NOKIA

NSD and NRC

Release 17.3

Installation and User Guide

3HE-12072-AAAA-TQZZA

Issue 1

March 2017

Contents

List of tables

About this document

Purpose

The *NSP NSD and NRC Installation and User Guide* serves as an introduction to the NSD and NRC modules of the NSP, and provides detailed information regarding their operation.

Safety information

For your safety, this document contains safety statements. Safety statements are given at points where risks of damage to personnel, equipment, and operation may exist. Failure to follow the directions in a safety statement may result in serious consequences.

Document support

Customer documentation and product support URLs:

- Customer Documentation Welcome Page
- Technical support

How to comment

Documentation feedback

- Documentation Feedback

Part I: Getting started

Overview

Purpose

This volume serves as an introduction to the NSD and NRC modules of the NSP, and explains their function within the broader solution.

Contents

1 Safety information

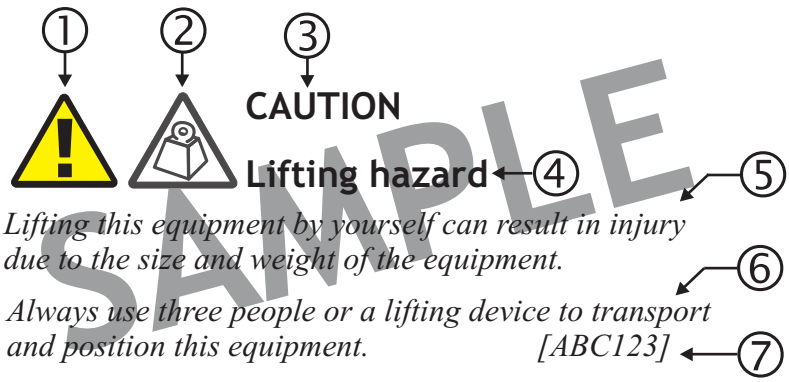
1.1 Structure of safety statements

1.1.1 Overview

This topic describes the components of safety statements that appear in this document.

1.1.2 General structure

Safety statements include the following structural elements:



Item	Structure element	Purpose
1	Safety alert symbol	Indicates the potential for personal injury (optional)
2	Safety symbol	Indicates hazard type (optional)
3	Signal word	Indicates the severity of the hazard
4	Hazard type	Describes the source of the risk of damage or injury
5	Safety message	Consequences if protective measures fail
6	Avoidance message	Protective measures to take to avoid the hazard
7	Identifier	The reference ID of the safety statement (optional)

1.1.3 Signal words

The signal words identify the hazard severity levels as follows:

Signal word	Meaning
DANGER	Indicates an extremely hazardous situation which, if not avoided, will result in death or serious injury.
WARNING	Indicates a hazardous situation which, if not avoided, could result in death or serious injury.
CAUTION	Indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.
NOTICE	Indicates a hazardous situation not related to personal injury.

2 What's new?

2.1 What's new in NSP Release 17

2.1.1 Introduction

This chapter highlights new features for NSP Release 17.3 and provides pointers into the documentation for more information. The *NSP NSD and NRC Release Description* provides Committed feature lists for all of Release 17.

2.1.2 Maintenance releases

Some maintenance releases may not be listed in this section, either because no new features are introduced or because the introduced features do not require documentation.

2.1.3 What's new in NSP Release 17.3

The table below lists the features added in NSP Release 17 and described in NSD and NRC customer documentation.

Key	Summary	See
NRC-F features		
NSP-1275	VIP-based flow steering	4.14 “To steer flows to next hops for VIP customers” (p. 52)
NRC-P features		
NSP-1112	Support for PCE-initiated LSPs	“PCE-initiated LSPs” (p. 30) 4.7 “To create PCE-initiated LSPs” (p. 47)
NSP-1116	Support for multiple integrated domains path computation	3.3.8 “Multi-domain path computation” (p. 32)
NSP-2049	Multi-instance topology support for OSPF and ISIS	8.1.13 “Router ID Mapping templates” (p. 128)
NSP-4230	Override profile routing for delegated LSPs	4.9 “To configure override path profiles for LSPs” (p. 49)
NRC-T features		

Key	Summary	See
NSP-707	1830 PSS 9.1 support: Flex Grid-related changes	7.9 “To provision OCh services” (p. 119)
NSP-2327	Explicit routing	7.9 “To provision OCh services” (p. 119)
NSP-3002	D5X500 support	7.9 “To provision OCh services” (p. 119) 7.8 “To provision ODU services” (p. 118)
NSP-3264	Support for Y-cable protected configurations	7.8 “To provision ODU services” (p. 118)
NSP-4203	Support for 200G line mode of 260SCX2	*NSP NSD and NRC Release Description*
NSP-4704	OPSB with 11QPEN4/11QPA4/ OTU4 mode 260SCX2	*NSP NSD and NRC Release Description*
NSD features		
NSP-1249	CPIPE support for NSD	7.5 “To provision C-Line services” (p. 110) 8.4 “To provision C-Line service templates” (p. 131)
NSP-2170	ELINE spans multi-domain with VLAN hand-off and MS-PW	“Multi-domain E-Line service provisioning” (p. 98)
NSP-2243	Cross-launch from NSD to Service Supervision Web App	7.11 “To modify services using the Service Fulfillment application” (p. 122)
NSP-2646	Support of IP link latency during service provisioning	8.8 “To provision L3 VPN service templates” (p. 135)
NSP-2725	Support of PCC-initiated LSPs	“PCC-initiated LSPs” (p. 127) 8.11 “To provision Tunnel Selection policies” (p. 139)
NSP-2739	Enhance the NSD scale for resync and discovery by making it multi-threaded	*NSD and NRC Release Description*
NSP-3268	Enhancements to brownfield LSP and SDP Support	7.12 “To manage service tunnel bandwidth” (p. 123)

3 NSD and NRC modules

3.1 Overview

3.1.1 Introduction

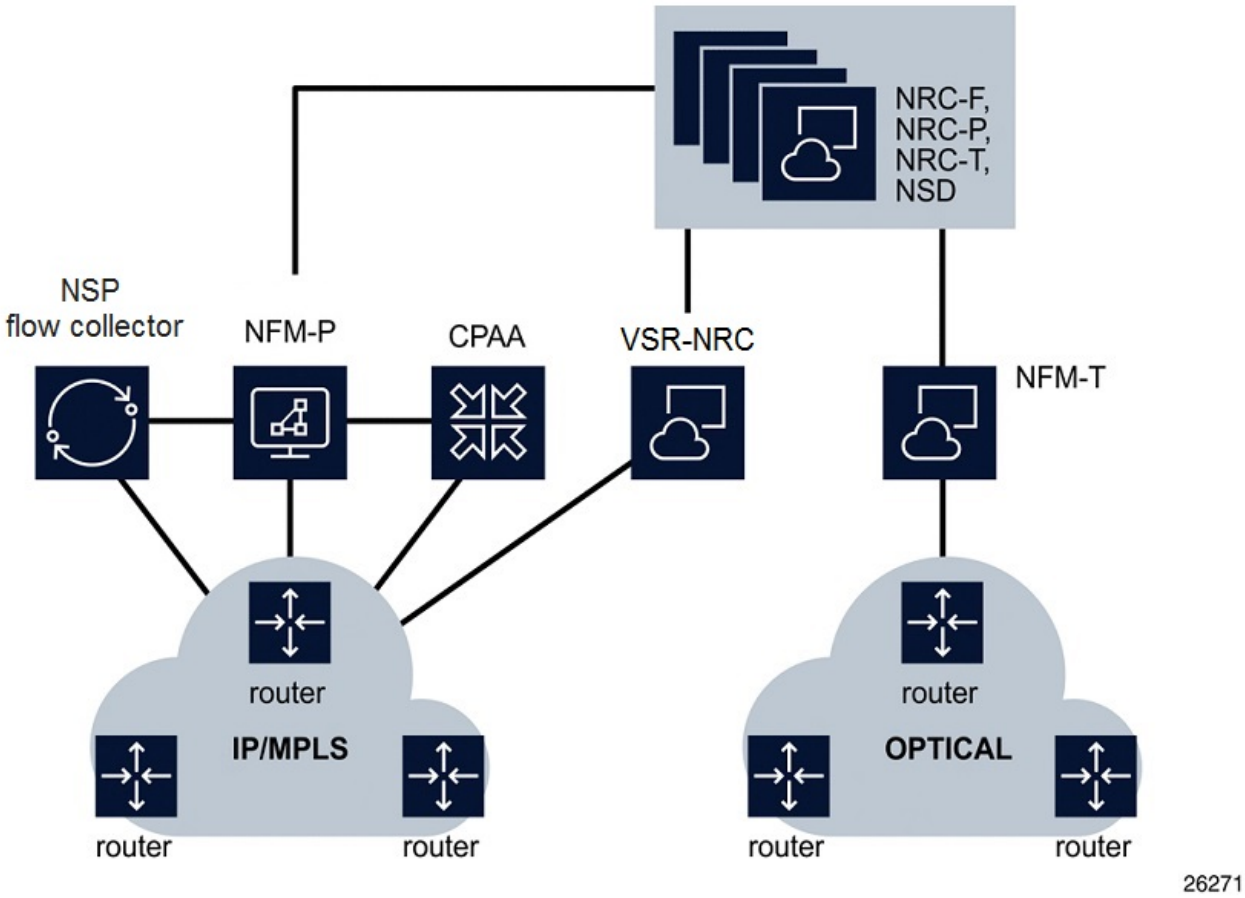
The NSD and NRC modules of the NSP form a carrier software-defined networking (SDN) platform that unifies service automation with network optimization, allowing network operators to deliver on-demand network services cost-effectively and with scalability. Using these modules, operator can define, provision, and activate network services across networks that span multiple layers (Layer 0 to Layer 3), services, and infrastructures (physical and virtual), as well as equipment from multiple vendors. The NSD and NRC modules are scalable, and based on standard protocols with multiple APIs.

The NSD and NRC modules act as bridge between the IT and network worlds. Upstream, they provide standard APIs, object models, and abstractions to IT OSS applications. Downstream, they manage network complexity by translating simple service requests into commands that program physical and virtual network elements. This is done automatically, across both IP and optical boundaries, and across multiple network vendors.

The following NSP and NRC modules are available for deployment as part of NSP Release 17.3:

- **Network Resource Controller — Flow** — flow-based traffic steering
- **Network Resource Controller — Packet (NRC-P)** — MPLS path computation
- **Network Resource Controller — Transport (NRC-T)** — Optical path computation
- **Network Services Director (NSD)** — multi-vendor service fulfillment

The following figure shows the required system architecture for an NSP deployment that includes all NSD and NRC modules:



Service automation

With the Network Services Director (NSD), OSS and IT applications are able to quickly communicate their service requests using simplified, abstract APIs and object models. Operators can also use policies to abstract their service placement intentions. For instance, a policy can be used to map a service request to the network path with the lowest latency, a specific amount of available bandwidth, and the least amount of congestion. If such a path is not available, the policy can dictate mapping to a secondary path and switch to a suitable path when the desired conditions have been met. If no suitable path is available, the policy can communicate this, or request a new path to be computed by passing the associated parameters to the network optimization modules. To complete service provisioning, the NSP handles the complex task of provisioning the

operator's multi-domain, multi-vendor network using SDN standards such as NetConf /Yang, OpenFlow, PCE-P, and other protocols. In the case of legacy equipment, traditional mechanisms such as CLI are used. This functionality is available from the Service Fulfillment application.

For more information, see 3.5 “NSD” (p. 34).

Network optimization

The Network Resource Controller (NRC) modules centralize path computation and network optimization in order to leverage a whole network view and make the best possible decision for each request. For IP, this is done with a packet PCE. For Optical, this is done with a transport PCE. For hybrid IP/Optical network, this is done with hierarchical PCEs. The latter for the simultaneous provisioning of services across IP and Optical networks. Centralized path computation elements are opened up to application and policy control, and to specialized algorithms. For instance, path computation can be enhanced to take link congestion into account. You can also make better use of your network assets and keep SLAs high by using KPIs and metrics to trigger optimization policies. You can also do all this in a multi-tenant way where each tenant – or in essence, each business unit - has their own abstracted view of the network and their own policies for maintaining service quality and assurance.

For more information, see 3.2 “NRC-F” (p. 20), 3.3 “NRC-P” (p. 24), or 3.4 “NRC-T” (p. 32).

External applications notifications

The NSD and NRC modules provide a base platform for asynchronous event notifications to external applications, such as orchestrators. These notifications are transported using HTTP Server Side Events (SSE) according to the IETF RESTCONF protocol specification. Notifications are defined in the YANG modeling language and encoded in JSON format. This base platform is used by the modules to realize different types of notifications.

Clients of the modules' northbound interface receive notifications whenever the state of a managed object changes. This simplifies synchronization with the modules, as periodic polling of the REST API is avoided. Notifications are provided for the operational and administrative status of services and endpoints.

Note: By default, a maximum of 10 users can be subscribed to these notifications. This amount can be modified from the configuration file in the directory where the NSD and NRC installer was extracted. See 5.3 “To install or upgrade a standalone NSD and NRC system” (p. 63) for more information.

