NorthStar Controller—Multilayer SDN Coordination and Optimization

Controller-to-controller interface exchanges abstract topology information, providing a straightforward and highly scalable approach for coordination between both network layers

Table of Contents

List of Figures

Executive Summary

Multilayer coordination between the transport and IP/MPLS layers of a network is one of the most promising approaches to building a more optimized and simpler multilayer network architecture. However, this also presents a significant technical challenge, as it involves coordination between very different technologies on each of the network layers, each with its own approach and legacy for network control and management.

Transport networks have traditionally relied on the use of centralized network management through a network management system (NMS), whereas the IP/MPLS layer uses a distributed control plane in order to build highly robust and dynamic network topologies. These fundamentally different approaches towards network control have proven to be a significant challenge over the years when the industry has tried to realize a closer integration between both network layers. A considerable body of research and development has been spent in the industry on multilayer coordination over the preceding two decades, but as of today only a few successful examples of multilayer coordination exist, and then typically only in a single vendor environment.

The recent development of centralized SDN controllers for the IP/MPLS layer enables new opportunities for packet-optical coordination between the transport and IP/MPLS layer. This architecture relies on a controller-to-controller interface to exchange abstract topology information, thereby providing a straightforward and highly scalable approach for coordination between both network layers [1]. This whitepaper illustrates the architecture of the controller-to-controller interface that allows for the exchange of topology information between the transport and IP/MPLS layers. It also details some of the use cases that this exchange of information will enable.

Introduction

The first question to consider is whether to use centralized versus distributed multilayer coordination. In the past, multilayer coordination focused mostly on Generalized MPLS (GMPLS) [1] control planes in the transport and IP/MPLS network layers. GMPLS also provides a distributed control plane for the transport layer, which opens up the possibility of control plane integration between both network layers. Within the transport layer, GMPLS control planes have been successfully deployed for several years, and multilayer control through GMPLS-UNI [1]/GMPLS-ENNI [2] architectures have more recently been available as well. However, control plane integration between different vendors and technology domains is notoriously difficult and generally requires a lot of software development and integration testing due to the fact that the two control planes need to be tightly integrated with each other. This is particularly difficult since most transport platforms have highly centralized management architectures, whereas the IP/MPLS layer is inherently based on a dynamic control plane. Hence, every combination of transport and IP/MPLS across system vendors requires extensive dedicated testing (and retesting after software upgrades). This doesn't scale very well, and although GMPLS is a technically sound approach, it is nearly impossible to productize it in a multivendor environment.

Recently there has been a focus on a more centralized control of the IP/MPLS layer. This is one of the key characteristics of SDN, as centralized control generally makes it easier to program or influence routing decisions through a northbound interface for integration into management, provisioning, and orchestration platforms. A centralized controller for the coordination and optimization of the IP/MPLS layer enables a global view of the network topology and provides benefits through the use of highly optimized traffic engineering algorithms that make use of that global network visibility. A centralized IP/MPLS network controller also provides the possibility of multilayer control through an interface between the transport layer and IP/MPLS layer SDN controllers.

A multilayer controller-to-controller interface maintains the demarcation between the transport and IP/MPLS layers, while at the same time addressing the challenges associated with multilayer coordination and optimization. A controller-to-controller interface between a transport and IP/MPLS network controller is therefore a much simpler approach compared to GMPLS integration between network layers, predominantly as it does not involve the synchronization of state across multiple distributed control planes. There is no protocol interoperability needed between the transport and IP/MPLS layers, and no requirement for coordinated software upgrades. The technologies used in each of the network layers can therefore continue to evolve in a multivendor, multi-technology environment, each with its own approach for network control and management.

In short, multilayer control through a controller-to-controller interface addresses most of the same goals and use cases of control plane integration based on GMPLS and can therefore be a suitable alternative, specifically for multivendor deployments.

Active Stateful PCE Architecture

Juniper's SDN architecture for centralized WAN control and optimization in based on a Path Computation Element (PCE) architecture. The PCE is an entity (component, application, or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints [3]. A Path Computation Client (PCC) is any client application requesting that a path computation be performed by a PCE. In a centralized PCE architecture, the network elements act as PCC, where the PCC interface is integrated into the router operating system. The Path Computation Element Protocol (PCEP, RFC5440) enables communications between a PCC and a PCE [4]. It is a TCP-based protocol defined by the IETF PCE Working Group, and defines a set of messages and objects used to manage PCEP sessions. The PCE sends path setup requests to the PCC via PCEP, and the PCC sends report messages to the PCE for it to learn the label-switched path (LSP) state in the network.

Traditionally, PCE architectures have been passive and stateless, i.e., the PCE only computes the path on request of the PCC and is not aware of the LSP state already in the network. More recently, active stateful PCE architectures have become popular [5]. An active stateful PCE obtains the LSP and bandwidth reservations in the network by synchronizing the LSP state between the PCE and the PCC running on the network elements using PCEP. It can, therefore, actively install and modify paths in the network based on user input through the GUI or based on input received through the RESTful northbound interface.

The actual path setup in the network happens through RSVP-TE, as shown in Figure 1, which is a widely used protocol suite for traffic engineering in IP/MPLS networks that has been extensively deployed for many years in WAN deployments worldwide. This allows the upgrade of Brownfield network deployments by a simple software upgrade of the network node acting as PCC, and it enables a gradual transition of network services from distributed control to an SDN-enabled centralized control.

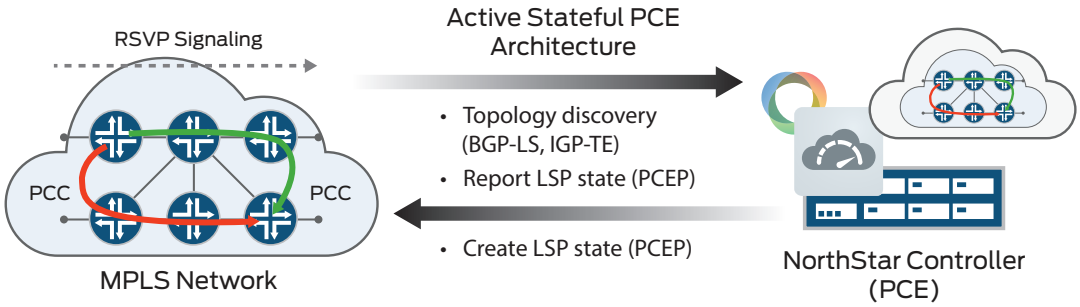

Figure 1: Interfaces between the IP/MPLS network and the NorthStar Controller

The NorthStar Controller

Juniper Networks® NorthStar Controller is an active stateful PCE and utilizes PCEP to obtain LSP state in the network for centralized control and optimization of network resources [6, 7]. Juniper Networks Junos® operating system integrates the PCC, allowing the NorthStar Controller to obtain the LSP state from any Juniper platform running a suitable Junos OS release, as well as any third-party routers supporting RFC5440. The NorthStar Controller itself is a software application running on any suitable third-party x86 hardware platform. NorthStar Controller peers with the network in order to obtain a global view of the network topology, typically using BGP-LS by importing the traffic engineering database (TED) [8]. Alternatively, it can also learn the network topology through the use of OSPF/ISIS-TE.

