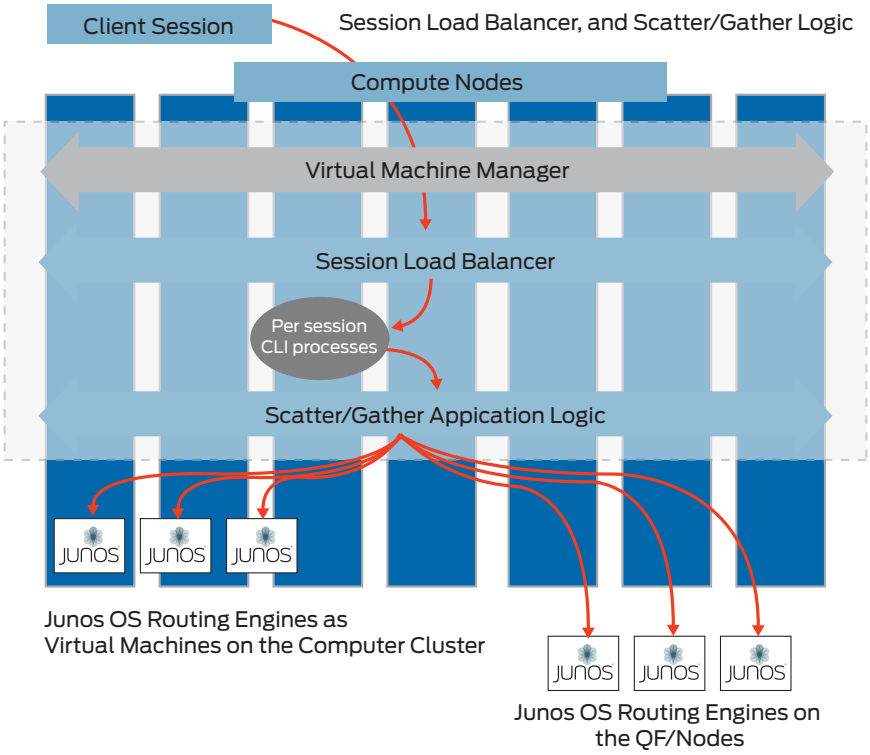


Figure 8: Clustered applications on the compute cluster

1. **Session load balancer**: CLI logins are load-balanced across the available compute nodes in the compute cluster by an application called the *session load balancer*. This not only reduces the load on each CPU in the compute nodes, but also provides a greater tolerance for compute node failures. Only those sessions on a failed node need to be restarted; the other user sessions are unaffected.
2. **Director software**: The director software contains the application logic that performs the scatter/gather function to implement the CLI for the distributed system. A single Junos OS CLI command is scattered across the individual nodes of the federated QFabric system. Upon successful local execution, results are returned to the gather logic and the final response is synthesized and presented to the user. This software ties together all of the elements needed to present a single Junos OS CLI view to the QFabric architecture operator (see Figure 9).

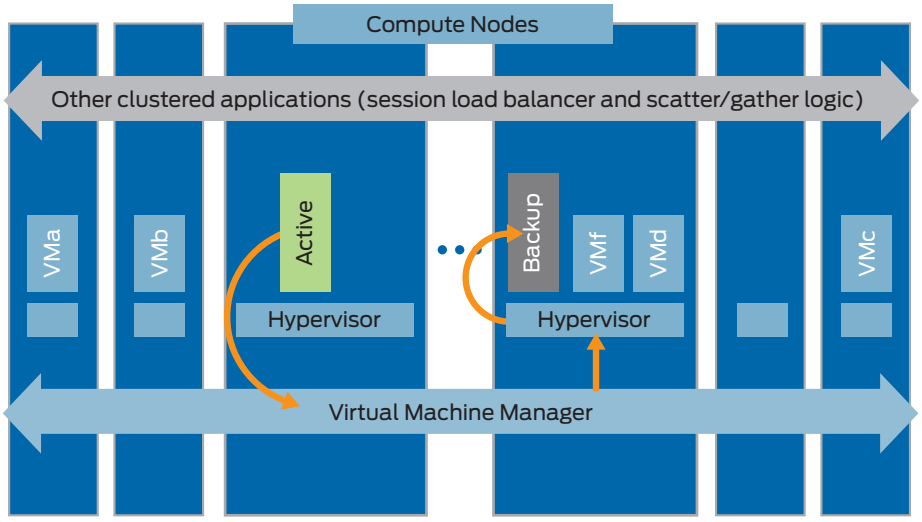


Figure 9: Junos OS Routing Engine virtual machines on a virtualized platform

3. **Virtualiz**ed compute nodes: Compute nodes are virtualized, allowing multiple Junos OS Routing Engine instances to be consolidated on a small set of compute nodes for greater compute efficiency by the QFabric Directors. (Details regarding the various Routing Engines are covered in the discussion of the control plane below.) The Junos OS instances run as virtual machines (VMs) on top of a virtualization platform—also called a clustered application—across the compute nodes. These VMs are organized as active/passive pairs across disjointed physical compute nodes to provide resiliency. A VM manager clustered application provides an API to the Junos OS instances for spawning additional VMs, and migrating VMs across the compute nodes to create and restore the active/backup pairs. Depending on their specific roles, the Junos OS instances on the compute cluster peer with the Routing Engines on the QFabric Nodes to selectively distribute network state information across the entire system.

Control Plane

The QFabric architecture is a distributed L2/L3 switch/router that scales from a small number of 10GbE ports to several thousand. Before looking at the scale-out model of peer-to-peer communications between Routing Engines in the system, this section will explain the three explicit user configurations of the QFabric Node, as well as the location and function of the Routing Engine in each.

User Configurations for QFabric Node

The QFabric Node can be configured in three distinct ways—server node group, redundant server node group, and network node group, as described in the following sections.

Server node group: A single QFabric Node that comprises a single logical edge entity in the QFabric architecture distributed system is a server node group (SNG). SNGs connect server and storage endpoints to the QFabric system. Members of a link aggregated group (LAG) from a server are connected to the SNG to provide a redundant connection between the server and the QFabric system. In use cases where redundancy is built into the software application running on the server (for example, many software-as-a-service (SaaS) applications), there is no need for cross QFabric Node redundancy. In those cases, an SNG configuration is sufficient. All QFabric Nodes in a QFabric system boot up as an SNG by default.