3.1.2 Applications

The NSD and NRC modules also provide functionality using browser-based applications. Each of these applications use the standard NSD and NRC REST security mechanisms for authentication and authorization, so every request sent to the server contains the provided session key. All applications are HTML5–based, and supported on the latest version of Google Chrome. Use the following URL to access the NSP dashboard, from which you can launch all supported applications:

https://<*server*>:8543

Where *server* is the hostname or IP address of your installed NSD and NRC server.

For more information about the individual applications, see Chapter 4, “Applications overview”.

3.1.3 NSD and NRC REST APIs

The NSD and NRC modules provide northbound RESTful APIs that expose a simplified view of the network. This view is constructed from the internal model, which is stored in the Topology Database. The APIs support queries, service creation requests, and many additional functions.

To view and interact with the APIs online, go to one of the following URLs:

- https://<*server*>:8543/sdn/doc
- https://<*server*>:8543/task-scheduler/doc

Where *server* is the hostname or IP address of your installed NSD and NRC server.

Offline representations of these REST APIs are available alongside the NSD and NRC modules' user documentation suite.

3.1.4 Additional components

The NSD and NRC modules rely on the following additional components to provide end-to-end functionality:

- *Topology Database* — The Topology Database contains a representation of the network in the form of a highly abstract, multi-layer graph. The graph is stored in a Neo4j database.
- *Network Mediation* — The Network Mediation component is responsible for populating the Topology Database with the network information and for deployment of network configuration. It is comprised of the generic plugin framework, as well as the mediation plugins that operate inside these. Plugins may interact with the network through Element Manager Systems (EMS) such as the NFM-P, and/or standard

communication protocols such as PCEP, BGP-LS, or OpenFlow. The NSD and NRC modules support the deployment of network tunnels, services, and, potentially, tunnels.

- *Service Connection Manager* — The Service Connection Manager is responsible for finding appropriate tunnels for services.
- *Algorithm Framework* — The Algorithm Framework is the component that provides a run time environment for the invocation and execution of both routing and optimization algorithms.
- *Network Deployment* — The NSD and NRC modules support the deployment of network tunnels, services, and, potentially, tunnels. This means that some plugins and mediation framework may support the “push to the network” function that involves the mapping and conversion of the Topology Database entities to the network objects.
- *Security* — Security is the component that handles sign-in, encryption, logging of operator actions, and network events.
- *Relational Database* — A PostgreSQL database that contains all non-topological information requiring persistence. This includes policies, templates, etc.
- *Global Cache* — The Global Cache enables the NSD and NRC modules to track resources being used by the network, including the resources of services that originate from the NFM-P. In order for the NSD to discover such services, they must have their “NSD-managed” flag enabled within the NFM-P. Once this is done, the usage of VLAN IDs, L3 VPN Route Distinguishers (RD), and L3 VPN Route Targets (RT) can be tracked across NFM-P/NSD managed networks. When the NSD requests one of these resources, the Global Cache verifies their availability before assignment. Only freed resources are considered available for usage. All services created using the NSD will be validated for resource usage, and therefore will not infringe upon the resources of existing services.

3.1.5 Security

SSL provides encryption on the following interfaces:

- The northbound REST interface that accepts requests from the GUI client and OSS systems
- The internal communication channels from the SDN application to the Policy Server and Openstack Remote application (Keystone intermediate)
- The communication between Neo4J instances in an redundant deployment (used to execute remote transactions)
- The southbound interface to the NFM-P (only if NFM-P has SSL enabled)

SSL on all northbound and internal interfaces is enabled by default and no additional configuration is required, as the installer will automatically generate keystores to be used on those interfaces. Keystores generated automatically at installation contain a generated, self-signed certificate shared by all NSD and NRC instances. Custom keystores can also be pre-generated by the user and provided to the installer. These can contain either a self-signed certificate, or a security certificate signed by a certificate authority (CA).

Note: If a pre-generated keystore containing a self-signed certificate is used, the user will only have to manually accept the certificate when they first launch the web GUI and connect to the server. If a pre-generated keystore is *not* provided to the installer, then the certificate must be manually accepted the first time that each server becomes master of the cluster.

For information about retroactively enabling SSL, see 5.14 "To retroactively enable SSL communication to the NFM-P" (p. 85).

For information about generating keystores, see 5.12 "To generate a keystore" (p. 83).

3.2 NRC-F

3.2.1 Introduction

The Network Resource Controller – Flow (NRC-F) is the NSP module responsible for implementing SDN-based, traffic-steering-related protocols and applications. On the southbound side, the NRC-F uses flow-based protocols such as OpenFlow and BGP FlowSpec, and routing protocols such as BGP for route injection.

The NRC-F supports OpenFlow protocol Specification version 1.3.1 on both the controller and on Nokia 7750, 7450, and 7950 routers. The NRC-F is able to discover previously-configured OpenFlow switches on these routers, and can be used to add OpenFlow rules to their flow tables, or to delete flows from these tables altogether. Nokia VSR-NRC OpenFlow Experimenters are supported, and can redirect traffic to alternate next hops using plugins.

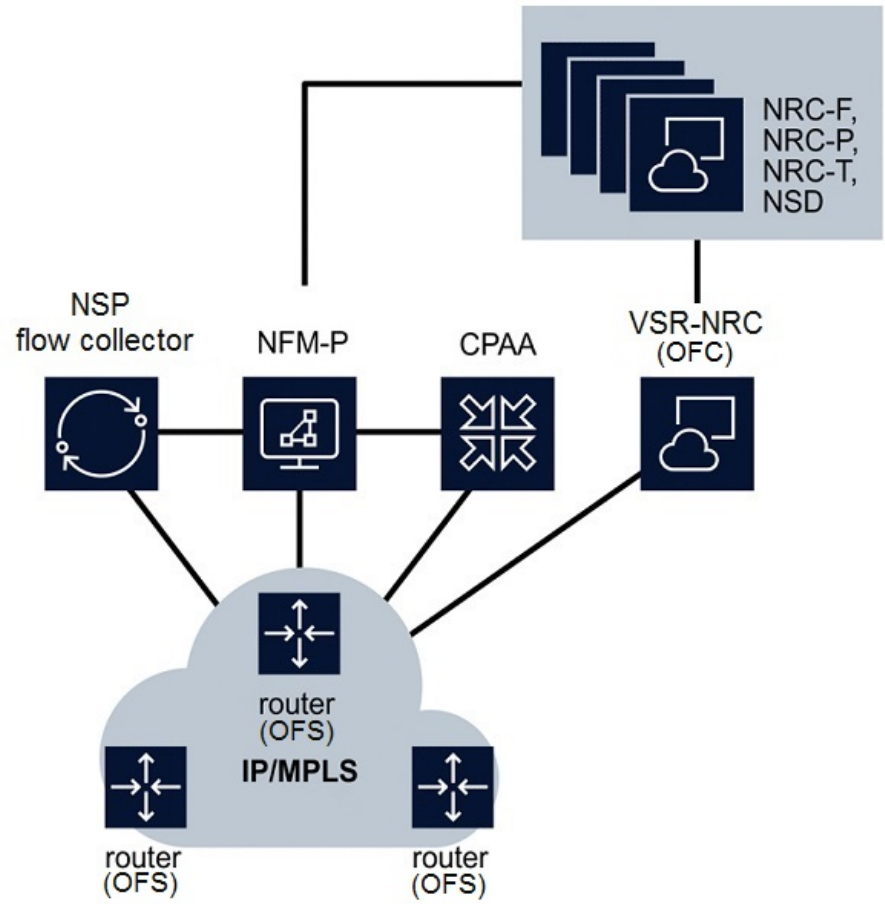
The NRC-F supports two applications:

- **Traffic Steering Controller** — The Traffic Steering Controller application provides explicit traffic manipulation to the granularity of a flow using flow-based steering protocols such as OpenFlow.
- **Autonomous System Optimizer** — The Autonomous System Optimizer application enables real-time viewing and steering of traffic flows per destination AS. This allows the user to be more responsive to congestion or high link utilization situations. The application monitors IPv4 traffic distribution on a set of router uplinks in a typical DC-WAN configuration and manually steers traffic, per destination AS, to alternate paths.

For information about accessing NRC-F functionality through these applications, see 4.12 "Autonomous System Optimizer" (p. 51) and 4.15 "Traffic Steering Controller" (p. 53).

For information about accessing NRC-F functionality through the NSP's REST API see the *NSP API Programmer Guide*.

3.2.2 NRC-F architecture



NFM-P

Routers monitored by the NRC-F are discovered from NFM-P network topology. In order for the NRC-F to receive information about the ports of these monitored routers, a TCA policy must be configured on the NFM-P. Each monitored port must be added to this policy. TCA Rules must also be created, with thresholds that represent the NSP's utilization bands: 20%, 40%, 60%, and 80%. These threshold values should include both rising and falling thresholds. An initial threshold can be set at 1% to allow for port utilization to be observed on low usage ports.

In order for the NRC-F to receive TCA notifications and real-time statistics, the following text must be added to the *opt/nsp/nfmp/server/nms/config/nms-server.xml* file:

```
<registry

 enabled=true

 zkConnectionString="<zookeeper address>

/>
```

Where <*zookeeper address*> is the IP address of the machine where the NSP's instance of Zookeeper is installed. In the case of redundant NSP deployments, two IP addresses separated by a semicolon must be provided.

Note: An NFM-P server restart is required in order for this configuration to take effect.

See the *NSP NFM-P User Guide* for more information.

See 5.8 "To enable TCAs for NRC-F" (p. 75) for more information about TCAs.

NSP flow controller

The NSP flow controller aggregates and relays statistics to the NFM-P. In order for the NRC-F to receive statistics from its monitored routers, the NSP flow controller must be configured to communicate with the NFM-P, and the ports of each router monitored by the NRC-F must have Cflowd collection enabled.

Once the NSP flow controller has been installed, the following configurations must be performed from the server CLI-based samconfig utility:

```
samconfig -m flow

<flow> configure category sys

<flow configure> back

<flow> apply

<flow> exit_all
```

Filters should also be configured to show monitored routers.

See the *NSP NFM-P Installation and Upgrade Guide* for more information.

CPAA

The CPAA is used to retrieve BGP prefixes for Autonomous Systems (AS) monitored by the NRC-F. In order for the NRC-F to monitor these ASs, the CPAA must be integrated with the NFM-P that is relaying network information to the NRC-F. This CPAA must also be configured with a BGP administrative domain.

In order for the NRC-F to monitor the ASs discovered by the CPAA, the BGP section of the */opt/nsp/configure/config/nrcf.conf* file must be populated as follows:

```
bgp {

# BGP Autonomous system number of CPAA router

cpaa_autonomous_system_number = <CPAA AS number>

# BGP prefix filter id used for fetching prefixes

prefix_filter_id = 65535

# BGP prefix fetch timeout (milliseconds)

prefix_fetch_timeout = 60000

# BGP As subnet info refresh timer  (hours)

as_subnet_refresh_timer = 24

}
```

Where *CPAA AS number* is the AS number of the CPAA.

Note: The retrieval of BGP AS subnets is based on the local AS of the BGP route, as seen by the CPAA. When retrieving AS subnets, the NSP will modify the specified BGP prefix filter list for the requested local AS. BGP ASes and CPAA ASes must match. See the *NSP CPAM User Guide* for more information.

VSR-NRC

In an NRC-F deployment, the VSR-NRC serves as an OpenFlow Controller (OFC). The OFC is used to push flows to the OpenFlow Switches (OFSes) under its control, as directed by the NRC-F.

Note: In order for the NRC-F to communicate with the OpenFlow Controller, the *openflow* parameter in the */opt/nsp/configure/config/sros-vms.conf* file must be set to true.

Note: The OpenFlow Controller CLI tree is visible on hardware, but cannot be used for NRC-F OpenFlow functions.

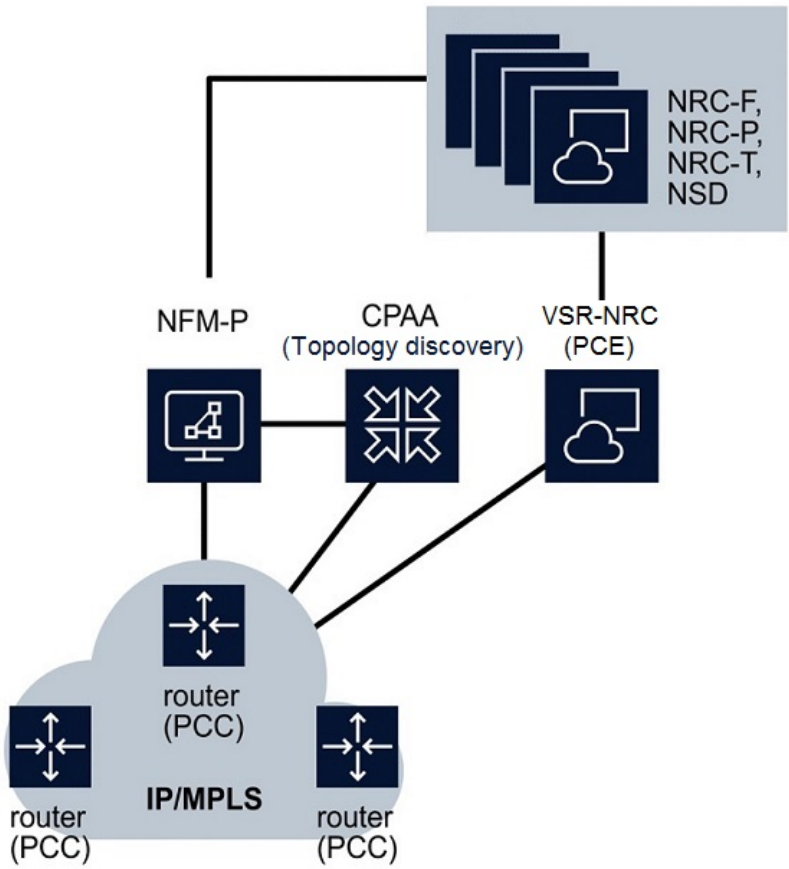
3.3 NRC-P

3.3.1 Introduction

The Network Resource Controller - Packet (NRC-P) leverages centralized, intelligent network control capabilities so that operators can rapidly adapt to changing demand and traffic patterns and run their networks more efficiently. The NRC-P accepts path connection requests from the NSD, from OSS and orchestration systems, and from physical/virtual network elements. The NRC-P calculates optimal paths through the network for a given set of business and technical constraints by leveraging centralized views of all available assets/topologies and their current state.