The combined visibility of the network topology and complete LSP state in the network allows the NorthStar Controller to provide more optimized traffic engineering (TE) compared to a distributed control plane where each network element has a complete view of the topology (within a domain/area) but only knows the local LSP state. NorthStar Controller, therefore, enables a diverse range of use cases such as diverse path computation, premium path computation, maintenance mode rerouting, bandwidth scheduling/calendaring, and bin packing or network defragmentation. The interface between NorthStar Controller and one or more transport layer SDN controllers further extends NorthStar Controller's visibility into the transport layer and allows it to address the multilayer coordination and optimization use cases described here.

Separation of Protocol Layers

Communication networks generally rely on a concept of separate protocol layers. The most common model to describe the separation of network layers is the OSI model, but in practice many different protocol stacks are deployed in today's networks. The concept of separate network layers was originally invented for scalability and interoperability considerations, and the ability to use standardized protocols. Over time, the split of communication networks into different layers has also influenced organization, security, and technology considerations. Different equipment is used to implement the different network layers, often built by different system vendors and operated by different parts of the operator's organization.

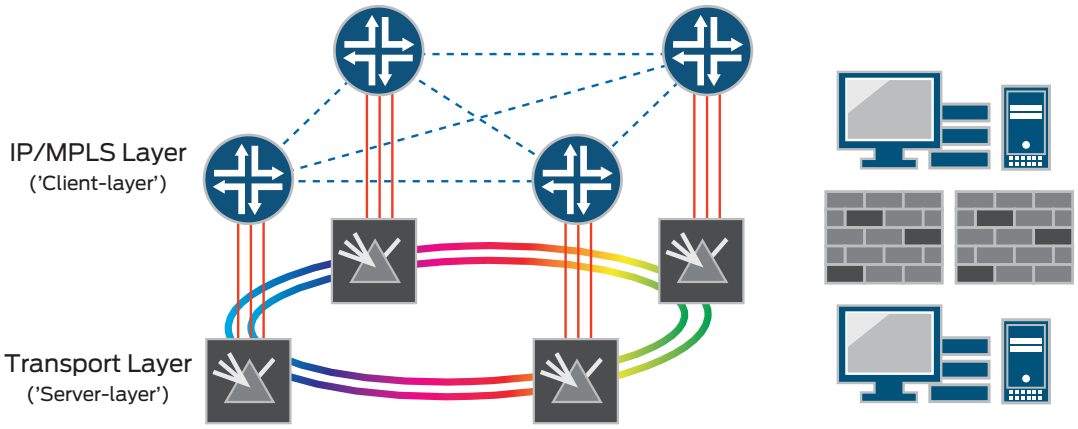

Figure 2: Transport and IP/MPLS network layers strongly differ in management and control interfaces

Transport and IP/MPLS are examples of different network layers that effectively use a client/server approach, where the IP/MPLS layer is a client of the transport server layer as illustrated in Figure 2. The client layer, in this configuration, has no or only very limited knowledge of the server layer, i.e., the IP/MPLS layer is "blind" to the actual topology and resource availability on the transport layer. Unfortunately, this loss of information also complicates network design and operations, and in particular the disconnected network layers are a burden for multilayer optimization (MLO) of network resources.

The lack of transport layer topology and resource availability information on the IP/MPLS layer results in more complicated service provisioning and disconnected planning processes. It is often desirable that the client layer influence the routing of the services it provides across the server layer. For example, services sometimes need to use paths that are as disjointed from each other as possible. If an interface or link in the network fails, the distributed IP/MPLS control plane can automatically restore traffic to an alternative path using MPLS fast reroute (FRR) or similar restoration mechanisms. However, for this restoration to be successful, it is critical that both active and standby paths do not share the same resources on the transport layer, i.e., Shared Risk Link Group (SRLG) information, which would result in a simultaneous failure of the active and standby path and therefore a complete loss of connectivity. Traditionally, the IP/MPLS layer has no direct visibly into SRLG information, as it only knows if a circuit is "up" or "down" and does not know which circuits are fate sharing. It can learn SRLG information by including the information in the IGP [9]. However, this requires manual configuration of the SRLG information based on data obtained from the transport and/or fiber topology, which is a labor-intensive and error prone process. In addition, for transport networks that use a reconfigurable optical add-drop multiplexer (ROADM) to periodically optimize wavelength planning, the SRLG information is not static and this necessitates an automated update mechanism of the SRLG information in the IGP.

Different services provided by the IP/MPLS layer can also have very different routing requirements. For example, some services need to be optimized based on lowest cost criteria, while other services have the best delay characteristics. In particular, the exchange of delay information enables more service-aware TE that allows the IP/MPLS layer to differentiate traffic engineering between lowest cost and premium lowest latency paths. As a result of the limited information exchange between both layers, it is typically a highly manual process involving offline planning and often relying on disconnected data sets to identify what services follow the most desired path across the transport layer. This requires an exchange of metrics between both network layers, ideally an automated one.