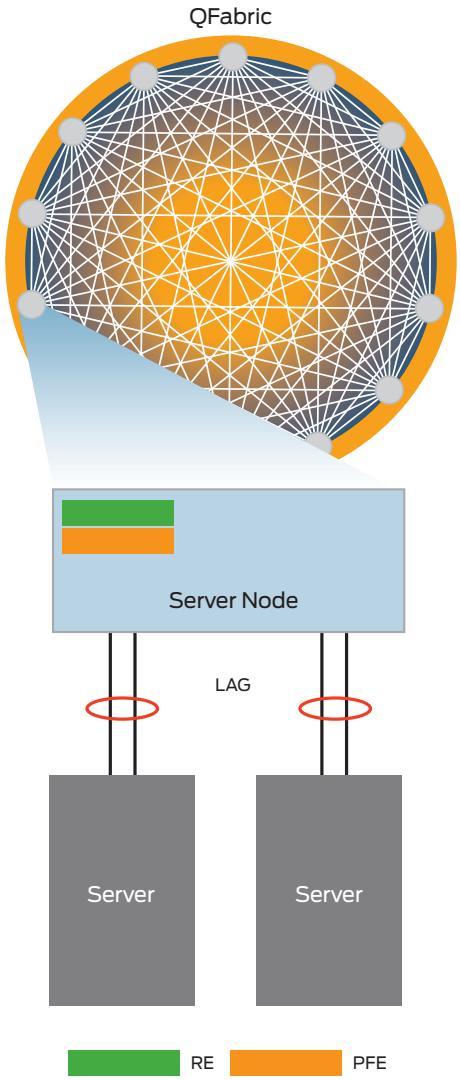


Figure 10: Server node group (SNG)

Redundant server node group: A pair of QFabric Nodes that comprise a single logical edge entity in the QFabric distributed system is called a redundant server node group (RSNG). Members of a LAG from a server are distributed across the RSNG to provide a redundant connection between the server and the QFabric system. In use cases where redundancy is not built into the software application running on the server, an RSNG configuration is desirable.

In most normal deployments, about 90% of all QFabric Nodes within a QFabric architecture will be configured as RSNGs. The RSNG's active Routing Engine is physically present only on one of the QFabric Nodes in the pair that makes up the RSNG (see Figure 11).

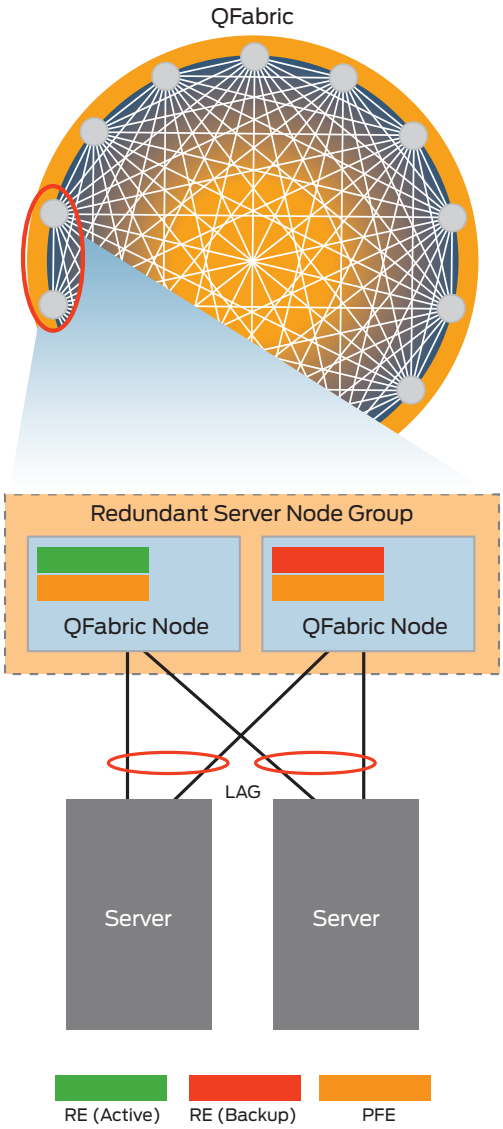


Figure 11: Redundant server node group (RSNG).

(R)SNGs only run server-facing protocols such as Link Aggregation Control Protocol (LACP), Address Resolution Protocol (ARP), and Internet Group Management Protocol (IGMP) snooping; they do not run STP, Protocol Independent Multicast (PIM), OSPF, or other such network protocols. (R)SNGs have mechanisms such as bridge protocol data unit (BPDU) guard and storm control to detect and disable loops across ports. Firewalls, load balancers, and other network devices can be connected to (R)SNGs via simple LAG connections.

Network node group: A set of QFabric Nodes running server-facing protocols as well as network protocols such as STP, OSPF, PIM, and BGP to external devices like routers, switches, firewalls, and load balancers is known as a network node group (NNG). NNGs can also fill the RSNG's function of running server-facing protocols. Only one NNG can run in a QFabric system at a time. With the introduction of physical partitions within the QFabric architecture (a roadmap feature), there will be a single NNG per partition. In most deployments, about 10% of the QFabric Nodes will be configured as NNGs (see Figure 12).