The NRC-P module is based on a Path Computation Element (PCE) architecture that integrates standard protocols such as PCEP to open up path computation to external control. This allows PCEs to be enhanced with various path optimization algorithms that ensure optimal path placement across the network. The NRC-P is stateful in nature and will maintain an up-to-date Traffic Engineering Database (TED), as well as the current RSVP based Label switched paths (LSP) and the segment routing path (SRP) state. It tracks RSVP BW and manages BW for the Segment-Routed TE paths.

3.3.2 NRC-P architecture (CPAA topology discovery)



26273

NFM-P

In order for the NRC-P to use a CPAA for IGP link-state topology discovery, an NFM-P with an integrated CPAA must be deployed alongside the NRC-P.

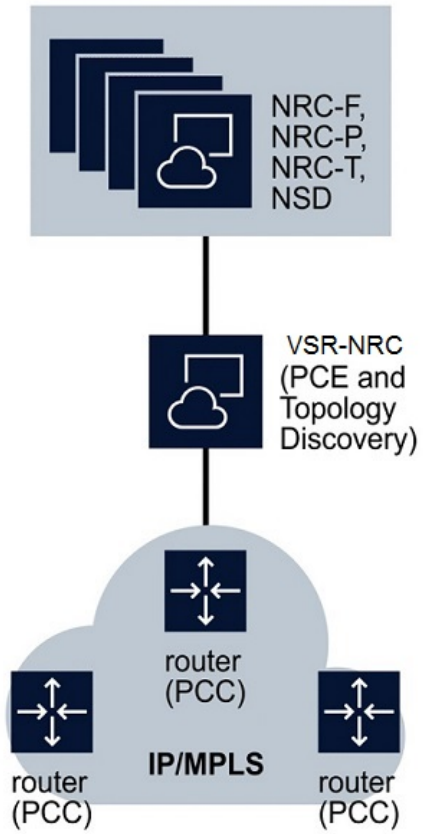
CPAA

The NRC-P can use a CPAA to discover IGP link-state topologies. This is accomplished by peering the CPAA with IGP network elements that have full visibility of the topology.

VSR-NRC

The VSR-NRC terminates PCEP connections and conveys path request messages from PCCs to the NRC-P. The NRC-P computes the requested path and responds to the VSR-NRC, which conveys the response to the PCCs. The communication between VSR-NRC and PCCs is accomplished using the PCEP protocol.

3.3.3 NRC-P architecture (VSR-NRC topology discovery)



26274

NFM-P

In order for the NRC-P to use a CPAA for IGP link-state topology discovery, an NFM-P with an integrated CPAA must be deployed alongside the NRC-P.

CPAA

The NRC-P can use a CPAA to discover IGP link-state topologies. This is accomplished by peering the CPAA with IGP network elements that have full visibility of the topology.

VSR-NRC

The VSR-NRC terminates PCEP connections and conveys path request messages from PCCs to the NRC-P. The NRC-P computes the requested path and responds to the VSR-NRC, which conveys the response to the PCCs. The communication between VSR-NRC and PCCs is accomplished using the PCEP protocol.

In order for the VSR-NRC to discover the IGP topology, it must be peered with IGP routers. It can then discover IGP link-state topologies using either IGP or BGP-LS. If using IGP, the VSR-NRC must have full visibility of the topology. For multi-area topologies, this means that the VSR-NRC must be connected to every area, or to the ABRs/(L1/L2s) via IGP (OSPF or ISIS) adjacencies. If using BGP-LS, the VSR-NRC must be peered with a BGP speaker, ABRs/(L1/L2s) that are BGP speakers, or a Router Reflector that is peered to a BGP speaker in each IGP area. In order for BGP-LS discovery to be successful, each BGP speaker must support BGP-LS.

3.3.4 RSVP LSPs

Any PCC node intending to request a path computation from the NRC-P must first set the PCE computation option in the LSP definition. The PCC then assigns a unique PLSP-ID to the LSP. This uniquely identifies the LSP within a PCEP session and is maintained for the lifetime of the LSP. The PLSP-ID is also associated to the tunnel and path ID.

Once the PLSP-ID is assigned, the PCC sends a PCReq message to the NRC-P PCE, requesting a path for the LSP. This request includes the LSP parameters in the METRIC object, the LSPA object, and the Bandwidth object. It also includes the LSP object with the selected PLSP-ID. The NRC-P is now able to compute a new path, check the bandwidth, and return the path in a PCRep message with the computed ERO in the ERO object. It also includes the LSP object with the unique PLSP-ID, the METRIC object with the computed metric value (if any), and the Bandwidth object.

The NRC-P will not keep track of the LSP yet. At this point, it has simply returned the ERO. The PCC has yet to confirm that the path was signaled. If the path was locally signaled, and the local TEDB has been updated, the NRC-P will receive the updates via BGP-LS and update its TEDB.

For stateful operation, which allows the NRC-P to track the LSP path and bandwidth (among other constraints), the PCE report option must be set in the LSP definition. When this option is set, the PCC sends both a PCRpt message to update the NRC-P with the state of UP, and the RRO object as confirmation. The RRO object now includes the LSP object with the unique PLSP-ID. With this, the NRC-P is able to display the LSP,

as well as its hops and constraints. The RRO also contains information about the protection that is enabled on the signaled path. Therefore, the NRC-P is aware of the protection at the hops, but not aware of the detour/bypass tunnel details. If a local failure causes the LSP on the PCC to switch to a detour or bypass, a PCE report is sent to the NRC-P, and the NRC-P becomes aware that the LSP is using a detour or bypass.

Note: In the VSR-NRC, the PCE reporting option can either be set globally, or on a per LSP basis.

The PCC can also delegate control of the LSP to the NRC-P for either active control or LSP optimization. This is known as active stateful behavior. The delegation is awarded using the PCE control option. Once the NRC-P is controlling the LSP, the operator can manually re-signal/re-optimize the LSP. Re-signalling routes the LSP using its original constraints, while re-optimizing routes the LSP using an optimization algorithm. The NRC-P also re-routes LSPs automatically on resource failures, or when calculating disjoint paths.

Note: When the PCC has delegated control of the LSP to the NRC-P, any change to the LSP definition (such as changes in constraints), requires the PCC to first revoke the delegation via the PCE report option, and then issue a new request to the NRC-P.

Secondary path behavior

The PCC sends PCE requests for standby secondary paths. A new PLSP-ID is used for these paths over the PCEP session, and is associated to the LSP path ID and the LSP tunnel ID. When a secondary path is not in standby, the PCE request is not sent until the primary path is down, or in FRR. However, if the path is delegated to the NRC-P, this will result in a PCE update from the NRC-P. The LSP may switch to the secondary path in the interim, but will switch back to the primary path as soon as possible.

The NRC-P maintains the active path in case both the primary and secondary paths are signaled, and also when the primary path is down. The NRC-P also maintains the shared explicit behavior when the primary and secondary paths share common link resources.

The NRC-P also indicates the active path between the primary and secondary pair.

FRR notification

Fast re-route (FRR) is signaled locally, with locally-created detour tunnels. These tunnels are not reported to the NRC-P, and therefore, the NRC-P is not aware of the detours and bypass. However, the types of node and/or link protection are communicated to the NRC-P via the PCE report.

RSVP LSP bandwidth management

The NRC-P manages the LSP bandwidth consumption on the TE links for both stateless and stateful PCC configurations. In a stateless configuration, the NRC-P receives TE

updates from the network as LSPs are signaled, thereby mimicking the TE DB bandwidth consumption on the nodes. This allows for accurate LSP path computation without maintaining state on the NRC-P. In a stateful case, wherein the reports are sent to the NRC-P from the PCC, the bandwidth is again communicated by the PCC to the NRC-P via the bandwidth object. Here, the NRC-P will reconcile the TE update with the specific LSP bandwidth update via the report. Therefore, the NRC-P maintains full LSP state along with the consumption on the TE links for these LSPs only.

It is possible that existing brownfield LSPs will not request paths from the NRC-P, and therefore, will have no state on the NRC-P. The NRC-P will not show these LSP reservations on the TE links. For a mixture of LSPs that are PCE-reported and non-PCE-reported, the NRC-P will track and show the actual TE consumption on a TE link in addition to the LSP reservation for PCE-reported LSPs.

3.3.5 Segment-routed TE LSPs

Any PCC node intending to request a path computation from the NRC-P must first set the PCE computation option in the LSP definition. The PCC then assigns a unique PLSP-ID to the LSP. This uniquely identifies the LSP within a PCEP session and is maintained for the lifetime of the LSP. The PLSP-ID is also associated to the tunnel and path ID.

Once the PLSP-ID is assigned, the PCC sends a PCReq message to the NRC-P PCE, requesting a path for the LSP. This request includes the LSP parameters in the SRP object, the METRIC object, the LSPA object, and the Bandwidth object. It also includes the LSP object with the selected PLSP-ID. The NRC-P will reserve bandwidth for the path to be returned, but will not keep track of the operational status or other requirements for the LSP yet. At this point, bandwidth is consumed and an ERO is returned. The PCC has yet to confirm that the path was signaled. If the path was locally signaled, and the local TEDB has been updated, the NRC-P will receive a REPORT from the PCC and the updates via BGP-LS and update its TEDB. If the PCC fails to send a report, after a period of time the bandwidth reserved will be released from the NRC-P. The path computed by the NRC-P is specified explicitly with the next hop interfaces and the adjacency SIDs encoded in the SR ERO sub-object.

When the PCE report option is set in the LSP definition, the PCC sends both a PCRpt message to update the NRC-P with the state of UP, and the RRO object as confirmation. The RRO object now includes the LSP object with the unique PLSP-ID. With this, the NRC-P is able to display the LSP, as well as its hops and constraints. The RRO also contains information about the protection that is enabled on the signaled path. Therefore, the NRC-P is aware of the protection at the hops, but not aware of the detour/bypass tunnel details. If a local failure causes the LSP on the PCC to switch to a detour or bypass, a PCE report is sent to the NRC-P, and the NRC-P becomes aware that the LSP is using a detour or bypass.

Note: In the VSR-NRC, the PCE reporting option can either be set globally, or on a per LSP basis.

The PCC can also delegate control of the LSP to the NRC-P for either active control or LSP optimization. This is known as active stateful behavior. The delegation is awarded using the PCE control option. Once the NRC-P is controlling the LSP, the operator can manually re-signal/re-optimize the LSP. Re-signalling routes the LSP using its original constraints, while re-optimizing routes the LSP using an optimization algorithm. The NRC-P also re-routes LSPs automatically on resource failures, or when calculating disjoint paths.

Note: When the PCC has delegated control of the LSP to the NRC-P, any change to the LSP definition (such as changes in constraints), requires the PCC to first revoke the delegation via the PCE report option, and then issue a new request to the NRC-P.

Bandwidth management

A bandwidth value that is specified on an LSP has no significance on the PCC/router because the SR TE does not maintain any state on the intermediate or destination routers. Therefore, no bandwidth tracking is done in the local TE DB. The bandwidth has to be tracked by the NRC-P if the LSP is configured to report bandwidth. Bandwidth tracking on the NRC-P is done only after a valid PCE report message is generated by the PCC. The NRC-P tracks the bandwidth reservation for SR TE LSPs separate from RSVP TE LSPs.

Note: A loose hop SR LSP whose bandwidth is specified and computed locally will not be tracked by the NRC-P, even with the PCE report option enabled. The NRC-P only tracks SR TE LSP paths computed by the NRC-P itself.

Failure detection

The head end router for an SR TE path, or an SR path, has no indication when a downstream link failure has impacted traffic for that SR TE or SR path. For a stateless and stateful application without PCE control, the SR TE tunnel on the head end router will remain up, as it receives no notification from the control plane either locally, or via NRC-P. For an LSP with delegated control to the NRC-P, the NRC-P will react to the topology change and issue a new ERO update to the PCC via PCE update.

PCE-initiated LSPs

The NRC-P supports the creation of PCE-initiated Segment-Routed TE LSPs. Operators can specify the LSP parameters and PCC address within an LSP creation form. Operators can also select a path profile to associate to the LSP path. See the 4.7 “To create PCE-initiated LSPs” (p. 47) procedure for more information.