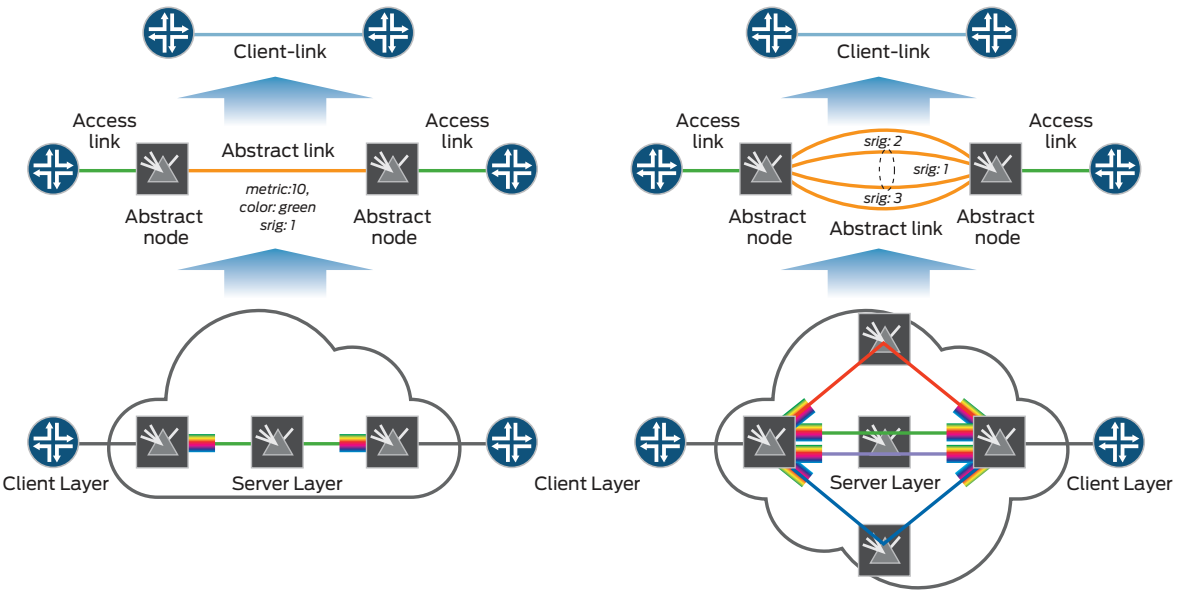

Figure 3: Abstraction of the transport-layer topology: point-to-point link (left), and meshed network (right)

Network Topology Abstraction

Despite the previously mentioned drawbacks, it is generally still desirable to limit the information exchange between the transport and IP/MPLS layers to the information that is directly useful to improve TE. This allows for higher scalability in large networks, but also addresses any organizational or security concerns that might exist when detailed configuration information is shared between both network layers.

This can be achieved by summarizing the detailed design of the transport layer topology to the minimum set of information required to address relevant multilayer TE use cases. The IP/MPLS layer only requires network topology and a limited set of metrics from the transport layer, and any more detailed information does not influence or improve traffic engineering accuracy. Detailed information on network element connectivity and optical transmission impairments of the transport layer can therefore be omitted from the information that is shared between the transport and IP/MPLS network layers. Instead, the transport layer (server layer) shares an abstracted topology model with the IP/MPLS layer (client layer). This abstracted topology model consists of a set of abstract links that represent the end-to-end reachability on the server layer, as well as the metrics of these links such as bandwidth, latency, SRLGs, and so on.

Figure 3 shows the abstraction of the transport layer into a set of abstract nodes and links. A link that connects a transport-layer node and a node in the IP/MPLS layer is referred to as an access link. Every node on the transport layer that has one or more access links to the IP/MPLS layer is defined as an abstract node. Any transport node that does not have any access link(s) is omitted from the abstracted topology, as it is not required to define the topology mapping between both network layers.

Abstract links are either actual or potential end-to-end links within the server layer that connect two abstract nodes together. An abstract link can be a direct point-to-point connection between two abstract nodes, but it can also represent the connectivity through a complex meshed network topology with multiple inline (ROADM) network elements. Multiple abstract links can share the same server-layer links, in which case they are part of the same SRLG. Any link or node in the server layer that is shared by multiple abstract links can be the basis for a separate SRLG, and an abstract link will typically be associated with a string of SRLGs.

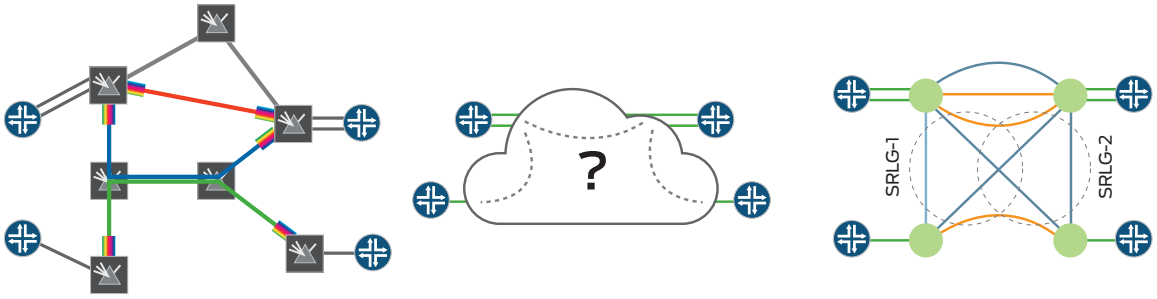

Figure 4: Actual transport and IP/MPLS network topology (left); view from the IP/MPLS layer without abstract topology exchange (middle); and view from the IP/MPLS layer with abstract topology exchange (right)

The reachability information in the abstract topology model covers both actually existing transport layer connectivity, as well as potential connectivity. An actual abstract link is a link that can be used by the client layer to realize connectivity; for a transport network, this will normally imply that one or more wavelengths are installed between the near-end and far-end transport nodes. Potential abstract links are all transport-layer circuits that can be supported given the server-layer topology within the restrictions imposed by ROADM connectivity, or optical reach restrictions, but where no circuits are installed. Each of the abstract links is described with a set of common "attributes" such as the bandwidth, cost, latency metrics, link coloring, and SRLGs. As a result, this abstracted topology concept considers all physical restrictions of the transport layer, while hiding the complexity of the optical transmission impairments and ROADM connectivity from the IP/MPLS layer.

Figure 4 gives an example of a transport and IP/MPLS network topology and the visibility on the IP/MPLS layer with and without exchange of the abstract server-layer topology. The set of abstract links defines the full set of actual and potential connections through the meshed network. Using this abstract server-layer topology, the IP/MPLS layer can now determine which server-layer connections are not part of the same SRLG and therefore are not fate sharing. In this way, it can guarantee diversity of a pair of LSPs without any manual SRLG configuration.