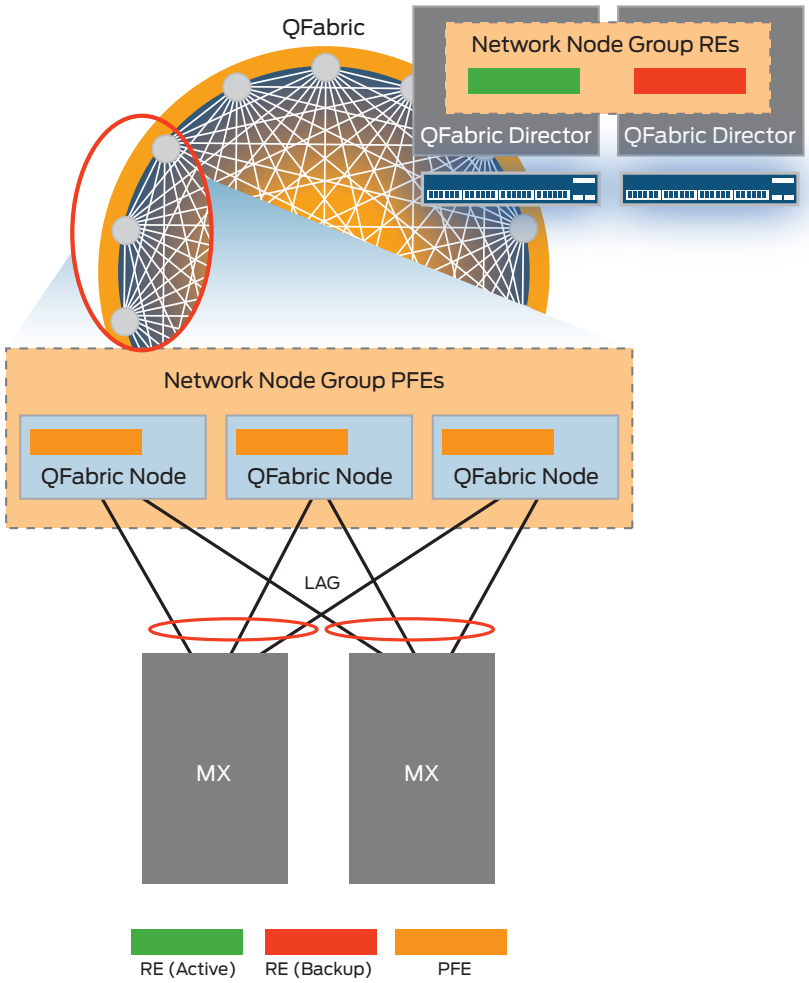


Figure 12: Network node group (NNG)

Although identical in their hardware makeup, one distinction between the RSNG and the NNG is the role played by their respective CPUs. In the RSNG, the CPUs function as in a pizza box form factor switch, performing both the Routing Engine and Packet Forwarding Engine (PFE) functionality. In the NNG, the CPUs perform only the PFE function. Just as in a modular switch, the Routing Engine function of the NNG is located externally in the QFabric Director cluster.

Every protocol stack in the QFabric system executes on a single CPU. However, given the large aggregate number of LACP and ARP sessions corresponding to the large number of server/storage devices, QFabric architecture distributes the overall load of running multiple sessions across all available CPUs in the system—hence the creation of Routing Engine functionality locally in the RSNGs. Every QFabric Node is an autonomous, full-fledged Junos OS switch/router. Given that QFabric architecture is essentially a large-scale host concentrator, the total number of OSPF adjacencies and BGP peers with external devices is small, which explains the location of the NNG Routing Engine functionality in the x86 processor-based QFabric Director.

Routing Engine Personalities

Within the context of QFabric Node configurations, there are a number of Routing Engine "personalities" that comprise the control plane functionality of the system.

1. **Server node group Routing Engine**: SNG Routing Engine performs the Routing Engine function on the (R)SNG QFabric Nodes. It runs as a master backup pair on an RSNG and as master only on an SNG.
2. **Network node group Routing Engine**: NNG Routing Engine performs the Routing Engine functionality on the NNG QFabric Nodes as described earlier. It runs as an active/backup pair of VMs across the physically disjointed compute nodes in the QFabric Director cluster.
3. **Fabric manager Routing Engine (FMRE)**: FMRE maintains accurate fabric inventory and topology for the entire system in a single, centralized database. Local fabric manager components on the QFabric Nodes and QFabric Interconnect learn the connectivity between themselves through periodic Hello messages sent on the fabric interconnect cables, then convey that information to the FMRE. The FMRE uses that information to form the link state database to reflect the overall physical topology of the QFabric architecture. The FMRE then calculates spray weights to control the distribution of traffic on the uplinks of every QFabric Node based on the information in the database. The FMRE runs as an active/backup pair of VMs across the physically disjointed compute nodes in the QFabric Director cluster.
4. **Fabric control Routing Engine**: The fabric control Routing Engine controls the exchange of routes between other Routing Engines in the system. It runs as an active/active pair of VMs across the physically disjointed nodes in the QFabric Director cluster. The logical peering between the collaborating Routing Engines that comprise the QFabric architecture control plane is shown in Figure 13.

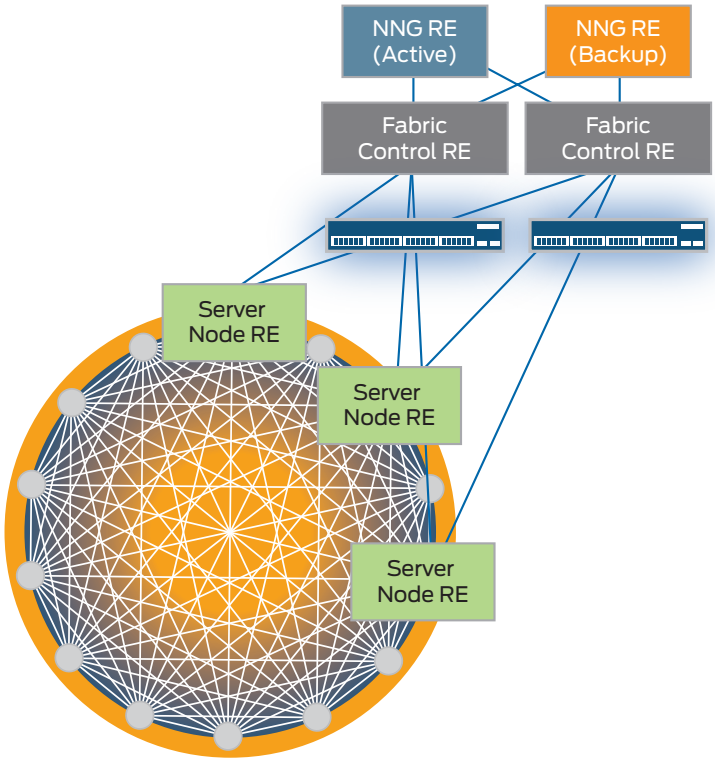


Figure 13: Network state exchange via peering between Routing Engines

QFabric Architecture Internal Protocols

Since the QFabric architecture components are not tightly coupled within the sheet metal of a single chassis, they must establish adjacencies by exchanging messages using software mechanisms. The following three internal-only protocols serve that purpose.