3.3.6 IRO object

The NRC-P supports the IRO object specification within a PCC request. The NRC-P computes a CSPF path from the source to the IRO object, and another CSFP path from

the IRO object to the destination. If the second CSPF path visits any of the nodes in first CSPF path, the path computation fails.

When used with a path profile that contains the bidirectional disjoint specification, a forward LSP and its matching reverse LSP must share the same IRO configuration. This means that the list of addresses in the IRO path must be the same, but their order reversed. This is because the disjoint algorithm is natively bidirectional strict. If the reverse LSP contained IROs that did not exist in the forward path, no path would be found, because it would no longer be bidirectional strict.

3.3.7 Algorithms

The NRC-P uses path computation algorithms to identify optimal paths within the network.

STAR algorithm

The NRC-P provides a load-balancing and optimal-path-placement algorithm, known as the STAR algorithm. This algorithm uses an internal metric, calculated from the current value of the TE bandwidth reservation, to route the CSPF paths. Every path that is allocated on a TE link changes the internal metric for both the link and the overall path. Initially, all links have the same star weight, or metric, so the first path requests for CSPF traversal will choose the shortest path that satisfies all constraints. If there are multiple paths that satisfy the user constraints, then a path will be chosen randomly. This behavior is the same for normal CSPF.

Subsequent requests will choose paths that possess the least star weight, thereby ignoring the path that the normal CSPF algorithm would have chosen. The calculation of the star weight is based on a formula that uses the current link reservation. The user constraints are still satisfied. This balances the overall network utilization.

The STAR algorithm is invoked per LSP by associating that LSP to a path profile. The path profile template is defined in the NRC-P and requires setting the objective to use STAR WEIGHT. The path profile is specified with the LSP definition and is conveyed to the NRC-P via a PCE request message.

See 8.13 “To provision Path Profile templates” (p. 142) for more information about configuring the path profile template.

Disjoint optimal path computation algorithm

The NRC-P provides support for disjoint path computation between a source destination pair and between two pairs of sources and destinations. Applications can use this algorithm to provide no-impact redundancy for a service offering. The algorithm provides node/link and SRLG types of disjoint path computations. The algorithm can also re-optimize an existing path if a second path request asks to be disjoint from the existing path. The ability to treat a pair of paths as mutually disjoint requires associating a path

profile ID to the path request. In addition, a path group ID specification is also essential to implicitly identify the path pair from other path pairs. The disjoint optimal path calculation algorithm can also compute paths that are bidirectionally symmetric, to ensure that forward and reverse traffic use the hops while being disjoint.

Note: The NRC-P can only compute bi-directionally symmetric forward or reverse paths. For an RSVP LSP with primary and secondary path specification, the profile is applied to both paths. For example, if there are two RSVP LSPs between the respective distinct sources and destinations, the primary path of LSP 1 will be mutually disjoint from the primary path of LSP2, and vice versa for secondary paths. The algorithm cannot be applied to ensure the primary and secondary paths between the same source and destination pair are mutually disjoint.

Global concurrent optimization algorithm

The NRC-P also supports optimizing the paths of existing LSPs by applying an optimization algorithm. This algorithm extracts the current resource availability on the current topology and re-routes the selected LSP paths such that the overall network consumption is minimized. The result is to utilize more network links, but also reduce the consumption on the links. LSPs must be delegated to the NRC-P and must be pre-selected. Profiles do not have to be associated to the paths in order to use this algorithm. The LSPs to be optimized are selected manually on from the NSD application.

Note: LSPs that have a profile with the disjoint option enabled are excluded.

3.3.8 Multi-domain path computation

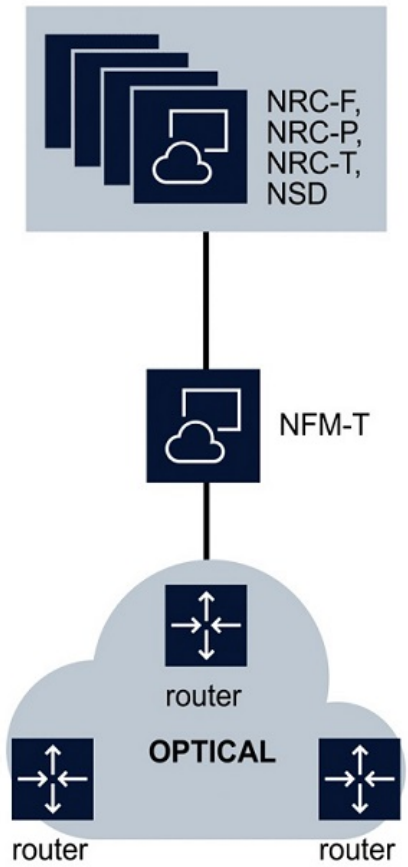
The NRC-P supports path computation across multiple IGP instances. These instances are discovered as admin domains with stitching points on the common ASBR routers. The path traversal algorithm uses a flat graph and computes the shortest path based on the required metric. Any optimization limiting domain traversals is not considered. Both Segment-Routed TE paths and RSVP TE paths are supported and deployed. Existing constraints such as the Max SID label depth apply.

3.4 NRC-T

3.4.1 Introduction

The Network Resource Controller – Transport (NRC-T) manages the creation of a transport path connection for Layer 1 Optical Transport Networks and Layer 0 Dense Wavelength Division Multiplexing (DWDM) networks. The NRC-T accepts path connection requests from the NSD, OSSes, orchestration systems, and from network elements (both physical and virtual). The NRC-T maintains an optical topology and current path database that is synchronized with the network elements to ensure that optimal paths are computed.

3.4.2 NRC-T architecture



26275

NRC-T

The NFM-T provides the NRC-T with a managed optical network model.

3.4.3 Service consistency

The NRC-T provides full consistency between services deployed by the NSD and the NFM-T. To achieve consistency in support of network troubleshooting, NSD-created services must be visible and identifiable in the NFM-T. In addition, all services (in any layer) created using the NFM-T must be correctly uploaded into the NSD.

Note: Only for unprotected services, services that are supported by the NSD, and higher-order ODU services.

3.4.4 NBI adaptation to multi-layer/multi-vendor orchestrator/controller

The NBI provides information about the reason for failure in path computation, such as infeasible, no wavelength available, or no regenerator available. Error messages are propagated through the NBI with the mentioned specifications. Enhanced attributes describing Network Elements are also provided, including: site name, geo-location, and optical node type. Ports determination is performed by adding filtering for NE ports.

3.5 NSD

3.5.1 Introduction

The Network Services Director, or NSD, is the network service fulfillment module of the NSP. It automates IP/MPLS, carrier Ethernet and optical service provisioning by mapping abstract service definitions to detailed service templates using operator-defined policies. The NSD also provides provisioning for complex multi-technology services across multi-domain networks.

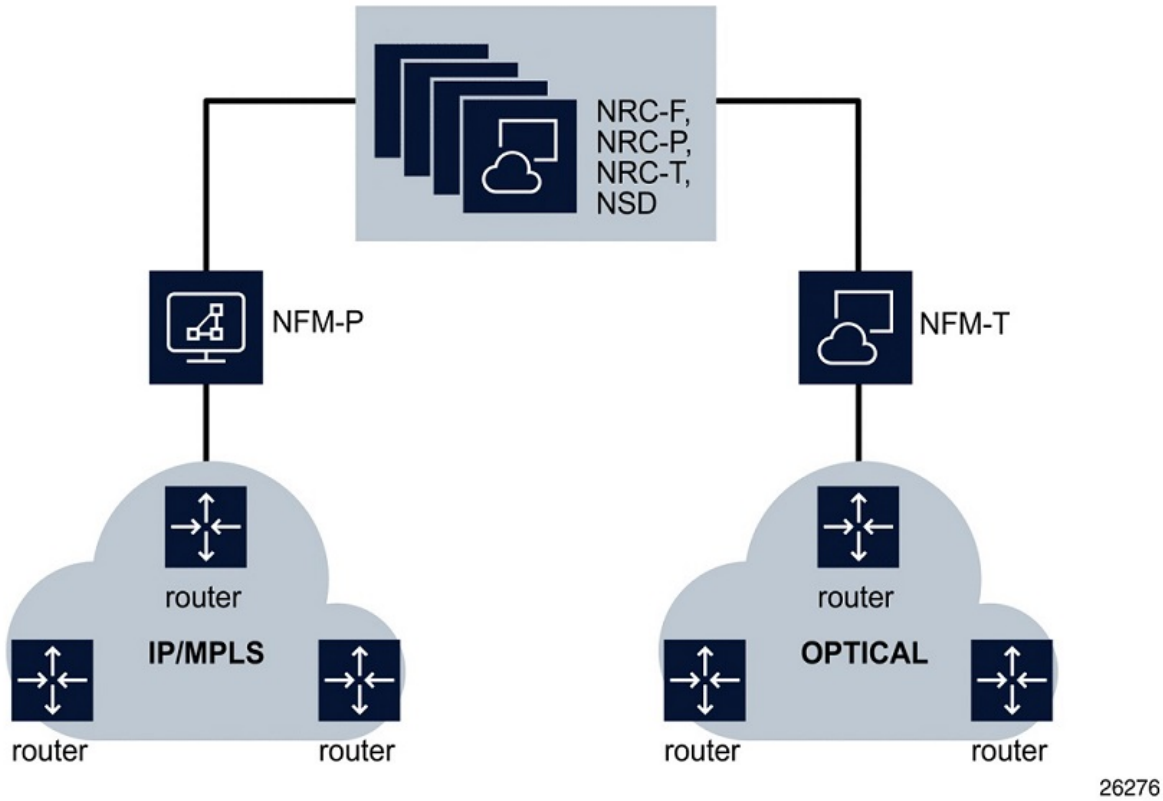
The NSD maintains abstracted service models that are based on YANG standards, and maps the models to device-specific models that are normalized for multi-vendor provisioning transparency.

The NSD provides network-aware management using a central service connection resource database to track tunnel bandwidth. As the NSD provisions a service, it performs an intelligent database search to choose the optimal path based on the required bandwidth, span, latency, cost, path diversity, and other constraints. Using the resource database and policies, the NSD directs service connection requests to tunnels or paths that have low utilization and thus averts link congestion.

An NSD operator can customize the binding of service connections to tunnels or paths using service-specific policies. If there is no service connection path that meets the specified requirements, the NSD can use a policy to request a new path from the NRC.

The NSD works with the NSP Assurance and Analytics functions for use cases such as IP/optical network-aware provisioning automation with service validation, and bandwidth-on-demand for IP/optical services with LAG resizing.

3.5.2 NSD architecture



NFM-P

When integrated with the NSD, the NFM-P provides a managed IP/MPLS network model. The NSD leverages this model to perform automated service provisioning and modification on the NFM-P's network.

NFM-T

When integrated with the NSD, the NFM-T provides a managed optical network model. The NSD leverages this model to perform automated service provisioning and modification on the NFM-T's network.

3.5.3 NSD service access QoS

This feature includes an implementation of a normalized model for access QoS: a generic QoS policy that can be used for 7450 ESS, 7750 SR, 7210 SAS, and third party routers. The enhancement in QoS also facilitates Bandwidth on Demand functionality.

This feature includes the following QoS setup and usage procedures:

1. The operator/admin defines QoS catalog including, for example, Gold, Silver, and Bronze categories. This user also uses the NFM-P GUI to define the QoS Generic Policies. The policy model is generic, therefore, it can be applied to 7450 ESS, 7750 SR, 7210 SAS, and third party routers.
2. While provisioning an endpoint, either via ReST NBI or Service Fulfillment application, the user (a tenant user or operator) can select one of the predefined QoS categories (gold, silver, or bronze).
3. If Bandwidth on Demand is used (meaning the bandwidth constraints are modified), then the user can only select another policy.

   The behavior is as follows for each of the supported node categories:
   - 7450 ESS and 7750 SR: the changes only affect the SAP that is being changed
   - 7210 SAS: queue changes are not addressed
   - Third party routers: changes are defined by the corresponding driver

3.5.4 Brownfield LSP and SDP tunnels

The NSD is capable of discovering LSP and SDP tunnels created previously within the NFM-P, including multi-vendor LSP and SDP tunnels, with the following exceptions:

- A single SDP tunnel using multiple LSPs
- Multiple SDP tunnels using the same LSP
- SR (Segment-Routed) LSPs

Service tunnels (SDP)

The NSD will discover, and will allow users to create services with bandwidth constraints on service tunnels created previously within the NFM-P. The NSD will operate with initial allocated bandwidth on these tunnels and will keep track of used bandwidth for all the services created by the NSD. It is assumed that the NSD is the only entity creating services on these tunnels. The NSD can be used to delete or resize the allocated bandwidth, as well as to modify the LSPs associated with service tunnels previously created within the NFM-P.

LDP

The NSD will discover, and will allow users to create service tunnels on RSVP-TE LSPs created previously within the NFM-P. The NSD will operate with initial allocated bandwidth on these LSPs and will keep track of used bandwidth for all the service

tunnels created on these LSPs by the NSD. It is assumed that NSD is the only entity creating service tunnels on these LSPs. The NSD cannot be used to delete, resize the allocated bandwidth, or modify LSPs previously created within the NFM-P.