Abstract Topology Exchange

The information exchange between the transport and IP/MPLS layers is illustrated in Figure 5. The NorthStar Controller obtains the topology information of the IP/MPLS layer by peering with the routers through, for example, BGP-LS, and it obtains the LSP state through the PCEP. The transport SDN controller will learn the transport network topology through southbound interfaces such as SNMP, OpenFlow, or NETCONF. The protocols used by each of the SDN controllers for control and configuration of the network, therefore, do not need to be identical, which is greatly beneficial in a multivendor multi-technology environment.

The topology exchange between the transport SDN controller and the NorthStar Controller is implemented by describing the abstract topology in a YANG model. YANG ("Yet Another Networking Grammar") is a data modeling language described in RFC6020 [10]. It is used to model configuration and state data manipulated by the Network Configuration Protocol (NETCONF, RFC6241) [11], NETCONF remote procedure calls, and NETCONF notifications. YANG generates XML-based configurations with a C-like syntax and a hierarchical structure. This makes it easy for operators to understand and engineers to implement YANG data models. The YANG data model describing the transport network in the form of an abstracted network topology is detailed in IETF draft-ietf-teas-yang-te-topo-01 [12].

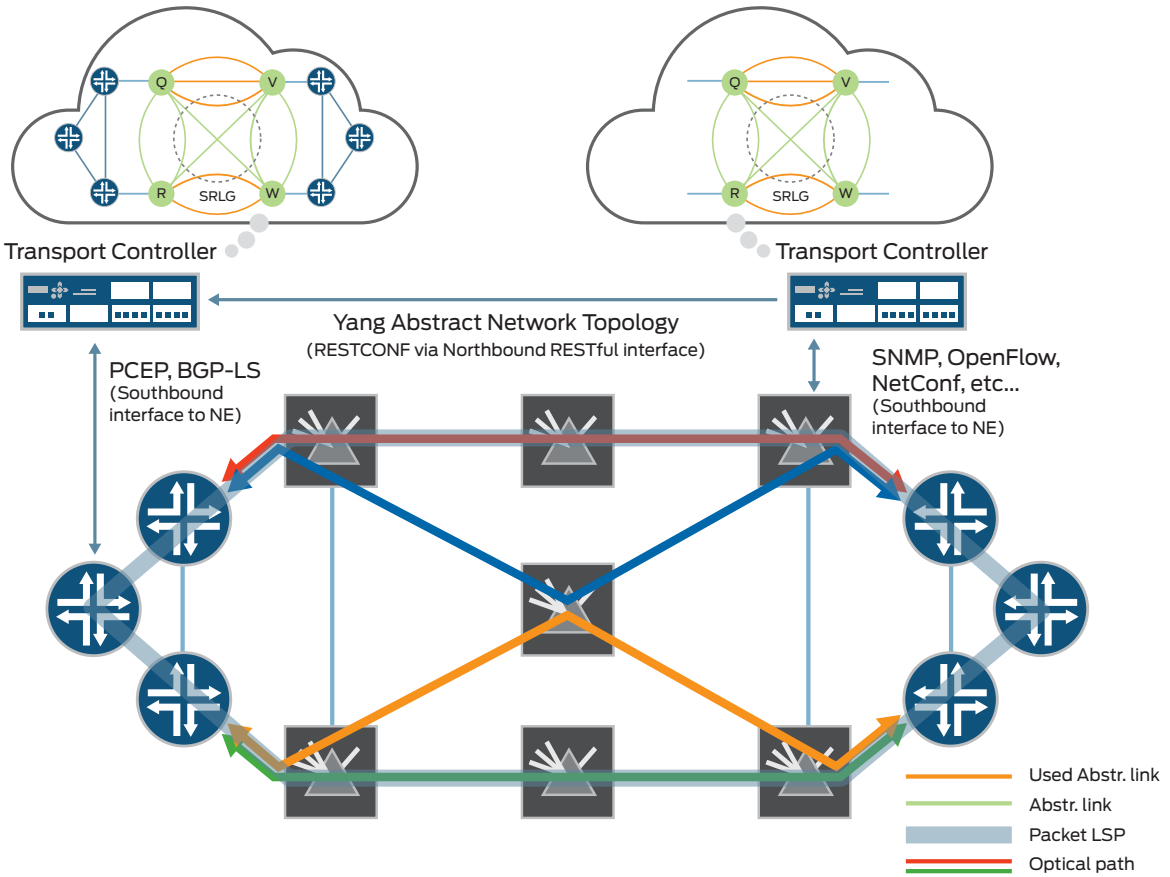

Figure 5: Controller-to-controller abstract topology exchange between the NorthStar Controller and a transport SDN controller

The abstract network topology is exchanged between the transport SDN controller and the NorthStar Controller via either a RESTCONF [13] or REST [14] interface. Representational State Transfer (REST) is an architecture style for information exchange mainly used for the design of networking applications, where APIs that adhere to the REST architectural constraints are called RESTful APIs. RESTful applications typically use HTTP requests to post data (create, or update, or both), read data (e.g., make queries), and delete data. A key property of a RESTful interface is that it is stateless, i.e., each request from any client contains all the information that is necessary to service the request. The session state is held only in the client. This is different from more traditional network management protocols such as SNMP that use a sequence of request and response calls between client and server and where state is held in both the server and client.

RESTCONF provides a lightweight alternative with a limited subset of the transaction capabilities of the NETCONF protocol. It is a REST-like management protocol running over HTTP that describes how to map a YANG specification to a RESTful interface. It defines a simplified transaction model that allows for the basic create, update, read, and delete operations on a hierarchy of conceptual resources.