1. **QFabric System Discovery Protocol (logical connectivity):** A traditional modular switch uses simple internal-to-chassis mechanisms to establish logical connectivity between the line cards, switch fabric, and supervisors. However, these hardware assist mechanisms are not available in a loosely coupled distributed system such as QFabric architecture. Therefore, QFabric system discovery is based on a software mechanism derived from the IS-IS routing protocol. These IS-IS protocol messages are exchanged across the control plane by the QFabric Nodes, QFabric Directors, and QFabric Interconnects, resulting in the creation of an internal LAN on which all of the components reside. The FMRE then assigns private IP addresses to all members of this internal LAN, which allows system components to communicate with each other over reliable TCP/IP. This establishes logical connectivity between the system components, but not the physical topology of what devices are directly connected to which
2. QFabric Topology Discovery Protocol (physical connectivity): As described in the case above, the QFabric architecture lacks internal-to-chassis assist mechanisms to establish physical connectivity between line cards and fabric cards. Therefore, it requires a protocol that can discover the physical connectivity between QFabric Nodes and QFabric Interconnects. This protocol is based on the neighbor discovery portion of the IS-IS protocol. QFabric Nodes and QFabric Interconnects discover connectivity between themselves by exchanging Hello messages across the 40 Gbps data links connecting them. Each discovery forms a contributing piece of information which is communicated to the FMRE on the QFabric Directors, where the data plane topology is assembled. Upon creation of this database, the FMRE then programs the appropriate spray weights on the QFabric Nodes to load-balance uplink traffic across all available QFabric Interconnects, as well as links to those Interconnects (see Figure 14).

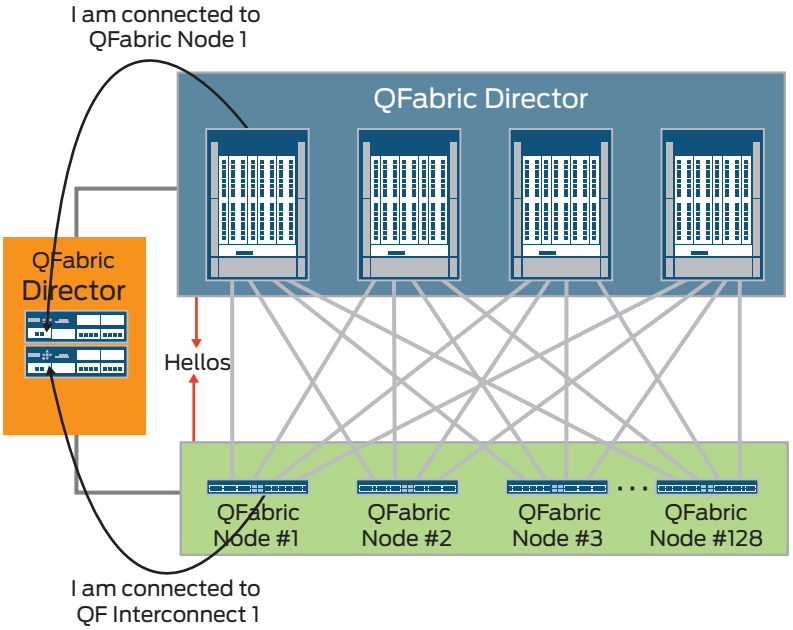
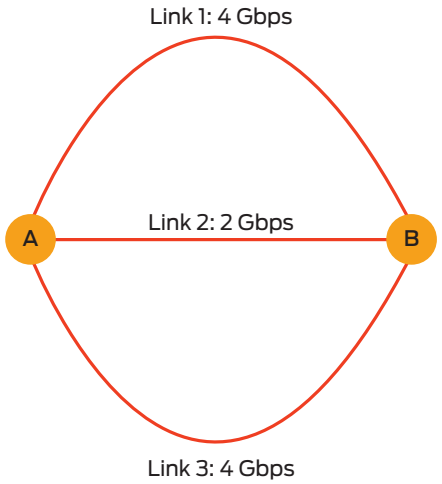


Figure 14: Fabric topology discovery in QFabric architecture

In the event of failure-induced unequal paths, the multipathing enabled by this centralized topology database works better than alternatives such as STP or equal-cost multipathing protocols such as TRILL. The following example shows how multipathing behaves in three technologies used today: STP, TRILL, and QFabric architecture.



	STP	TRILL	QFabric
Links used	1	2	All
Effective BW	4 Gbps	8 Gbps	10 Gbps

Figure 15: Advantages of multipathing with centralized fabric manager

3. **QFabric Control Protocol**: Given the large number of autonomous SNG and NNG Routing Engines, QFabric technology needs a mechanism to exchange network state between them. Furthermore, this mechanism must satisfy a number of requirements. For instance, the protocol must scale to thousands of Routing Engines, regardless of the scale of the initial system hardware implementation. Given the eventual need to create multiple partitions within a large QFabric architecture—each running a different version of Junos OS—it must have multiple version support. Instead of inventing different unique mechanisms for L2 and L3, the protocol must have a common mechanism to support the forwarding of reachability information for both layers. To support the large logical scale for use cases such as cloud and hosting services, the ability to handle overlapping IP addresses (across virtual routers) and overlapping MAC addresses (across VLANs) is of paramount importance. Finally, the protocol must be robust, hardened, and proven at large scale.

   The QFabric Control Protocol is based on BGP. The route reflector mechanism of BGP has shown over multiple years that it can scale very well in the toughest of network environments. BGP is an open standard and its ability to support type, length, and value (TLV) mechanisms makes it suitable to support multiple Junos OS versions running concurrently in the QFabric architecture. Juniper has added new address families to multi-protocol BGP to allow it to carry MAC routes in addition to IP and VPN routes. The mechanism of route distinguishers (RDs) allows the conversion of overlapping addresses into unique global addresses to be sent across a common infrastructure. Route targets (RTs) allow the application of filters while exporting and importing routes into/from the common infrastructure, thereby allowing the selective sharing of network state. In a classic L3 VPN, the user must explicitly configure parameters such as RDs and RTs, but that is not the case with QFabric technology. Instead, the QFabric architecture has user-configured VLANs and virtual routers for that purpose, so the creation and use of RDs and RTs remain totally transparent to the user.

The following diagram shows a small network topology after network reachability state for L2 has been shared between all SNG Routing Engines through the FCRE.