Bandwidth update on existing LSP

When the reserved bandwidth of a previously-discovered LSP is modified, the NSD receives an event, and will update the both initial and available bandwidth on the LSP and SDP tunnel. This case should apply to all LSP and SDP tunnels managed by the NSD, regardless of their origin, with the following exceptions:

- A single SDP tunnel that uses multiple LSPs
- Multiple SDP tunnels that use the same LSP
- SR (Segment-Routed) LSPs

Note: When an LSP is used by an SDP tunnel, but is not yet bound to any service, that LSP's initial and available bandwidth will be updated. However, since the LSP is used by an SDP tunnel, the SDP tunnel will take the entire bandwidth. As there are no services using the SDP tunnel, the available bandwidth should be equal to current bandwidth.

Note: When an LSP is used by an SDP tunnel and there are services bound to the SDP tunnel, the SDP tunnel will take all of the LSP's current bandwidth. The LSP's available bandwidth should be 0, and depending on the services bound to the SDP tunnel, the available bandwidth of the SDP tunnel will be adjusted to reflect the current bandwidth, minus the total bandwidth of all services running on that SDP tunnel.

3.5.5 External topology visualization

A rendering of the NSD's service topology map can be embedded in any third-party web page that has sufficient access rights. The user has only read-only access to the objects in this map rendering. For example, this means that the position of the objects within the map rendering can be modified, but that these changes are not stored.

Third-party web pages embed this map rendering using an HTML 'iframe' tag. The map rendering is then retrieved using a well-defined URI, for example:

`https://<nsd address>:8543/nsd?embeddedMode=true&map=service&mapView=<view type>&serviceId=<service ID>&token=<token>&hiddenLayers=<layers>`

where

nsd address is the IP address of the NSD and NRC server

view type is the type of map view to be used: flat, stacked, or overlay

service ID is the ID of the UUID of the service to display

token is a valid token used for connecting to the NSD and NRC server

layers is a comma-separated list of numbers that represent the layers of the map to be hidden. Valid values are:

Number	Layer
1	IP
2	Physical
3	MPLS
5	OCh
6	ODU
7	SVCT
8	Service

Note: If no token is provided in the URI, the web page will display a login screen, and the user must provide a valid user name and password combination.

Note: Multi-layer topology visualization is only available to users that have the admin role assigned. If the provided token does not have these rights, only the service layer is displayed.

If a valid token is provided in the URI, but the map rendering is not ready, the web page will display an initialization screen. Once the map rendering is ready, it will post an HTML 5 message with the following format:

```
{

 isReady: <true|false>

}
```

The web page can listen for this event using JavaScript. Until this message is posted, the map rendering is not ready.

An HTML5 message can be used to dynamically change the service that is being displayed. In this 'postMessage', the JavaScript objects correspond to the field names of the URI parameters:

```
{

 commandType: "NSD_APP_COMMAND"

 map: "service"
```

```
 serviceId: <service ID>

 mapView: <view type>

 hiddenLayers: <layers>

}
```

4 Applications overview

4.1 Introduction

4.1.1 NSD and NRC applications

This chapter provides information about the following NSD and NRC applications:

- 4.2 “Service Fulfillment” (p. 41)
- 4.10 “Policy Management” (p. 50)
- 4.11 “Task Scheduler” (p. 50)
- 4.12 “Autonomous System Optimizer” (p. 51)
- 4.15 “Traffic Steering Controller” (p. 53)

All NSP applications are HTML5–based, and are supported on the latest version of Google Chrome.

Note: The NSD and NRC applications' view of the network can be affected whenever activities are drawing heavily on CPU and memory usage, such as when a large number of services are being created, modified, or deleted via the NSD and NRC REST APIs.

Localized language support

All NSD and NRC applications support localized language display. Localized language display, also known as internationalization, displays GUI text in a specified language. The localized language setting applies to most GUI objects, except system components and database objects. Contact Nokia technical support for more information about localized language support.

Note: The NSD and NRC modules support localized language settings using predefined strings, and do not translate data to different languages.

4.2 Service Fulfillment

4.2.1 Introduction

The Service Fulfillment application allows for multi-vendor service provisioning and activation across all networks accessible to the NSD. It authorizes northbound interface (NBI) service requests, executes routing algorithms that allocate network resources for these services, and then deploys the services to the network. Network deployment is performed through the mediation framework. The Service Fulfillment application can use existing tunnels or create new tunnels to satisfy service demands. The services that can

be provisioned from the Service Fulfillment application include L3 VPN, E-Line, C-Line, LAG, OCh, and ODU. For more information about these service types, see Chapter 7, “Services”.

The Service Fulfillment application also provides an abstract, real-time view of the network resources that can be consumed by services, allowing service providers and end users to interact with the network through simple APIs, and to programmatically control the network. Network abstraction is used to simplify how the network appears to the IT/OSS layer. This allows services to be defined and enhanced more quickly by presenting only the subset of network services and endpoints that are relevant to a specific application, thereby greatly reducing the complexity the application is exposed to.

After a service request has been communicated through simple RESTful APIs, or through the Service Fulfillment application, the NSD uses operator-defined policies to guide dynamic network resource selection and automated provisioning. These policies use a real-time view of the network (including link and tunnel utilization) to map service connection requests to the best available tunnels/paths (Layer 0 to Layer 3) that meet the customer’s Service Level Agreement (SLA) requirements and the operator’s network efficiency goals. For example, the NSD can track booking and use real-time network KPIs to assess whether existing tunnels/paths are congested. If so, the NSD uses operator-defined policies to bind incoming service requests to less utilized paths that provide approximately the same connection attributes. It can revert the services to the optimal paths when demand subsides. If no path that meets the requested attributes is available, the NSD asks the relevant NRC module to compute a new path.

The Service Fulfillment application can be accessed at the following URL:

https://<*server*>:8543/nsd

Where *server* is the hostname or IP address of your installed NSD and NRC server.

Note: For IP-only deployments, the NSD must integrate with NFM-P, CPAM, and v7701 CPAA. For Optical-only deployments, the NSD must integrate with NFM-T.

4.3 To customize the Service Fulfillment topology map view

4.3.1 Purpose

The following procedure can be used to customize the view of the network that is presented by the Service Fulfillment topology map.

4.3.2 Steps

1

Choose a map type from the Map drop-down menu and perform one of the following:

a. Choose the IP Area map type. Go to step Step 2.

b. Choose the IP IGP map type. Go to step Step 3.

c. Choose the Layer 2 map type. No further configuration is required.

d. Choose the Optical OCh map type. No further configuration is required.

e. Choose the Optical ODU map type. No further configuration is required.

f. Choose the Physical map type. Go to step Step 4.

g. Choose the Service map type. Go to step Step 5.

2

Choose an admin domain from the Admin Domain drop-down menu and an IP area from the IP Area drop-down menu. No further configuration is required.

3

Choose an admin domain from the Admin Domain drop-down menu and an NE view from the NE view drop-down menu. To create an NE view, perform Step 6. No further configuration is required.

4

Choose an NE view from the drop-down menu. To create an NE view, perform Step 6. No further configuration is required.

5

Click on the All Services tab and select a service from the list. No further configuration is required.

6

Click on the + button next to the NE view drop-down menu. The Create a Network Element View form opens.

a. Configure the Name parameter.

b. Use the filtering fields to refine your search, then enable the Visible checkbox for each NE to be included as part of the new view.

c. Click Apply & Save. The new NE view is applied to the map.

d. If required, click on the appropriate icon next to an existing NE view's Name in the drop-down menu to Edit or Delete that NE view.

END OF STEPS

4.4 To modify the Service Fulfillment topology map

4.4.1 Purpose

The following procedure can be used to modify the appearance of the Service Fulfillment topology map in ways that can assist with visualizing topology changes.

4.4.2 Steps

1

Click on the menu button in the top left corner of the Service Fulfillment application.

2

Perform one of the following:

a. Choose Map from the drop-down list. Continue to Step 3.

b. Choose Path Highlight From the drop-down list. Go to Step 4.

c. Choose Resignal All PCE LSPs. The PCE LSPs are resignalled.

d. Choose Resync NMS. The topology map is resynchronized with the NMS.

3

Perform one of the following:

a. Choose Refresh. The map is refreshed.

b. Choose Rebuild. The map is rebuilt.

c. Choose Auto Layout. The map is automatically laid out.

d. Choose Sync With Physical Map. The map is synchronized with the physical map.

4

The Path Highlight form opens. Configure the required parameters:

Parameter	Description
Disjoint	The Disjoint mode to be used in path computation
Bidirectional	The birdirectional mode to be used in path computation
Objective (Optimize on)	Specifies the primary goal when identifying paths for path computation

Parameter	Description
Max Hops (Span)	The Max Hops constraint to be used in path computation
Max Cost	The Max Cost constraint to be used in path computation
Bandwidth (Mbps)	Specifies the bandwidth required for the path
Reverse Bandwidth (Mbps)	Specifies the bandwidth required for the returning path
Segment Routing	Specifies whether or not segment routing can be used for the path
RSVP	Specifies whether or not RSVP can be used for the path
Source	Specifies the network element that will serve as the source for the path
Destination	Specifies the network element that will serve as the destination for the path
Secondary Source	Specifies the network element that will serve as the secondary source for the path
Secondary Destination	Specifies the network element that will serve as the secondary destination for the path

The path is highlighted.

END OF STEPS

4.5 To customize area colors

4.5.1 Purpose

Links on the IGP map are assigned a color that is specific to their area. Use this procedure to customize the color of these areas. Once a color is assigned to an area, that color cannot be used in any highlights originating, traversing, or terminating in that area.

4.5.2 Steps

1

Select the IP Area map or IP IGP map from the Map drop-down list.

2

Expand the Look and Feel heading from the Controls on the right side of the map.

3

Expand the Link Colors heading.

4

Hover over an Area's color and choose an alternate color from the palette.

5

If required, click on Reset Colors to return all areas to their original colors.

END OF STEPS

4.6 To create physical links between ports

4.6.1 Purpose

The Service Fulfillment application can be used to create a physical link between ports. This can also be accomplished using the NSD and NRC REST APIs. For more information, see the *NSP API Programmer Guide*.

Physical links will exist only in the NSD database; nothing will be provisioned to the NFM-P or NFM-T. The operational state of the physical links will be determined by the operational state of one, or both, of the linked ports. Updates to the link may be required when the ports' operational state changes. The NSD can also be used to delete physical links that have been created using the NSD.

4.6.2 Steps

1

Right-click on the desired node and choose Create Physical Link from the contextual menu. The Create Physical Link form opens.

2

Use the drop-down menus to specify the Source Port, the Destination Network Element, and the Destination Port.

3

Click Submit. The physical link is created.

4

If required, right-click on the physical link and choose Delete Physical Link to delete the physical link.

END OF STEPS

4.7 To create PCE-initiated LSPs

4.7.1 Purpose

Use this procedure to create PCE-initiated LSPs from the Service Fulfillment application

4.7.2 Steps

1

Click on the PCE LSPs tab.

2

Click on the Add button. The Initiate PCEP LSPs form opens.

3

Configure the required parameters:

Parameter	Description
Path Name	The name of the PCE-initiated LSP
PCC Address	The address of the PCC
Objective (Optimize on)	Specifies the primary goal when identifying path resources
Max Hops (Span)	Specifies the maximum number of hops to consider
Bandwidth (Mbps)	Specifies the bandwidth required for the LSP
Include Any Bit Pos	Specifies any bit between 0 and 31 to exclude
Exclude Any Bit Pos	Specifies any bit between 0 and 31 to exclude

Parameter	Description
Path Type	Specifies the type of path (must be Segment Routing)
Source	Specifies the source node for the path
Destination	Specified the destination node for the path
Profile ID	Specifies the identifier of the path profile to apply
Group ID	Specifies the identifier of the group to which this LSP belongs

4

Click Submit. The PCE-initiated LSP is created.

END OF STEPS

4.8 To manually resignal LSPs

4.8.1 Purpose

Use this procedure to manually resignal one or more LSPs.

4.8.2 Steps

1

Click on the PCE LSPs tab.

2

Select one or more LSPs in the list and right-click.

3

Choose Resignal from the contextual menu. The LSPs are resignalled.

Note: Only LSPs that have been delegated to the NRC-P can be resignalled.

END OF STEPS

4.9 To configure override path profiles for LSPs

4.9.1 Purpose

Use this procedure to override specific LSP behaviors by associating an override path profile to a PCE-delegated LSP, even those that already have an associated path profile.

4.9.2 Steps

1

Click on the PCE LSPs tab.

2

Select one or more LSPs in the list and right-click.

3

Choose Profile Override > Configure Path Profile Override from the contextual menu. The Configure Path Profile Override form opens.

4

Perform one of the following:

a. To associate an override path profile to the LSP(s), choose a previously-configured Path Profile from the Profile ID drop-down menu and click Submit.

b. To remove an override path profile from the LSP(s), choose Remove Profile Override from the Profile ID drop-down menu and click Submit.

5

If override path profile association fails, select the affected LSPs in the list and right-click.

6

Choose Profile Override > Reset Failed Override Profile from the contextual menu to retry the association.

Note: The path profile may need to be modified in order for the association to succeed.

END OF STEPS

4.10 Policy Management

4.10.1 Introduction

The Policy Management application enables customized, policy-driven behavior. It also maintains rules that allow the NSD and NRC modules to customize network policies such as global or domain rules for routing algorithms selection and execution. The Policy Management application provides service definitions that describe services in highly abstract, customizable ways while supporting mapping (mediation) of these definitions to the low-level network concepts.