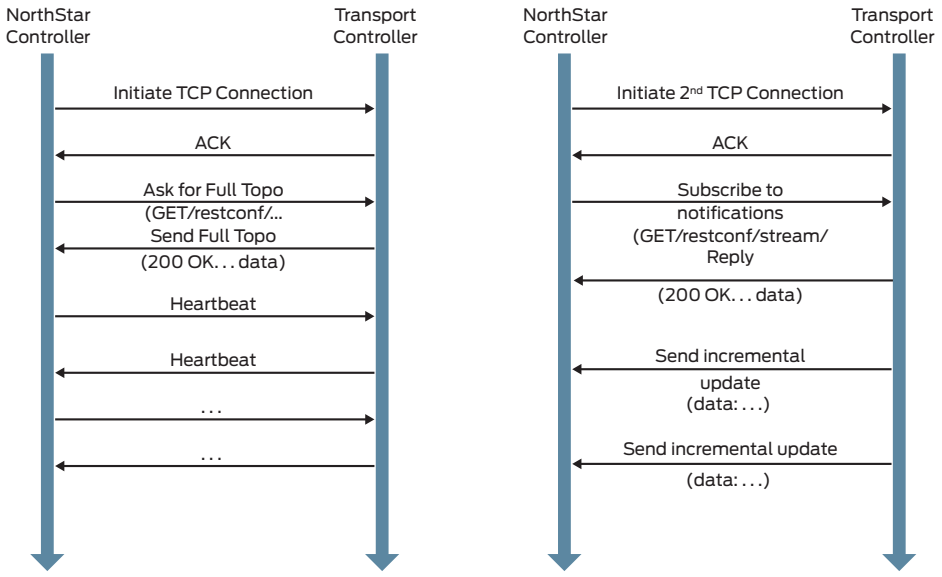

Figure 6: Initial topology synchronization (left) and incremental topology updates (right)

NorthStar Controller obtains the abstract network topology from the transport-layer controller by sending a GET request as shown in the example workflow diagrams in Figure 6. The initial GET request will result in the retrieval of the complete abstract network topology by the NorthStar Controller. It will then use the abstract topology to update its topology database in order to form an abstract server-layer network model. This enables NorthStar Controller to optimize TE on the IP/MPLS layer taking into account the server-layer abstract topology, in this case by signaling a pair of diverse LSPs between both endpoints.

When changes happen in the transport layer, the abstract network model is updated instantaneously such that NorthStar Controller can properly adapt the traffic engineering on the IP/MPLS layer. Changes to the abstract network topology, via incremental updates, leverage a PUSH model through a notification subscription model. This allows individual abstract links and abstract nodes to be updated, but links and nodes can also be created and deleted as part of the topology updates.

Multilayer Coordination Use Cases

The RESTful interface to exchange abstracted topology information between the transport and IP/MPLS layers improves coordination between both network layers, and it enables more optimized and simpler multilayer network architectures. This includes use cases such as multilayer topology visualization, path diversity and maintenance, as well as visibility to transport-layer restoration and protection schemes. The different use cases are described in detail below.

Multilayer Topology Visualization

For a consistent visualization of the multilayer network topology, it is beneficial to automatically retrieve correlated IP/MPLS and transport-layer topology information from the network, instead of collecting such information manually from each of the network layers individually. Without a well-defined model for multilayer topology representation, the information needs to be obtained from databases maintained by each network layer separately and then stitched together—which is often a labor-intensive and error prone process.

NorthStar Controller uses the abstract topology information to create a multilayer topology representation of both the IP/MPLS and transport layers of the network. Figure 7 shows an example NorthStar Controller GUI representing both the IP/MPLS (red) and transport (green) layer topology. A multilayer network topology simplifies the correlation of information and events between network layers. For example, a failure on the transport layer that causes several IP/MPLS layer circuits to fail is now easily identifiable as the root cause. This allows an IP/MPLS-layer network operator to more easily identify any weak points in the IP/MPLS-layer topology, as well as more quickly identify the root cause of any failure scenarios.

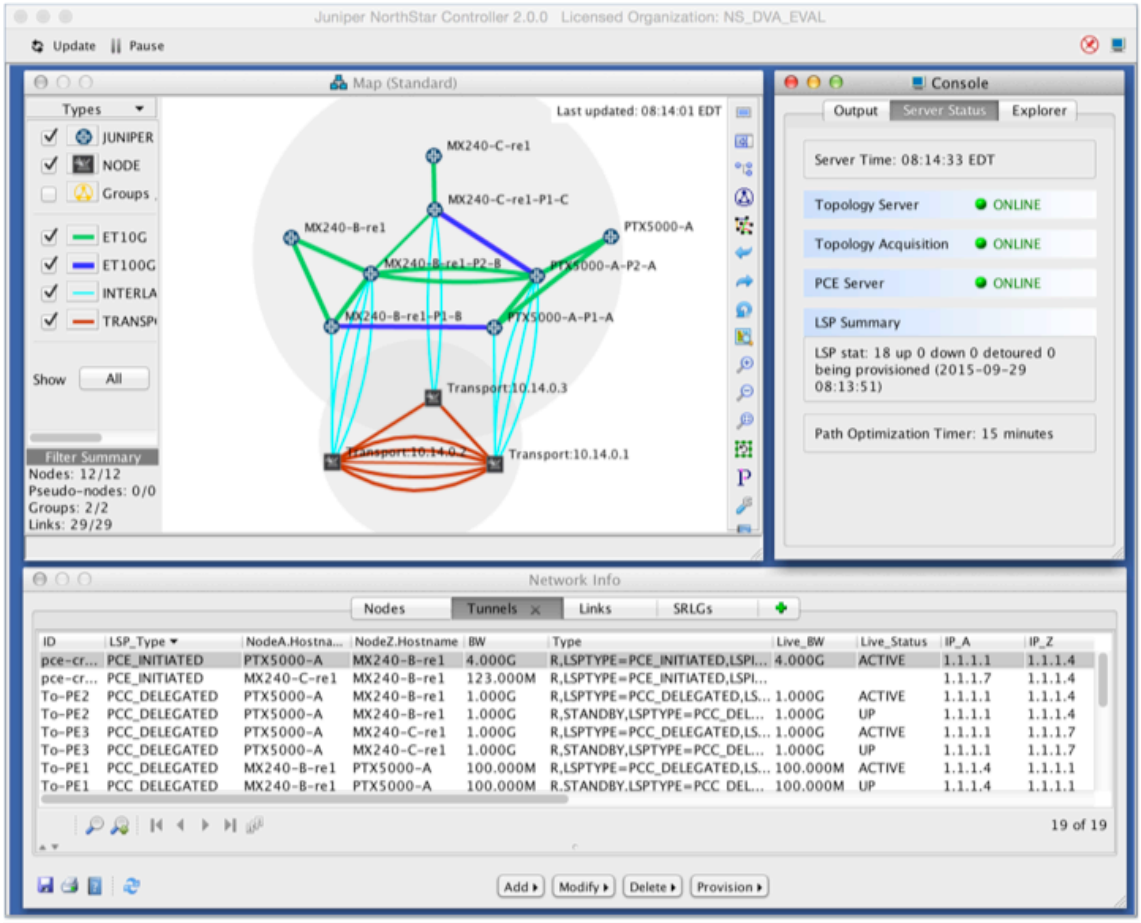

Figure 7: Example GUI of a multilayer network topology in the NorthStar Controller.