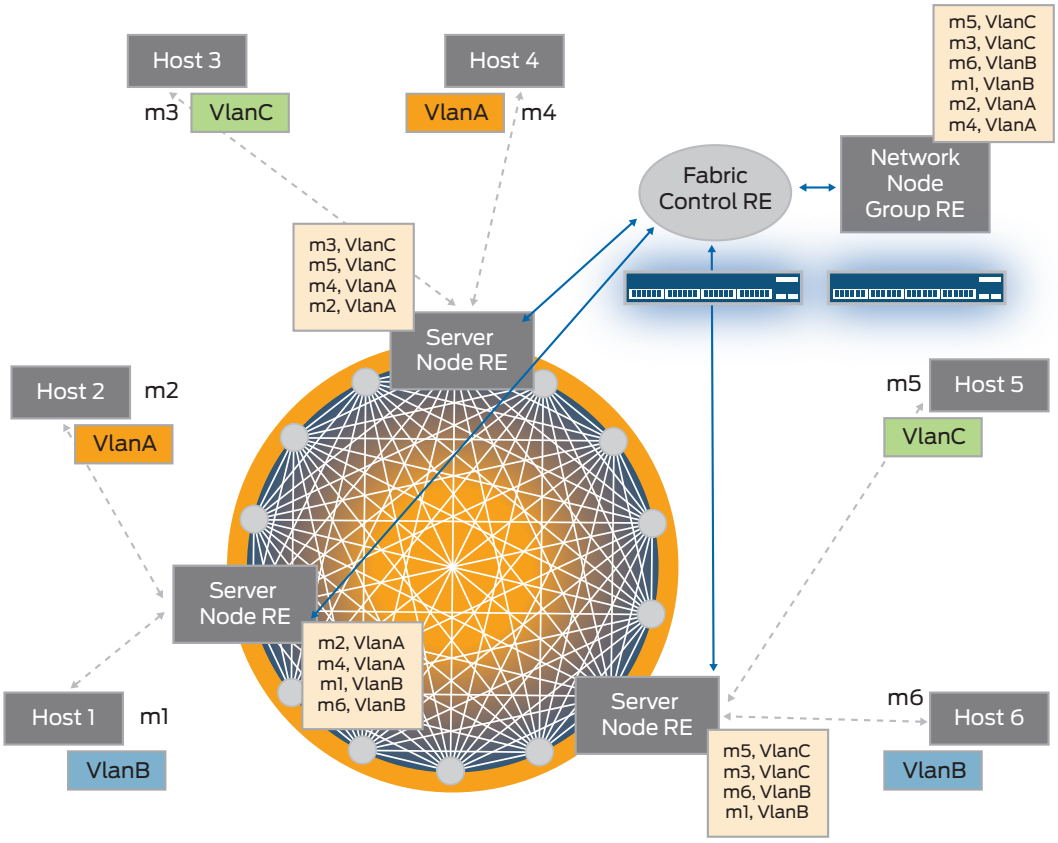


Figure 16: QFabric architecture route exchange model—Layer 2

The QFabric architecture engineering team has worked with other networking vendors and some of Juniper's customers to standardize the QFabric architecture control plane's procedures for routing L2 information using BGP MPLS-based Ethernet VPNs. When implemented on the WAN edge devices, E-VPN (originally referred to as MAC-VPN) can also be used for exchanging L2 information across data centers as shown at **http://tools.ietf.org/html/draft-raggarwa-sajassi-l2vpn-evpn-02**.

Packet Forwarding in QFabric Architecture

This section discusses how packets are forwarded in a QFabric architecture and compares the benefits of this approach to legacy Ethernet switching.

The task of forwarding packets in QFabric architecture, as in other switches, can be broken down into two distinct elements:

- How the control plane acquires forwarding state and sets up the forwarding tables in silicon-based packet forwarding engines
- The packet processor pipeline (in silicon) that a packet traverses on its way to being switched to its destination

Layer 2 Unicast Forwarding in QFabric Architecture

Learning in QFabric Architecture—A Control Plane-Driven Affair

First, let's look at how the Layer 2 unicast forwarding state is learned within the QFabric architecture. At its simplest, L2 forwarding information is an association of a <MAC Address, VLAN> pair to a physical port in the system. When new MAC Address M on VLAN V shows up at port P of QFabric Node A (as the source address in the Ethernet header), the Routing Engine for the QFabric Node (irrespective of whether the QFabric Node is part of an SNG, RSNG, or NNG) learns the new MAC address through the standard learning mechanism used in other modular switches. The Routing Engine for QFabric Node A updates the MAC address table in QFabric Node A's packet processor with the new entry <M, V>->P. The Routing Engine for QFabric Node A also uses the QFabric Control Protocol to advertise the <MAC Address M, VLAN V>-><QFabric Node A, port P> binding to other QFabric Nodes where VLAN V has a footprint. All "remote" Routing Engines then update the MAC address table in the packet processors of their respective QFabric Nodes with the entry MAC Address M, VLAN V>->QFabric Node A. As shown in Figure 16, the remote QFabric Nodes can all resolve the MAC address to the destination QFabric Node only. The destination QFabric Node A is the entity that provides the final resolution to the destination port. This principle of "information hiding" allows the QFabric architecture to achieve unprecedented scale-out by storing fine-grained forwarding state only where it is actually needed, not everywhere in the system (see Figure 17).