For more information about the templates and policies that can be created and modified using the TPolicy Management application, see Chapter 8, “Templates and policies”.

The Policy Management application can be accessed at the following URL:

https://<*server*>:8543/policy-template

Where *server* is the hostname or IP address of your installed NSD and NRC server.

4.11 Task Scheduler

4.11.1 Introduction

The Task Scheduler application enables users to do CRUD operations with respect to scheduling bandwidth modification requests/tasks on an existing E-Line service. The user is able to schedule a one time, or repeatable service modification request (such as bandwidth modification) for their E-Line service. After accepting a scheduled task, the application allows the end user to view, modify, or delete existing tasks. In the case of modification, the user can change both the start date and the task execution intervals. The user can view all of their current requests and the state of those requests (Scheduled / Running / Disabled). The user can see a historical log of all executed tasks and their Success/Fail status and results.

For more information about the bandwidth modification tasks that can be created and modified using the Task Scheduler application, see Chapter 9, “Bandwidth modification”.

The Task Scheduler application can be accessed at the following URL:

https://<*server*>:8543/scheduler

Where *server* is the hostname or IP address of your installed NSD and NRC server.

4.12 Autonomous System Optimizer

4.12.1 Introduction

The Autonomous System Optimizer application is used to steer traffic on monitored routers, on a per-destination-AS-basis, to alternate next hops. Steering per destination AS implies that steering will be performed for all prefixes associated with a given destination AS. The NRC-F will automatically correlate the destination AS number to the set of prefixes associated with it. Steering is accomplished using the NRC-F Openflow controller, by automatically adding an Openflow flow rule per destination subnet. This allows the user to offload high traffic usage from the uplinks onto alternate paths on a per-AS-basis.

Users can monitor traffic distribution on a set of uplinks so that link congestion and/or high bandwidth utilization can be identified per AS. The traffic monitoring is accomplished by collecting flow statistics, per AS, on Nokia 7750, 7450, and 7950 routers. These flow statistics are then communicated to the collector with IPFIX record encoding.

The application allows users to identify the set of top bandwidth consumption per destination AS, while the set of destination subnets associated with a given AS are automatically identified. Threshold Crossing Alarms (TCAs) on the monitored links can be tracked and a user can plot both real-time and historical port utilization.

For more information about using the AS-Based Traffic Optimization application to steer flows, see 4.13 “To steer flows to next hops for autonomous systems” (p. 51).

For more information about using the NSD and NRC REST APIs to steer flows, see the *NSP API Programmer Guide*.

The Autonomous System Optimizer application can be accessed at the following URL:

https://<*server*>:8543/traffic-optimization-as/routers

Where *server* is the hostname or IP address of your installed NSD and NRC server.

4.13 To steer flows to next hops for autonomous systems

4.13.1 Purpose

Use this procedure to steer AS-destined traffic from a monitored router to an alternate next hop.

4.13.2 Steps

1

From the AS page of the Autonomous System Optimizer application, click on the Steer AS button. The Steer Flows form opens.

2

Choose one or more destination ASes from the drop-down menu and click CONTINUE.

3

Perform one of the following:

a. Choose a next hop from the Select Next Hop drop-down menu and click CONTINUE.

b. Enter a valid IPv4 IP address in the Custom Next Hop field and click CONTINUE.

4

Verify your changes and click FINISH. The new flow is added.

END OF STEPS

4.14 To steer flows to next hops for VIP customers

4.14.1 Purpose

Use this procedure to steer the subnets of a VIP customer to a dedicated next hop.

4.14.2 Steps

1

Perform one of the following:

a. From the VIP customers page of the Autonomous System Optimizer application, click on the Subnets button. The Subnets page opens. Continue to Step 2.

b. From the VIP customers page of the Autonomous System Optimizer application, click on the Steer VIP customer button. The Steer Flows form opens. Go to Step 3.

2

Click on the Steer VIP Subnets button. The Steer Flows form opens.

3

Perform one of the following:

a. Click CONTINUE.

b. Enable the *Steer all subnets not already steered* checkbox and click CONTINUE.

c. With the *Steer all subnets not already steered* checkbox enabled, delete subnets from the Selected Subnets drop-down menu and click CONTINUE.

d. With the *Steer all subnets not already steered* checkbox disabled, choose one or more subnets from the Subnet(s) to steer drop-down menu and click CONTINUE.

4

Perform one of the following:

a. Choose a next hop from the Select VIP Next Hop drop-down menu and click CONTINUE.

b. Choose a next hop from the Select Next Hop drop-down menu and click CONTINUE.

c. Enter a valid IPv4 IP address in the Custom Next Hop field and click CONTINUE.

5

Verify your changes and click FINISH. The new flow is added.

END OF STEPS

4.15 Traffic Steering Controller

4.15.1 Introduction

The Traffic Steering Controller application allows for the manipulation of flow rules, such as the addition or deletion of flow rules using the NRC-F Openflow controller. The application also allows flow tables from the Openflow switches of any 7x50 router within a network to be displayed. Nokia SROS OpenFlow Experimenters are supported, and can redirect traffic to alternate next hops, using plugins.

For more information about using the Traffic Steering Controller application to add flows, see 4.16 “To add a flow” (p. 54).

For more information about using the NSP's REST API to add flows, see the *NSP API Programmer Guide*.

The Traffic Steering Controller application can be accessed at the following URL:

https://<*server*>:8543/flow-placement/switches

Where *server* is the hostname or IP address of your installed NSD and NRC server.

4.16 To add a flow

4.16.1 Purpose

Use this procedure to add a flow entry to the flow table of a router's Openflow switch.

4.16.2 Steps

1

From the Switches page of the Traffic Steering Controller application, select a switch from the list. The Switch Details form opens.

2

Click on the Add button. The Add Flow Entry from opens.

3

Configure the following parameters:

Parameter	Description
Application ID	The ID of the application that deployed the flow. Negative numbers are reserved and should not be used.
Cookie	The hexadecimal, controller-issued ID for the flow
Priority	The priority of the flow

Click Continue.

4

As required, choose one or more match criteria from the drop-down list and click Continue.

5

Choose an instruction type from the drop-down list and click Continue. The flow entry is created.

END OF STEPS

Part II: Installation

Overview

Purpose

This volume describes the NSD and NRC installation process, and all necessary integration activities.

Contents

5 NSD and NRC installation and upgrade

5.1 Introduction

5.1.1 Overview

This chapter describes the NSD and NRC installation and upgrade processes, as well as related operations.

Note: Before the NSD and NRC modules can be used, an NSP token must be acquired, a user and a tenant must both be created, and the user must be assigned to the tenant. For more information about these tasks, see the *NSP API Programmer Guide*.

5.2 RHEL OS installation requirements

5.2.1 Introduction

Each NSD and NRC server requires the following:

- a specific RHEL Software Selection as the base environment
- the installation and removal of specific OS packages

Note: The RHEL rpm utility requires hardware driver files in binary format. If the RHEL driver files provided by your server hardware vendor are in source rpm format, you may need to install additional packages in order to compile the files into binary format. See the station hardware documentation for information.

5.2.2 Using the yum utility

To simplify package management, it is recommended that you use the RHEL yum utility to install and remove OS packages.

The package installation syntax is the following:

```
yum -y install package_1 package_2 ... package_n ↵
```

The package removal syntax is the following:

```
yum -y remove package_1 package_2 ... package_n ↵
```

Note: Package installation using yum requires a yum repository. The following repository types are available:

- local repository, which you can create during the RHEL OS installation

- Internet-based repository, which you can access after you register with the Red Hat Network

See the RHEL documentation for information about setting up a yum repository.

Note: If a package has dependencies on one or more additional packages that are not listed in a table, the yum utility installs the additional packages.

5.2.3 Description

During the RHEL OS installation for an NSD and NRC server, you must do the following.

- Specify “Minimal Install” as the Software Selection in the RHEL installer.
- Install specific OS packages, as described in 5.2.4 “RHEL OS packages to install” (p. 57)
- Remove specific OS packages, as described in 5.2.5 “ RHEL OS packages to remove” (p. 61)

5.2.4 RHEL OS packages to install

You must install a set of RHEL OS packages that are common to each NSD and NRC server. Most of the common packages are available from the RHEL ISO disk image and the default RHEL package repository. Such packages are listed in “Required packages, RHEL ISO image or default RHEL repository” (p. 57).

You must also install additional packages that are available only from the RHEL optional package repository. Such packages are listed in “Required packages, RHEL optional package repository” (p. 61).

Required packages, RHEL ISO image or default RHEL repository

The RHEL ISO image and default package repository each contain the following OS packages that you must install. To facilitate the installation, copy the following command block and paste it in a CLI:

```
yum -y install @base @gnome-desktop @legacy-x @x11
yum -y install autofs bc.x86_64 binutils.x86_64 compat-libcap1.x86_64
yum -y install dialog elfutils-libelf-devel.x86_64 elfutils.x86_64
yum -y install firefox.x86_64 ftp gcc.x86_64 gcc-c++.x86_64 glibc.i686
yum -y install glibc.x86_64 glibc-devel.i686 glibc-devel.x86_64
yum -y install gtk2.i686 haproxy.x86_64 irqbalance.x86_64
yum -y install keepalived.x86_64 ksh.x86_64 libaio.i686 libaio.x86_64
yum -y install libaio-devel.i686 libaio-devel.x86_64 libgcc.i686
yum -y install libgcc.x86_64 libibverbs.x86_64
yum -y install libstdc++.i686 libstdc++.x86_64 libstdc++-devel.i686
yum -y install libstdc++-devel.x86_64 libXi.i686 libXi.x86_64
yum -y install libXrender.i686 libXtst.i686 libXtst.x86_64 lshw.x86_64
yum -y install lsof.x86_64 make.x86_64 man net-snmp net-snmp-utils
yum -y install nfs-utils ntp numactl-devel.i686 numactl-devel.x86_64
```

```
yum -y install openssh.x86_64 openssh-askpass.x86_64
yum -y install openssh-clients.x86_64 openssh-server.x86_64
yum -y install procps rsync.x86_64 tcpdump.x86_64 unzip.x86_64
yum -y install which xinetd.x86_64 zip.x86_64
```

Table 1 Required OS packages from default RHEL repository or ISO image

Package name	Description
@base	Base package group
@gnome-desktop	Gnome package group
@legacy-x	Legacy X package group
@x11	X11 package group
autofs	A tool for automatically mounting and unmounting filesystems
bc.x86_64	GNU's bc (a numeric processing language) and dc (a calculator)
binutils.x86_64	A GNU collection of binary utilities
compat-libcap1.x86_64	Library for getting and setting POSIX.1e capabilities
dialog	A utility for creating TTY dialog boxes
elfutils.x86_64	A collection of utilities and DSOs to handle compiled objects
elfutils-libelf-devel.x86_64	Development support for libelf
firefox.x86_64	Mozilla Firefox web browser
ftp	The standard UNIX FTP client
gcc.x86_64	Various compilers, for example, C, C++, Objective-C, and Java
gcc-c++.x86_64	C++ support for GCC
glibc.i686	The GNU libc libraries
glibc.x86_64	The GNU libc libraries
glibc-devel.i686	Object files for development using standard C libraries
glibc-devel.x86_64	Object files for development using standard C libraries
gtk2.i686	The GIMP ToolKit (GTK+), a library for creating GUIs for X
hdparm.x86_64	Utility for displaying and/or setting hard disk parameters
irqbalance.x86_64	Daemon that evenly distributes IRQ load across multiple CPUs
ksh.x86_64	The Original ATT Korn Shell
libaio.i686	Linux-native asynchronous I/O access library
libaio.x86_64	Linux-native asynchronous I/O access library
libaio-devel.i686	Development files for Linux-native asynchronous I/O access

Table 1 Required OS packages from default RHEL repository or ISO image (continued)

Package name	Description
libaio-devel.x86_64	Development files for Linux-native asynchronous I/O access
libgcc.i686	GCC version 4.8 shared support library
libgcc.x86_64	GCC version 4.4 shared support library
libibverbs.x86_64	Core user space library that implements hardware abstracted verbs protocol
libstdc++.i686	GNU Standard C++ Library
libstdc++.x86_64	GNU Standard C++ Library
libstdc++-devel.i686	Header files and libraries for C++ development
libstdc++-devel.x86_64	Header files and libraries for C++ development
libXi.i686	X.Org X11 libXi runtime library
libXi.x86_64	X.Org X11 libXi runtime library
libXrender.i686	X.Org X11 libXrender runtime library
libXtst.i686	X.Org X11 libXtst runtime library
libXtst.x86_64	X.Org X11 libXtst runtime library
lshw.x86_64	Hardware lister
lsof.x86_64	Provides a utility to list information about open files
make.x86_64	GNU tool which simplifies the build process for users
man	A set of documentation tools: man, apropos and whatis
mcelog	Tool to translate x86-64 CPU Machine Check Exception data
net-snmp	The SNMP Agent Daemon and documentation
net-snmp-utils	SNMP clients such as snmpget and snmpwalk
nfs-utils	NFS utilities and supporting clients and daemons for the kernel
ntp	The NTP daemon and utilities
numactl-devel.i686	Development package for building Applications that use numa
numactl-devel.x86_64	Development package for building Applications that use numa
openssh.x86_64	Open source implementation of SSH protocol versions 1 and 2
openssh-askpass.x86_64	Passphrase dialog for OpenSSH and X
openssh-clients.x86_64	Open-source SSH client application
openssh-server.x86_64	Open source SSH server daemon
procps	OS utilities for /proc