Multilayer Path Diversity

The controller-to-controller interface between NorthStar Controller and a transport SDN controller allows for an automated exchange of SRLG information between the IP/MPLS and transport layers. Every abstract link is a member of one or more SRLGs. Each SRLG effectively represents a link or node on the transport layer that is traversed by multiple abstract links. With this information, NorthStar Controller can avoid using abstract links that share any SRLGs and thereby ensure that there is no fate sharing when it computes diverse paths or when computing a backup path to an already existing primary path. When the SRLG information changes on the transport layer, for example due to a restoration event that switches wavelengths to a different path, the abstract topology is automatically updated. Any changes to the SRLG information are thus immediately visible to the IP/MPLS layer through the PUSH model, which can subsequently optimize any affected LSPs.

NorthStar Controller has true end-to-end visibility of the complete IP/MPLS layer topology and can therefore provide correlation between SRLG information from different sources. The transport SDN controller, on the other hand, is limited to its own transport domain and only has a partial view of the network topology and SRLG information. The IP/MPLS-layer controller might, therefore, need to correlate SRLG information from different sources in order to obtain a complete view of fate sharing in the network. For example, there might be fate sharing between a transport link using dense wavelength-division multiplexing (DWDM) interfaces and a router-to-router interconnect using grey client interfaces,

when both fiber pairs are part of the same fiber-optic cable. Similarly, a single fiber-optic cable might contain fiber pairs used independently by equipment from different system vendors. This is a key difference between multilayer integration based on the exchange of an abstract topology and a traditional user-to-network (UNI) interface that requests path diversity from the server layer without having actual visibility into the server-layer (abstract) topology.

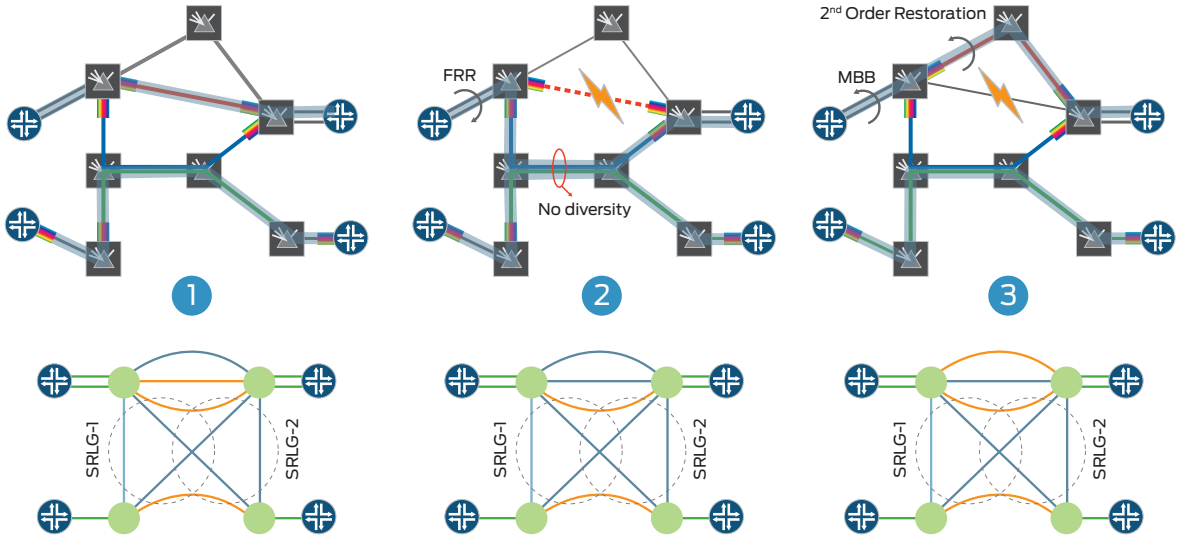

Figure 8: SRLG information exchange during transport-layer restoration (2). Both the transport and IP/MPLS network exchange topology changes (3)

Visibility to Transport-Layer Restoration

In case of outages on the optical layer, for example a fiber cut, first order traffic restoration will normally take place on the IP/MPLS layer using restoration mechanisms such as FRR that can converge in less than 50 ms. After first order restoration, the (worse-case) interface utilization will typically have increased due to the loss in network capacity, and it is not desirable for the network to run in such a state for a long period of time. The use of second order restoration on the optical layer can make (part of the) lost optical connectivity available again in order to restore the network to normal utilization levels, thereby eliminating the need for additional protection bandwidth on the IP/MPLS layer.

The transport layer can either use a distributed GMPLS control plane or the centralized controller to make restoration decisions. Typically, the endpoints of the wavelength circuit will remain the same and a different path through the meshed network is selected by appropriate switching of wavelength paths in the ROADMs. The access links and the stitching of the transport and IP/MPLS topology do not change, only the abstract link changes. The update of the abstract topology after second order restoration is immediately visible to the IP/MPLS layer, which can subsequently optimize the LSPs and reduce the worse-case interface utilization.

The abstract topology update resulting from topology changes on the transport layer is exemplified in Figure 8. Initially, two LSPs are signaled that are diverse from each other on both the IP/MPLS and transport layer. Once the upper (red) optical circuit fails due to a fiber cut, the LSP is restored through MPLS fast reroute (FRR) to a backup path using the blue optical circuit. However, the resulting pair of LSPs is now no longer diverse and therefore a possible second outage might bring down both LSPs at the same time—resulting in a complete loss of east-west connectivity. After second order restoration, the failed red optical circuit is switched to a different path in the transport layer, restoring the diversity on the transport layer. Through an update of the abstract topology, NorthStar Controller learns that a diverse optical path is now available, which allows it to re-optimize the pair of LSPs to diverse paths using hitless make-before-break (MBB) restoration.