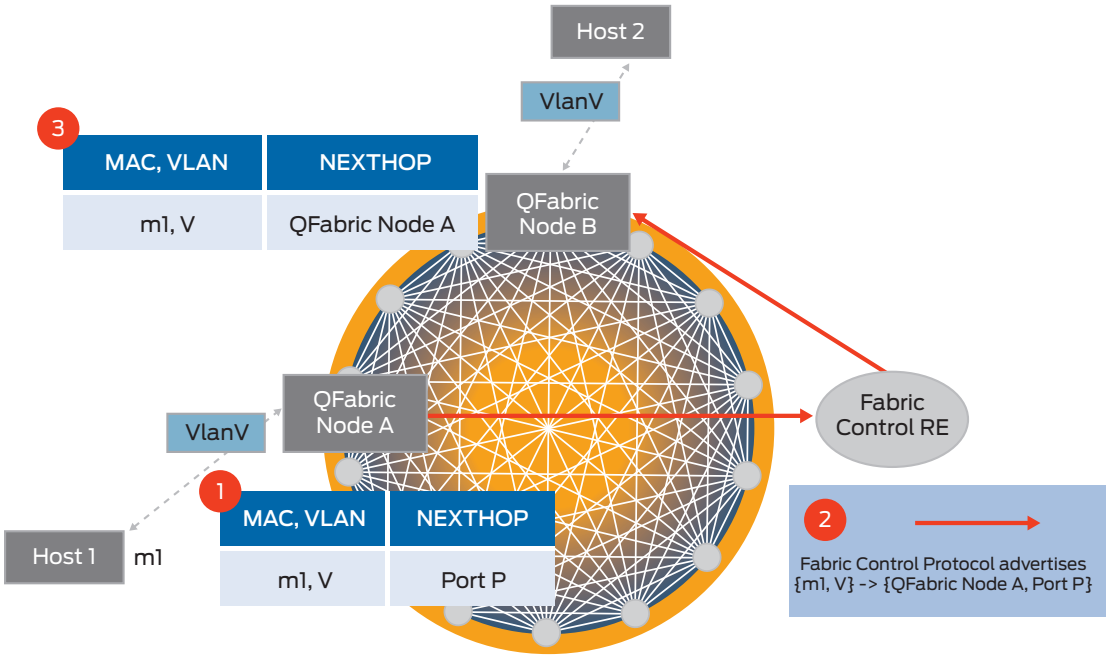


Figure 17: Control plane learning for Layer 2 unicast

Data Path for Layer 2 Unicast Forwarding

This section describes the different lookups that a packet undergoes while being switched to its destination. When a packet with destination MAC Address M on VLAN V (previously learned) shows up on a QFabric Node B, it matches the corresponding entry in the MAC address table on QFabric Node B. As described above, the result of the lookup is the resolution that the packet needs to be sent to QFabric Node A.

Each QFabric Node has a Remote QFabric Node Reachability table, which contains an entry for each remote QFabric Node, and enumerates all the different possible paths to reach the remote QFabric Node through the fabric. The Fabric Topology Manager programs the Remote QFabric Node Reachability table in all QFabric Nodes.

In this example, QFabric Node B consults its Remote QFabric Node Reachability table to see how many paths it has to get to QFabric Node A. It picks one of those paths based on a flow hash, prepends a fabric header onto the packet, and sends it to the selected QFabric Interconnect. The fabric header identifies the packet as destined for QFabric Node A.

When the packet arrives at the chosen QFabric Interconnect, it looks up its QFabric Node Reachability table to determine the various paths available to reach QFabric Node A. The QFabric Interconnect also picks one of the paths based on a flow hash. Note that the QFabric Interconnect only looks into the fabric header and does not look up the MAC address. This simplifies the design of the QFabric Interconnect, allowing it to attain interface densities than would not otherwise be possible.

Finally, when the packet reaches the egress QFabric Node A, it undergoes a MAC address lookup that results in the egress port being determined. The packet is sent out the egress port after performing any necessary rewrites.

Layer 3 Unicast Forwarding in QFabric Architecture

Layer 3 unicast forwarding follows the general contour of L2 unicast forwarding. At its simplest, L3 unicast forwarding information is an association between an IPv4 address and the MAC address, VLAN, and port: IP Address-> <MAC Address, VLAN, Port>.

When a host is attached to the QFabric architecture ARPs for its default gateway (which is QFabric architecture itself, for L3 forwarding), the ARP packet is trapped onto the Routing Engine of the QFabric Node where the ARP packet arrives. The Routing Engine forms the L3 entry IP Address-> <MAC Address, VLAN, Port>. It programs its packet processor with this entry and also uses the QFabric Control Protocol to send this L3 entry to all other Routing Engines. These "remote" Routing Engines then program the entry into their packet processors. Once again, the principle of "information hiding" is followed whereby the remote QFabric Nodes only associate the destination QFabric Node with the IP address, and the destination QFabric Node has the MAC, VLAN rewrite information, as well as the destination port.

In the data path, when a packet with the host's IP address as the destination IP address (and belonging to a different IP subnet than the source IP address) arrives at a QFabric Node, it triggers an IP lookup. It matches the host entry which results in a destination QFabric Node identity. The packet is sent toward one of the QFabric Interconnects based on a lookup of the QFabric Node Reachability table; the QFabric Interconnect then sends the packet to its final destination based on the QFabric Node Reachability table lookup. When the packet arrives at the egress QFabric Node, it matches the IP forwarding entry, which is the appropriate MAC address (destination). The VLAN is picked up from the entry, and the packet is sent out the appropriate egress port.

The QFabric architecture offers a number of benefits compared to an STP-based Ethernet network:

- QFabric architecture learns about source MAC and IP addresses via the control plane, not the data plane triggered learning which occurs in Ethernet. In a QFabric architecture, this removes the need for data plane flooding when packets from other points in the network are sent to the previously mentioned source addresses. Flooding of packets, when combined with other undesirable phenomenon such as transient loops, can cause undesirable network behavior. The QFabric architecture significantly reduces the probability of such incidents by minimizing data plane flooding.
- In traditional Ethernet, when links fail and STP has to recompute a new tree, forwarding tables are flushed to prepare for learning to start afresh. In a QFabric architecture, link failures do not lead to MAC table flushing, as QFabric technology uses control plane learning and is not dependent on a re-learned tree topology for the data to flow.

Broadcast, Unknown Unicast, and Multicast Forwarding in QFabric Architecture

There are three points where a multi-destination (broadcast, unknown unicast, or multicast) data stream is replicated within the QFabric architecture:

- **Ingress QFabric Node**: Replicates to all of its relevant local ports and onto one or more of its fabric uplinks (toward the QFabric Interconnects).
- **QFabric Interconnects**: Replicate between themselves toward all QFabric Nodes (except the ingress one) where a multi-destination stream needs to be sent. Only a single copy of the packet is sent to each interested QFabric Node. The replication load of a particular QFabric Node (i.e., the number of QFabric Nodes to which a particular QFabric Interconnect replicates) is based on how the control plane builds the multi-destination trees for that group (or VLAN).
- **Egress QFabric Node**: Receives a single copy of the packet from a particular QFabric Interconnect and then replicates that packet onto its local ports.