Table 1 Required OS packages from default RHEL repository or ISO image (continued)

Package name	Description
rsync.x86_64	A program for synchronizing files over a network
tcpdump.x86_64	Command-line packet analyzer and network traffic capture; used by technical support for debugging
unzip.x86_64	A utility for unpacking zip files
which	Displays where a particular program in your path is located
xinetd.x86_64	A secure replacement for inetd
zip.x86_64	A file compression utility

Required packages, RHEL optional package repository

The RHEL optional package repository contains the following OS packages that you must install. To facilitate the installation, copy the following command and paste it in a CLI:

```
yum -y install compat-libstdc++-33.i686 compat-libstdc++-33.x86_64
```

Table 2 Required OS packages from RHEL optional package repository

Package name	Description
compat-libstdc++-33.i686	Compatibility standard C++ libraries
compat-libstdc++-33.x86_64	Compatibility standard C++ libraries

5.2.5 RHEL OS packages to remove

Table 3, “RHEL OS packages to remove” (p. 62) lists the OS packages that you must remove after you install the required OS packages on a component station. To facilitate the package removal, copy the following command block and paste it in a CLI:

```
yum -y remove anaconda-core.x86_64 anaconda-gui.x86_64
yum -y remove anaconda-tui.x86_64 avahi.x86_64 biosdevname
yum -y remove dnsmasq.x86_64 dosfstools gnome-boxes.x86_64
yum -y remove initial-setup.x86_64 initial-setup-gui.x86_64 kexec-tools
yum -y remove libstoragemgmt.x86_64 libstoragemgmt-python.noarch
yum -y remove libvirt-daemon-config-network.x86_64
yum -y remove libvirt-daemon-driver-network.x86_64
yum -y remove libvirt-daemon-driver-qemu.x86_64
yum -y remove libvirt-daemon-kvm.x86_64 libvirt-gconfig.x86_64
yum -y remove libvirt-gobject.x86_64 NetworkManager.x86_64
yum -y remove NetworkManager-libreswan.x86_64
yum -y remove NetworkManager-libreswan-gnome.x86_64
yum -y remove NetworkManager-team.x86_64 NetworkManager-tui.x86_64
```

```
yum -y remove NetworkManager-wifi.x86_64 qemu-kvm.x86_64
yum -y remove qemu-kvm-common.x86_64 setroubleshoot.x86_64
yum -y remove setroubleshoot-plugins.noarch
yum -y remove setroubleshoot-server.x86_64
yum -y remove subscription-manager-initial-setup-addon.x86_64
```

Table 3 RHEL OS packages to remove

Package name	Description
biosdevname	Utility that provides an optional convention for naming network interfaces
NetworkManager.x86_64	Network connection manager and user applications
NetworkManager-libreswan.x86_64	NetworkManager VPN plugin for libreswan
NetworkManager-libreswan-gnome.x86_64	NetworkManager VPN plugin for libreswan - GNOME files
NetworkManager-team.x86_64	Team device plugin for NetworkManager
NetworkManager-tui.x86_64	NetworkManager curses-based UI
NetworkManager-wifi.x86_64	Wifi plugin for NetworkManager
anaconda-core.x86_64	Core of the Anaconda installer
anaconda-gui.x86_64	Graphical user interface for the Anaconda installer
anaconda-tui.x86_64	Textual user interface for the Anaconda installer
avahi.x86_64	Local network service discovery
dnsmasq.x86_64	A lightweight DHCP/caching DNS server
gnome-boxes.x86_64	A simple GNOME 3 application to access remote or virtual systems
initial-setup.x86_64	Initial system configuration utility
initial-setup-gui.x86_64	Graphical user interface for the initial-setup utility
libstoragemgmt.x86_64	Storage array management library
libstoragemgmt-python.noarch	Python2 client libraries and plug-in support for libstoragemgmt
libvirt-daemon-config-network.x86_64	Default configuration files for the libvirtd daemon
libvirt-daemon-driver-network.x86_64	Network driver plugin for the libvirtd daemon
libvirt-daemon-driver-qemu.x86_64	Qemu driver plugin for the libvirtd daemon
libvirt-daemon-kvm.x86_64	Server side daemon & driver required to run KVM guests
libvirt-gconfig.x86_64	libvirt object APIs for processing object configuration

Table 3 RHEL OS packages to remove (continued)

Package name	Description
libvirt-gobject.x86_64	libvirt object APIs for managing virtualization hosts
qemu-kvm.x86_64	QEMU metapackage for KVM support
qemu-kvm-common.x86_64	QEMU common files needed by all QEMU targets
setroubleshoot.x86_64	Helps troubleshoot SELinux problem
setroubleshoot-plugins.noarch	Analysis plugins for use with setroubleshoot
setroubleshoot-server.x86_64	SELinux troubleshoot server
subscription-manager-initial-setup-addon.x86_64	Initial setup screens for subscription manager

5.3 To install or upgrade a standalone NSD and NRC system

5.3.1 Purpose

Use this procedure to install or upgrade a standalone NSD and NRC system. Upgrades are supported from NSP Release 2.0 R1 and later. If you need to upgrade from NSP Release 1.1 R2 or earlier, please contact your Nokia support representative.

Note: By supplying new values for the parameters within the configuration file, then executing the installation commands, this procedure can also be used to update the capabilities of an existing NSD and NRC system.

5.3.2 Before you begin

Before executing the NSD and NRC installer, ensure that your system meets the hardware and software requirements described in the *NSP NSD and NRC Planning Guide*.

An NRC-F, NRC-P, NRC-T, or NSD license must be obtained from Nokia personnel and placed in the license folder. The modules will not initialize without a valid license file in this folder.

Installation of the NSD and NRC modules requires IP reachability between the modules and Openstack Keystone, as well as any external systems with which the modules will integrate, such as NFM-P or NFM-T. For information about installing these components, see their respective documentation suites.

If the NSD and NRC modules are being upgraded from an earlier release of NSP to NSP Release 17.3 or later, and the NFM-P module will be part of the deployment, 5.7 “To configure the NSP security message” (p. 73) will need to be performed.

5.3.3 Steps

1

Download the NSD and NRC installer bundle from OLCS and extract it on any system running a supported version of RHEL 7. This does not have to be the system on which the NSD and NRC modules will be installed, as the installer is able to perform remote installations.

Note: When performing remote operations, SSH connections are used between the system where the NSD and NRC installer bundle was extracted and the system(s) on which it will execute its tasks. Therefore, SSH connections must be possible between these systems without the use of passwords, which requires the configuration of SSH keys, or the *--ask-pass* argument must be used when running the install.sh or uninstall.sh utilities, which requires that all systems share the same root user SSH password.

2

Create a hosts file in the directory where the NSD and NRC installer bundle was extracted and add the following entries:

```
[nspos]

<ip address>

[sdn]

<ip address>
```

where

IP address is the IP address of the server where the NSD and NRC software will be deployed. This same interface will also be used by the NSD and NRC modules.

Note: If performing an upgrade, use the IP address of the server where a previous version of the NSD and NRC modules are deployed.

Note: A standalone NSD and NRC system can be upgraded and converted to a redundant NSD and NRC system simultaneously by populating the hosts file with the IP address of the NSD and NRC server that will serve as the standby site. See 5.5 “To convert a standalone NSD and NRC system to a redundant NSD and NRC system” (p. 69) for more information.

3

Create a YAML or JSON configuration file in the directory where the NSD and NRC installer bundle was extracted and add only the configuration blocks that apply to your deployment. The *examples/* folder, which is bundled with the NSD and NRC installer, contains a sample configuration file for reference purposes.

Note: The Keystone *ip* parameter is the only parameter that must be populated in order for a standalone NSD and NRC instance to initialize.

The configuration file parameters are defined in the table below:

Table 4 NSD and NRC configuration file parameters

Parameter	Definition
auto_start	Specifies whether or not the NSD and NRC modules will start once installation is complete
keystone — Keystone identity parameters	
ip	The IP address of the Keystone server
token	The token used when connecting to the Keystone server
nfmp — Used when integrating with NFM-P	
primary_ip	The IP address of the primary NFM-P server
standby_ip	The IP address of standby NFM-P server
cert_provided	Specifies whether or not a custom SSL certificate is to be used to connect to the NFM-P, true or false
nfmt — Used when integrating with NFM-T	
primary_ip	The IP address of the primary NFM-T server
standby_ip	The IP address of standby NFM-T server
username	The user name used to login to the NFM-T
password	The password used to login to the NFM-T
cert_provided	Specifies whether or not a custom SSL certificate is to be used to connect to the NFM-T, true or false
sros — Used when integrating with vSROS	
enabled	Specifies whether or not to enable integration with vSROS

Table 4 NSD and NRC configuration file parameters (continued)

Parameter	Definition
ip	The IP address of the vSROS
router_ID	The router ID of the vSROS
ssl — Used to customize SSL security	
custom_keystore_path	The path to the custom keystore
custom_truststore_path	The path to the custom truststore
custom_keystore_password	The password used to access the custom keystore
custom_truststore_password	The password used to access the custom truststore
custom_key_alias	The alias of the certificate used in the custom keystore
custom_key_password	The password used to access the key within the custom keystore
ean — External applications notifications parameters	
max_subscribers	The maximum number of subscribers who can receive external applications notifications

Note: If performing an upgrade, the parameter values should be configured to align with your existing NSD and NRC system.

Note: Parameters not being configured should be removed from the configuration file entirely. Failing to provide a value for a parameter may have undesired consequences.

4

Copy the appropriate license file(s) into the license directory where the NSD and NRC installer bundle was extracted.

5

If required, copy the SSL certificates into the installer directory. The folders are *ssl/nfmp* and *ssl/nfmt*.

6

Install the NSD and NRC. As root user, execute the following commands:

```
cd bin

./install.sh
```

Note: Following an upgrade, the API certificates of all northbound platforms that are integrated with the NSD and NRC system must be refreshed.

END OF STEPS

5.4 To install or upgrade a redundant NSD and NRC system

5.4.1 Purpose

Use this procedure to install or upgrade an NSD and NRC system with 1+1 redundancy, which requires the installation of both a master NSD and NRC instance, and a standby NSD and NRC instance. See the *NSP NSD and NRC Planning Guide* for more information about redundant deployments. Upgrades are supported from NSP Release 2.0 R1 and later. If you need to upgrade from NSP Release 1.1 R2 or earlier, please contact your Nokia support representative.

An NSD and NRC redundant license must be obtained from Nokia personnel and placed in the license folder. The instances will not initialize without a valid license file in this folder.

5.4.2 Before you begin

Before executing the NSD and NRC installer, ensure that your system meets the hardware and software requirements described in the *NSP NSD and NRC Planning Guide*.

An NRC-F, NRC-P, NRC-T, or NSD license must be obtained from Nokia personnel and placed in the license folder. The modules will not initialize without a valid license file in this folder.

Installation of the NSD and NRC modules require IP reachability between the modules and Openstack Keystone, as well as any external systems with which the modules will integrate, such as NFM-P or NFM-T. For information about installing these components, see their respective documentation suites.

Before performing an upgrade, all processes should be stopped and a database backup should be taken.

If the NSD and NRC modules are being upgraded from an earlier release of NSP to NSP Release 17.3 or later, and the NFM-P module will be part of the deployment, 5.7 “To configure the NSP security message” (p. 73) will need to be performed.

5.4.3 Steps

1

Download the NSD and NRC installer bundle from OLCS and extract it on any system running a supported version of RHEL 7. This does not have to be the system on which the NSD and NRC modules will be installed, as the installer is able to perform remote installations.

Note: When performing remote operations, SSH connections are used between the system where the NSD and NRC installer bundle was extracted and the system(s) on which it will execute its tasks. Therefore, SSH connections must be possible between these systems without the use of passwords. Otherwise, the *--ask-pass* argument must be used when running the install.sh or uninstall.sh utilities, which will require that all systems share the same root user SSH password.

2

Create a hosts file in the directory where the NSD and NRC installer bundle was extracted and add the following entries:

```
[nspos]

<primary server address> dc=<location>

<standby server address> dc=<location>

[sdn]

<primary server address> dc=<location>

<standby server address> dc=<location>
```

where

primary server address is the IP address of the primary NSD and NRC server

standby server address is the IP address of the standby NSD and NRC server

location is the datacenter in which the given server resides. This string must be unique to each server in the redundant deployment

Note: If performing an upgrade, use the IP addresses of the servers where a previous version of the NSD and NRC modules are deployed.

3

Create a YAML or JSON configuration file in the directory where the NSD and NRC installer bundle was extracted and add only the configuration blocks that apply to

your deployment. The *examples/* folder, which is bundled with the NSD and NRC installer, contains a sample configuration file for reference purposes.

Note: The Keystone *ip* parameter is the only parameter that must be populated in order for a redundant NSD and NRC instance to initialize.

The parameters are defined in Table 4, “NSD and NRC configuration file parameters” (p. 65).

Note: If performing an upgrade, the parameter values should be configured to align with your existing NSD and NRC system.

Note: Parameters not being configured should be removed from the configuration file entirely. Failing to provide a value for a parameter may have undesired consequences.