Visibility to Transport-Layer Protection In Use

The limited visibility between both network layers tends to result in an inefficient use of resilience schemes. Resilience is an inherent part of both the transport and IP/MPLS layers and coordination between both layers is therefore essential. Simple protection switching solutions such as 1+1 optical path protection make it straightforward to implement point-to-point link resilience on the transport layer. Restoration, on the other hand, is typically better implemented on the IP/MPLS layer, since it allows for full end-to-end diversity. Knowledge of any resilience schemes in use on the transport layer is therefore essential, so that restoration and protection schemes can cooperate effectively.

Typically, the IP/MPLS layer has no visibility into the use of resilience mechanisms on the transport layer. For example, when circuits in the transport layer make use of 1+1 path protection, there is a significant difference in outage probability of the protected circuit relative to the unprotected circuits. This is useful for the NorthStar Controller to take into account when computing diverse paths—particularly when the topology does not allow for full link diversity, and it might be

acceptable that both active and standby LSPs use the same protected resources on the transport layer. In some other cases, it might be beneficial if a primary and secondary path in an active/active configuration use the same protected resources in order to minimize the delay difference between both paths. The packet loss resulting from protection switching on the transport layer can also potentially trigger restoration on the IP/MPLS layer, which might result in undesirable network instability without the use of hold-off timers to suppress restoration.

By setting a "protected" flag for a particular abstract link, the coordinated multilayer protection allows the transport layer to notify the IP/MPLS layer that this abstract link is protected. NorthStar Controller can then consider this as part of the path diversity computation by ensuring adequate diversity and protection across both network layers. When the transport layer switches to the secondary path in case of a link failure, it will remove the "protected" flag and update the abstract topology accordingly. This allows the NorthStar Controller to restore diversity on the IP/MPLS layer.

The exchange of resilience information between both network layers is shown in Figure 9. The upper (red) optical circuit uses 1+1 path protection with a primary and secondary path, and protection switching is triggered once the primary circuit fails. Since the IP/MPLS layer is now aware of the transport-layer protection in use, it can suppress any packet-layer restoration so that nothing changes on the IP/MPLS layer. The blue/green circuits do not use transport-layer protection, and once the blue optic circuit fails, the LSP is restored through MPLS FRR to a backup path using the green optical circuit.

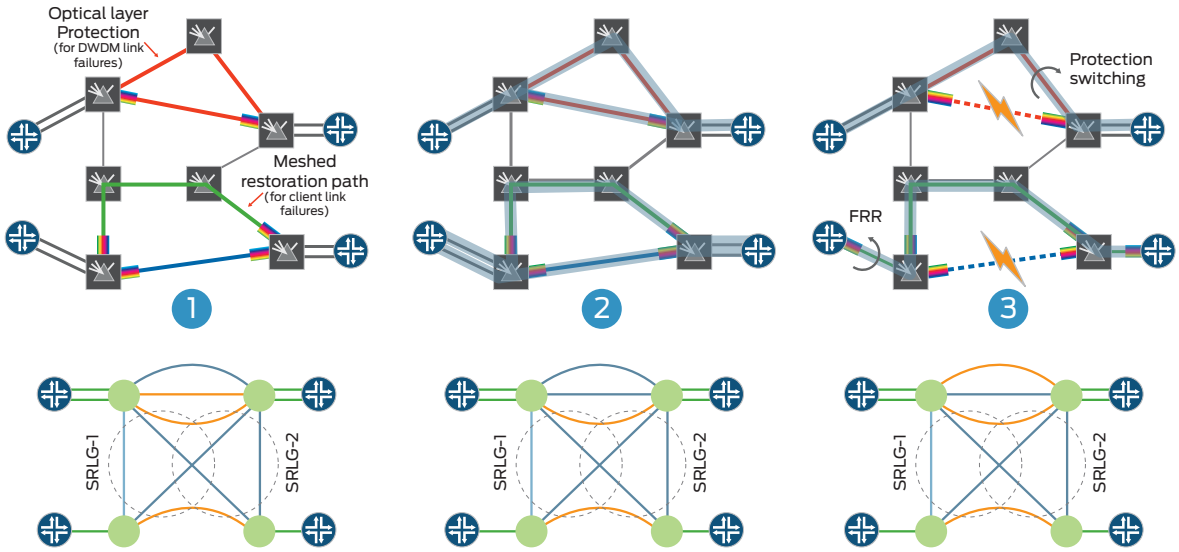

Figure 9: Exchange of protection information between transport layer and IP/MPLS network topology

The 1+1 path protection on the transport layer only requires a single DWDM port. However, transport-layer protection does not provide any resilience against client port failures on either the router or transport equipment, and therefore requires additional (meshed) restoration on the IP/MPLS layer. Restoration on the IP/MPLS layer requires additional DWDM ports. However, shared meshed protection is typically very efficient and will only result in a limited increase in the total installed capacity, as the additional capacity is shared between many restoration paths. NorthStar Controller has true end-to-end visibility of the complete IP/MPLS-layer topology and can therefore optimize the network capacity reserved for restoration purposes. The transport SDN controller, on the other hand, is limited to optimizing restoration in its own transport domain. This will result in a less optimum choice of restoration paths prompting an increase in the required network capacity for restoration, particularly in multivendor and multi-technology networks.

The use of transport-layer resilience mechanisms, in addition to restoration on the IP/MPLS layer, is therefore mainly advantageous when the fiber outage probability is much higher than the router equipment outage probability, and the network needs to be designed for more than one simultaneous fiber break.

Multilayer Maintenance Coordination

Any network layer periodically needs to schedule maintenance windows, for example for hardware or software upgrades or replacement, and when this happens, part of the network resources need to be taken temporarily offline. The scheduling of maintenance windows often involves considerable organizational effort as different parts of the organization need to be aware of any upcoming service disruptions and prepare for the impact the maintenance has on their platforms and services. The automation and simplification of maintenance windows can therefore provide significant OpEx efficiencies and savings.

With multilayer maintenance coordination, the transport SDN controller communicates with the NorthStar Controller upfront when maintenance work is scheduled on the transport layer. This is achieved by complementing the abstract topology exchange with timestamp information, which identifies the time frame a particular abstract link will be available or unavailable. This can be used to communicate the future availability of new network resources as well as the upcoming unavailability of existing resources. Using the time stamp information, NorthStar Controller can automatically identify the affected resources on the IP/MPLS layer and gracefully reroute traffic to ensure that no traffic is affected once the maintenance window commences. This ensures that there is no disruption of services, and all IP/MPLS traffic is rerouted to the new optimal path (taking into consideration the resources that are not available during the maintenance window). This also minimizes the need to deploy additional spare network capacity that is only used during maintenance windows, and can therefore provide both a significant OpEx and CapEx saving.