The problem of replicating a packet inside the QFabric architecture can be reduced to merely constructing a tree that spans all of the QFabric Nodes that need to send out a copy of that packet. The QFabric architecture is connected in a mesh topology (multiple pathes between any two QFabric Nodes), and a tree allows for loop-free forwarding for multi-destination packets. For the purposes of broadcast and multicast forwarding, the relevant subset of QFabric Nodes is determined by the footprint of the associated VLAN. For Layer 2 multicast, the relevant QFabric Nodes are a subset, as determined by IGMP snooping, of the QFabric Nodes for the VLAN.

When a multi-destination packet arrives at a QFabric Node, a lookup, which is based on VLAN ID in the case of broadcast and unknown unicast, and on the MAC address/VLAN pair for multicast, associates the packet with a TreeId. An index lookup of the TreeId gives the set of local ports on the ingress QFabric Node to which the packet needs to be replicated. The TreeId lookup also tells the packet processor to which QFabric Interconnects it should replicate the packet. The packet processor prepends the packet with a fabric header that contains the TreeId and sends it to the relevant QFabric Interconnects. On each QFabric Interconnect, a lookup of the TreeId determines the QFabric Nodes to which the packet needs to be replicated. Crucially, as in the unicast case, the only lookup in the QFabric Interconnect is on the TreeId and not on the Ethernet/IP headers of the packet. On the egress QFabric Node, a lookup of the TreeId again determines the set of local ports to which the packet needs to be replicated.

A subtle but interesting point to note is that forwarding state is "hidden" (not based on Ethernet/IP) from the core (QFabric Interconnect) in a QFabric architecture for both unicast and multi-destination forwarding. In several competing architectures—TRILL-based networks in particular—while there is state reduction in the core for unicast forwarding, the same is not true for multi-destination forwarding.

Conclusion

The QFabric architecture is purpose-built to address the challenges posed by new trends in data center networking. It is the industry's first truly scale-out architecture that allows users to build a network incrementally, with a single top-of-rack device enabling the connectivity to an additional rack of servers. With all components working together, the QFabric architecture behaves as a single logical switch that seamlessly integrates into the existing data center infrastructure.

The QFabric architecture achieves this scale by several innovations in the data, management, and control planes. The combination of smart edge, simple transport in the data plane, single switch abstraction in the management plane, and the single collapsed forwarding technology for L2 and L3 in the control plane result in an environment that is low latency, high cross-sectional bandwidth, simple to manage, resilient, and spanning-tree free. From small high-performance computing (HPC) clusters, to server virtualized and I/O converged enterprise data centers, to cloud mega data centers, QFabric architecture significantly reduces the CapEx and OpEx cost of building and running the network in modern data centers.

Appendix: New Architecture for Storage Networking

Today, customers are finally able to invest in standards-compliant, convergence-enabling network equipment that manages all types of storage traffic. Built upon both the IEEE DCB enhancements to Ethernet and the INCITS T11 FC-BB-5 standard for FCoE, the Juniper Networks QFX3500 Switch is the industry's first networking product to combine I/O convergence technologies with:

- Single ultralow latency ASIC (latency of less than 1 us)
- 1.28 terabits per second (Tbps) of bandwidth
- Diverse connectivity options of GbE, 10GbE, 40GbE connectivity options, and 2/4/8 Gbps FC

For more information about QFabric architecture and FCoE convergence, read the "**FCoE Convergence at the Access Layer with Juniper Networks QFX3500 Switch**" white paper.

The following figures show the topologies supported by the standalone QFX3500.

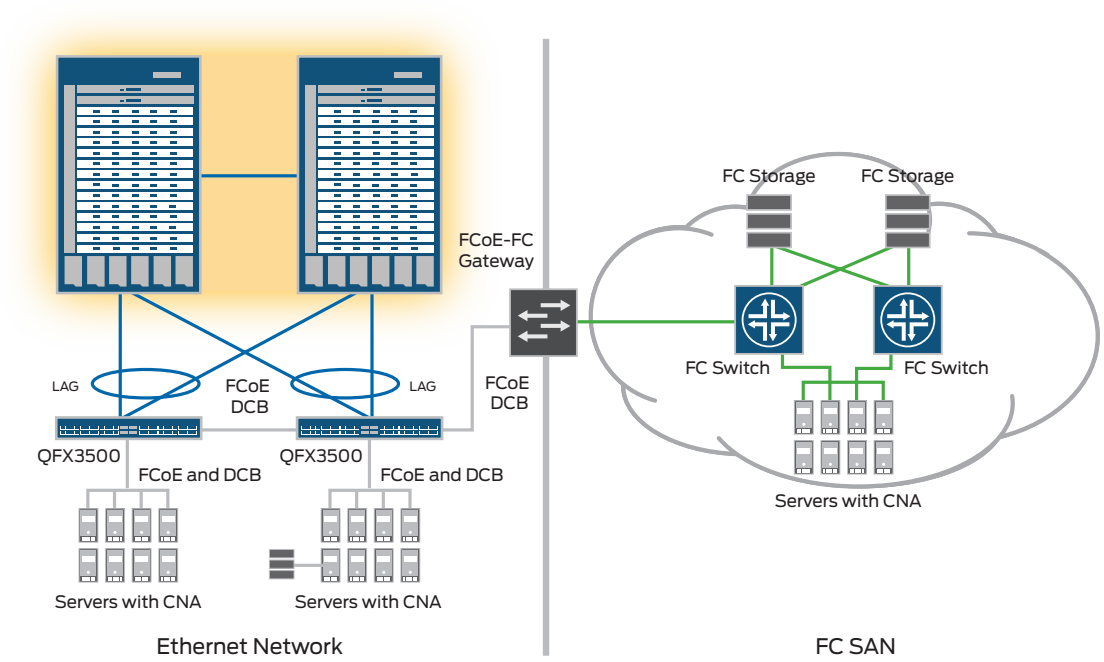


Figure 18: QFX3500 as a standalone FCoE transit switch

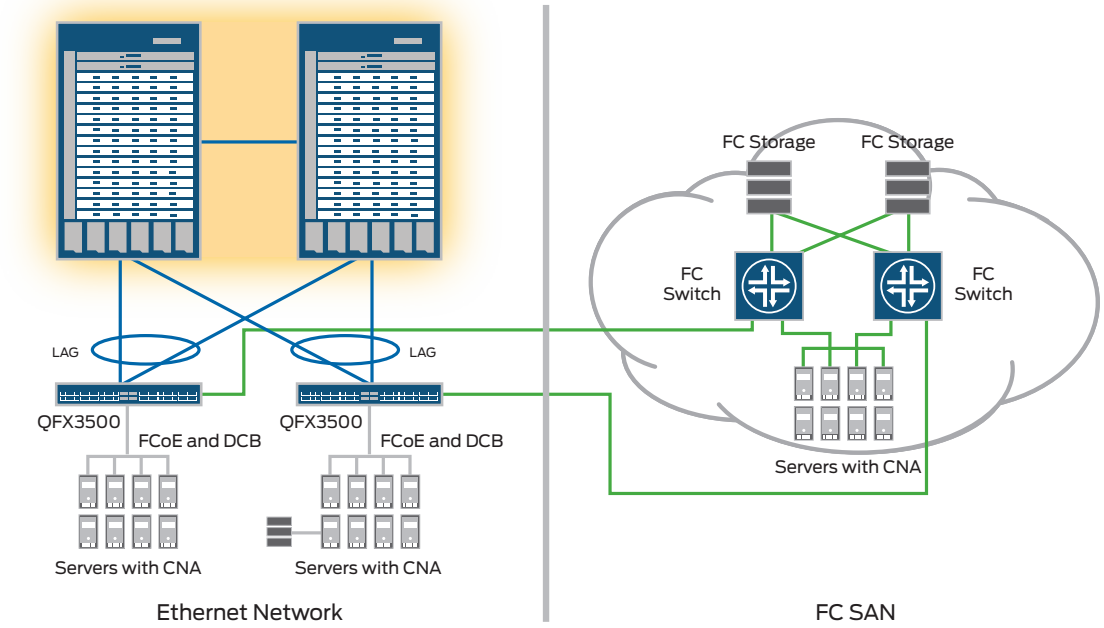
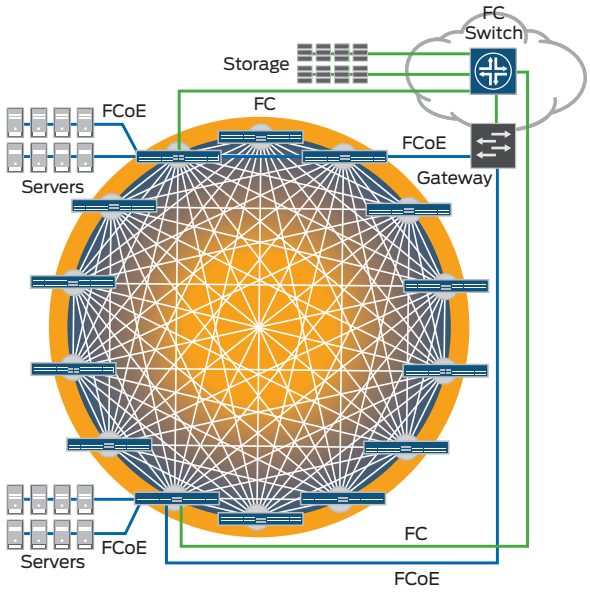


Figure 19: QFX3500 as a standalone FCoE-to-FC gateway

By means of a simple software configuration change, the same QFX3500 device now becomes the QFabric Node, forming the edge of the QFabric architecture. As a QFabric Node, it continues to support the functions of an FCoE transit switch and FC to FCoE gateway. In addition, it also supports cross-fabric FCoE transit switch functionality. All three deployment scenarios are shown in the figure below.



Convergence

FCoE Transit Switch

- Converged Enhanced Ethernet Standards based (CEE or DCB)
- Provides perimeter protection with FIP snooping

FCoE-FC Gateway

- Ethernet or fiber channel gateway with FC ports at the QFabric Node

Figure 20: QFabric architecture convergence

The networking industry recognizes that the first wave of standards listed above does not meet the needs of full convergence, and it is therefore is working on a second wave of standards, including FC-BB-6. It is Juniper Networks' goal to implement full Fibre Channel Forwarding (FCF) functionality in the QFabric family of products.

While it is possible today to build a large routed Ethernet network with thousands of standalone devices (from any vendor) and run a protocol such as OSPF between them, unfortunately the same is not true for Fibre Channel (FC) networks. FC networks do not scale beyond a few hundred (in theory) or tens (in practice) of switches today. The architectural innovations that enable the QFabric architecture to implement a scale-out distributed L2/L3 Ethernet system providing a single switch abstraction to the operator apply to FC in the exact same way.

Today, large-scale deployments have multiple server-only and storage-only racks spread across a data center and connected to each other by Ethernet and FC networks. As discussed above, the QFabric family of products can be deployed to build the network topologies for all of these scenarios (for example, FCoE transit switch and FCoE-to-Fibre Channel gateway). Also, QFabric architecture lends itself to build a single distributed FCF, solving the challenge of scaling a storage area network (SAN) with multiple standalone FC switches.

For smaller scale configurations where a single switch must connect to both servers and storage, Juniper believes that FC-BB-6 VN2VN will be the preferred FCoE end-to-end model in the years to come.

About Juniper Networks

Juniper Networks is in the business of network innovation. From devices to data centers, from consumers to cloud providers, Juniper Networks delivers the software, silicon and systems that transform the experience and economics of networking. The company serves customers and partners worldwide. Additional information can be found at **www.juniper.net**.

Corporate and Sales Headquarters
Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
Phone: 888.JUNIPER (888.586.4737)
or 408.745.2000
Fax: 408.745.2100
www.juniper.net

APAC Headquarters
Juniper Networks (Hong Kong)
26/F, Cityplaza One
1111 King's Road
Taikoo Shing, Hong Kong
Phone: 852.2332.3636
Fax: 852.2574.7803

EMEA Headquarters
Juniper Networks Ireland
Airside Business Park
Swords, County Dublin, Ireland
Phone: 35.31.8903.600
EMEA Sales: 00800.4586.4737
Fax: 35.31.8903.601

To purchase Juniper Networks solutions, please contact your Juniper Networks representative at 1-866-298-6428 or authorized reseller.

2000443-002-EN Mar 2012 Printed on recycled paper