Figure 10 illustrates the exchange of abstract topology information during multilayer maintenance coordination. In steady state, both red and green optical circuits are used to provide connectivity for a diverse pair of LSPs. When a maintenance window is scheduled for transport node "R," the transport controller pre-announces the unavailability of the red optical circuit by adding a time stamp to the relevant abstract link. Once the maintenance window commences, NorthStar Controller can gracefully reroute the affected LSP to the new optimum path. Subsequently, optical restoration can reroute the red optical circuit to a different path that is not affected by the unavailability of transport node "R" and update the abstract topology information accordingly. NorthStar Controller can then re-signal the LSP to make use of the now optimum path.

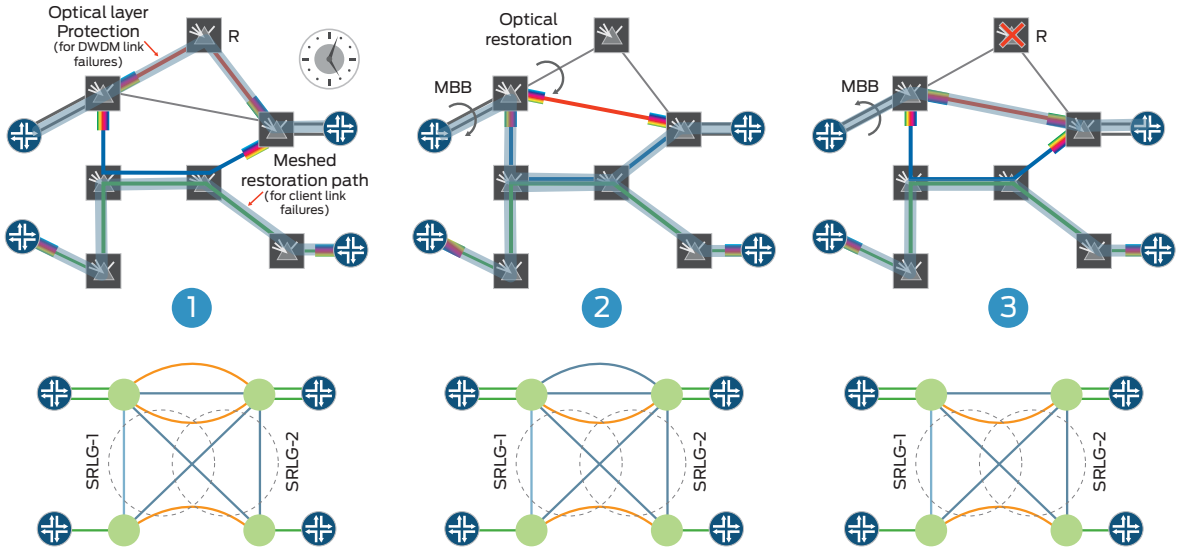

Figure 10: Coordinated maintenance between transport and IP/MPLS layers through the exchange of abstract link with time stamps (1) and actual network topology changes (3)

Conclusion

The NorthStar Controller implements a flexible approach for multilayer network optimization, based on a controller-to-controller interface towards a transport SDN controller. The exchange of an abstract topology YANG model through a RESTCONF or REST interface with the transport SDN controller is an open standards-based approach that relies on programmatic interfaces and lightweight data models that are straightforward to implement across IP/MPLS and transport-layer SDN controllers. By learning the transport-layer connectivity and metrics, the NorthStar Controller is capable of more optimum traffic engineering on the IP/MPLS layer of the network. This allows for highly relevant use cases that enable the building of more optimized and simpler multilayer network architectures such as multilayer path diversity, visibility on the IP/MPLS layer to transport layer restoration events, and any protection schemes in use, as well as coordinated multilayer maintenance.

This approach maintains the operational boundaries that are customary in a transport and IP/MPLS network. The topology information and metrics of the transport layer that are relevant for TE on the IP/MPLS layer are exchanged between both network layers, while any information and configuration of the transport layer that is irrelevant for the IP/MPLS layer is omitted from the abstract topology model. This ensures that best-in-class technologies used in each of the network layers can continue to evolve in a multivendor, multi-technology environment with their own approach for network control and management.

References

[1] https://datatracker.ietf.org/doc/rfc4208/ GMPLS-UNI

[2] https://datatracker.ietf.org/doc/draft-beeram-ccamp-gmpls-enni/ GMPLS-ENNI

[3] https://datatracker.ietf.org/doc/rfc4655/ PCE architecture

[4] https://datatracker.ietf.org/doc/rfc5440/ PCEP

[5] https://datatracker.ietf.org/doc/draft-ietf-pce-stateful-pce/ Stateful PCE

[6] http://www.juniper.net/us/en/products-services/sdn/northstar-network-controller/ NorthStar public website

[7] https://www.juniper.net/assets/us/en/local/pdf/datasheets/1000494-en.pdf NorthStar data sheet

[8] https://datatracker.ietf.org/doc/draft-ietf-idr-ls-distribution/ BGP-LS

[9] http://www.juniper.net/documentation/en_US/junos15.1/topics/example/srlg-configuring.html SRLG in IGP

[10] https://datatracker.ietf.org/doc/rfc6020/ YANG

[11] https://datatracker.ietf.org/doc/rfc6241/ NETCONF

[12] https://datatracker.ietf.org/doc/draft-ietf-teas-yang-te-topo/ Abstract topology YANG model

[13] https://datatracker.ietf.org/doc/draft-ietf-netconf-restconf/ RESTCONF

[14] https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm REST

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.
1133 Innovation Way
Sunnyvale, CA 94089 USA
Phone: 888.JUNIPER (888.586.4737)
or +1.408.745.2000
Fax: +1.408.745.2100
www.juniper.net

APAC and EMEA Headquarters
Juniper Networks International B.V.
Boeing Avenue 240
1119 PZ Schiphol-Rijk
Amsterdam, The Netherlands
Phone: +31.0.207.125.700
Fax: +31.0.207.125.701

2000615-001-EN Oct 2015