4

Copy the appropriate license file(s) into the license directory where the NSD and NRC installer bundle was extracted.

5

Install the NSD and NRC. Execute the following commands:

```
cd bin
```

```
./install.sh
```

The NSD and NRC modules are automatically deployed on both servers.

Note: Following an upgrade, the API certificates of all northbound platforms that are integrated with the NSD and NRC system must be refreshed.

END OF STEPS

5.5 To convert a standalone NSD and NRC system to a redundant NSD and NRC system

5.5.1 Purpose

Use this procedure to convert a previously-installed standalone NSD and NRC system to a redundant NSD and NRC system.

5.5.2 Steps

1

Modify the existing hosts file in the directory where the NSD and NRC installer bundle was extracted as follows:

```
[nspos]

<primary server address> dc=<location>

<standby server address> dc=<location>

[sdn]

<primary server address> dc=<location>

<standby server address> dc=<location>
```

where

primary server address is the IP address of the primary NSD and NRC server

standby server address is the IP address of the standby NSD and NRC server

location is the datacenter in which the given server resides. This string must be unique to each server in the redundant deployment

2

Copy the appropriate license file(s) into the *license/* folder where the NSD and NRC installer bundle was extracted.

3

Install the NSD and NRC. Execute the following commands on one of the servers:

```
cd bin

./install.sh
```

The NSD and NRC modules are automatically deployed to both servers.

END OF STEPS

5.6 To migrate from an NSD and NRC system in HA mode to a redundant NSD and NRC system

5.6.1 Purpose

Use this procedure to convert a previously-installed NSD and NRC system in HA mode to a redundant NSD and NRC system.

Note: External systems that had been configured to interact with the NSD and NRC system in HA mode will need to be reconfigured so as to be aware of the IP addresses for both the primary and standby NSD and NRC servers.

Note: If the deployment includes an NFM-P module that will be upgraded to 17.3, and a single SSL certificate will be used for both the NFM-P and the NSD and NRC modules, that certificate must be generated with the appropriate values populated in the SAN section. See 5.12 “To generate a keystore” (p. 83) for more information.

5.6.2 Before you begin

Before executing the NSD and NRC installer, ensure that your system meets the hardware and software requirements described in the *NSP NSD and NRC Planning Guide*.

An NRC-F, NRC-P, NRC-T, or NSD license must be obtained from Nokia personnel and placed in the license folder. The modules will not initialize without a valid license file in this folder.

Installation of the NSD and NRC modules require IP reachability between the modules and Openstack Keystone, as well as any external systems with which the modules will integrate, such as NFM-P or NFM-T. For information about installing these components, see their respective documentation suites.

Before performing an upgrade, all processes should be stopped and a database backup should be taken.

5.6.3 Steps

Reduce the HA cluster from three servers to two servers

1

Shutdown all NSD and NRC instances. Execute:

```
/opt/nsp/scripts/nsp-control stop
```

2

In the hosts file that corresponds to the installed NSD and NRC version (such as 2.0.R4), insert only the IP address of the server to be removed.

Note: If this is an HA disaster recovery deployment, the removed server should be one of the two in the primary site.

3

Remove the software from that server. Execute:

```
cd bin

./uninstall.sh
```

Install the two remaining servers that will form the redundant NSD and NRC system

4

In the 17.3 config.yml file, configure the *auto_start* parameter with a value of *false*.

5

Copy the appropriate license file(s) into the *license/* folder where the NSD and NRC installer bundle was extracted.

6

If the NSD and NRC deployment will include an NFM-P, the corresponding templates must be updated. See 5.15 “To install required NFM-P templates” (p. 86) for more information.

7

Execute:

```
bin/install.sh
```

8

On the server that will serve as the primary server, execute:

```
systemctl start nspos-nspd
```

9

Monitor the nsp.log file to ensure that the upgrade script completes.

10

Connect to the system and perform a basic sanity check.

11

On the standby server, execute:

```
systemctl start nspos-nspd
```

END OF STEPS

5.7 To configure the NSP security message

5.7.1 Purpose

Use this procedure to configure the security message that is displayed on the NSP login page.

Note: This procedure must be performed when the following conditions are met:

- upgrade is from an earlier release of NSP to NSP Release 17.3 or later
- deployment includes NSD and NRC modules, as well as NFM-P

5.7.2 Steps

Preserve the system security message

1

Copy the existing security message from the Java client.

2

Paste the copied message into an empty file, and save the file in text format.

3

Copy the file to a secure location that is unaffected by the system upgrade activity.

Upgrade the NSD and NRC and start the nspOS

4

Perform one of the following:

- Upgrade your standalone NSD and NRC system, as described in 5.3 “To install or upgrade a standalone NSD and NRC system” (p. 63).
- Upgrade your redundant NSD and NRC system, as descibed in 5.4 “To install or upgrade a redundant NSD and NRC system” (p. 67).

5

Start the nspOS.

Configure the NSP security message

6

Log in to the NSD and NRC server as the nsp user.

7

Open a console window.

8

Open the following file using a plain text editor, such as vi:

/opt/nsp/os/tomcat/webapps/cas/WEB-INF/classes/static/nls/<*language*>|18nStrings.json

where <*language*> is the language identifier, for example, en for English

Note: If the NSD and NRC system has been deployed in a redundant configuration, the above file must be modified on both the primary and standby servers.

9

Locate the following section:

```
"admin message": {
```

10

This section contains the following line:

```
"loginMessage-example" : "Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed do elusmod tempor incididunt
ut laboreet dolore"
```

11

Add a comma to the end of the line, which now reads:

```
"loginMessage-example" : "Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed do elusmod tempor incididunt
ut laboreet dolore",
```

12

Insert a line below the existing line that reads:

```
"loginMessage" : "<security_message>"
```

where <security_message> is the security message to display on the NSP login page, such as the text copied in 5.7.2 “Preserve the system security message” (p. 73).

The two lines now read:

```
"loginMessage-example" : "Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed do elusmod tempor incididunt
ut laboreet dolore",

"loginMessage" : "<security_message>"
```

13

Save and close the file.

END OF STEPS

5.8 To enable TCAs for NRC-F

5.8.1 Purpose

Use this procedure to enable Threshold Crossing Alarms (TCAs), which allow the NRC-F to receive port utilization information.

5.8.2 Steps

1

After completing NSD and NRC installation, execute the following command to stop the SDN and nspOS services:

```
nspdctl stop
```

2

In the /opt/nsp/configure/config/nrcf.conf file, set the value of the *tca* parameter to *true*.

3

Restart the SDN and nspOS services. Execute:

```
nspdctl start
```

END OF STEPS

5.9 To add the NSD and NRC modules to an existing NFM-P system

5.9.1 Before you begin

The following steps describe how to add the NSD and NRC modules to an existing NFM-P system.

5.9.2 Steps

CAUTION

Service Disruption

Performing this procedure involves stopping and starting each NFM-P main server, which is service-affecting.

This procedure must only be performed during a scheduled maintenance period of low network activity.

Note: The following user privileges are required:
- on each NFM-P main server station — root, nsp
- on each NSD and NRC server station — root

Note: The following RHEL CLI prompts in command lines denote the active user, and are not to be included in typed commands:
- # —root user
- bash$ —nsp user

Generate security files

1

Log in to an NFM-P main server station as the nsp user.

2

Open a console window and execute the following command:

```
bash$ cd /tmp ↵
```

3

Display the main server keystore. Execute:

```
bash$ /opt/nsp/os/jre/bin/keytool -list -v -keystore /opt/nsp/os/ssl/nsp.keystore ↵
```

The output includes the following:

```
#2: ObjectId: x.x.x.x Criticality=false
SubjectAlternativeName [
   IPAddress: 127.0.0.1
   IPAddress: nfmp_1_IP
   IPAddress: nfmp_2_IP
```

4

Generate a keystore file. Execute:

```
bash$ /opt/nsp/os/jre/bin/keytool -genkeypair -keystore nsp.
keystore -alias nsp -keyalg RSA -keypass <password> -
storepass <password> -validity 730 -dname "CN=NSP, O=Nokia" -
ext bc=ca:true -ext san=IP:127.0.0.1,IP:<nfmp_1_IP>,IP:<nfmp_
2_IP>,IP:<nsd_nrc_1_IP>,IP:<nsd_nrc_2_IP> ↵
```

where

password is the password to use for the key and keystore

nfmp_1_IP and *nfmp_2_IP* are the public IP addresses of the NFM-P main servers

nsd_nrc_1_IP and *nsd_nrc_2_IP* are the public IP addresses of the NSD and NRC servers

Note: The keypass and storepass password values must be identical.

Note: For a standalone NFM-P or NSD and NRC system, omit the IP: entry for the second server.

5

Record the password value that you specify.

Export certificate for NSD and NRC installer

6

Execute:

```
bash$ /opt/nsp/os/jre/bin/keytool -exportcert -keystore nsp.
keystore -storepass password -alias nsp -rfc -file nsp.pem ↵
```

where *password* is the password recorded in Step 5

Import certificate to new truststore file

7

Execute:

```
bash$ /opt/nsp/os/jre/bin/keytool -importcert -alias nsp -
keystore nsp.truststore -storepass password -noprompt -file
nsp.pem ↵
```

where *password* is the password recorded in Step 5

Transfer security files to servers

8

Copy the following generated files to an empty directory in each NFM-P main server station:

- nsp.keystore
- nsp.truststore

9

Copy the following generated files to an empty directory in each NSD and NRC server station:

- nsp.keystore
- nsp.truststore
- nsp.pem

Install the NSD and NRC modules

10

Perform 5.3 “To install or upgrade a standalone NSD and NRC system” (p. 63) or 5.4 “To install or upgrade a redundant NSD and NRC system” (p. 67).

Note: During installation, the *auto_start* parameter in the config.YAML file must be set to false, so that the NSD and NRC system does not start upon completion.

11

Log in to an NSD and NRC server as the root user and navigate to the directory where the NSD and NRC installer bundle was extracted.

12

Using a plain-text editor such as vi, open the config.yml file that was created and locate the `ssl` block.

13

Configure the following parameters as follows:

- custom_keystore_path: <path>/nsp.keystore
- custom_keystore_password: <password>
- custom_key_alias: nsp
- custom_key_password: <password>
- custom_truststore_path: <path>/nsp.truststore
- custom_truststore_password: <password>

where

path is the directory location of the security files transferred to the server

password is the password recorded in Step 5

14

Locate the `nfmp` block and configure the following parameters as follows:

- primary_ip: <nfmp_1_IP>
- standby_ip: <nfmp_2_IP>
- cert_provided: true

where *nfmp_1_IP* and *nfmp_2_IP* are the public IP addresses of the NFM-P main servers

15

Set the value of the `auto_start` parameter to `false`.

16

Edit the other parameters in the file as required for your deployment, then save and close the config.yml file.

17

Execute:

```
# cp <path>/nsp.pem ssl/nfmp/nfmp.pem ↵
```

where *path* is the directory location of the security files transferred to the server

18

Execute:

```
# bin/install.sh ↵
```

Perform NFM-P data migration

19

Log in to the primary NFM-P main server station as the nsp user and execute the following commands:

Note: In a redundant system, you must stop the standby main server first.

```
bash$ cd /opt/nsp/nfmp/server/nms/bin ↲
bash$ ./nmsserver.bash stop ↲
bash$ ./nmsserver.bash appserver_status ↲
```

The server status is displayed. The server is fully stopped if the status is the following:

```
Application Server is stopped
```

If the server is not fully stopped, wait five minutes and then repeat this step. Do not perform the next step until the server is fully stopped.

20

Execute:

```
bash$ nspdctl start ↲
bash$ nspdctl logs ↲
```

21

Monitor the console output until the following is displayed:

```
Node /nokia/nsp/cluster created with value
```

22

Execute:

```
bash$ nspdctl backup -d nspos_migration -f ↲
```

A data backup operation begins.

23

Execute:

```
bash$ nspdctl backup status ↲
```

Output like the following is displayed:

```
Last-known backup status :  status
Last-known backup time   :  time
Last-known backup files  :  /opt/nsp/backup/nspos_migration/
nspos-neo4j_backup_timestamp.tar.gz
```

Ensure that the *status* value is `success`, and that the *time* value is current.

24

Execute:

```
bash$ nspdctl stop ↲
```

25

Transfer the */opt/nsp/backup/nspos_migration/nspos-neo4j_backup_<timestamp>.tar.gz* file created by the backup operation to the */tmp* directory on both the primary and standby NSD and NRC servers.

26

Log in as the nsp user on an NSD and NRC server station and execute the following commands to restore the backup:

```
bash$ mkdir /tmp/nspos-neo4j_backup
bash$ tar -xv -C /tmp/nspos-neo4j_backup -f /tmp/nspos-neo4j_backup_<timestamp>.tar.gz
bash$ /opt/nsp/os/neo4j/bin/neo4j-admin restore --from=/tmp/nspos-neo4j_backup/graph.db --database=graph.db --force
```

27

Login as the root user on each NSD and NRC server station, open a console window, and execute the following command to start the NSD and NRC server:

```
# systemctl start nspos-nspd ↲
```

Reconfigure NFM-P

28

Follow the 'SSL Configuration workflow' in the *NSP NFM-P Installation and Upgrade Guide* to reconfigure SSL for the NFM-P with the newly-generated certificates.

29

Perform the following steps on each NFM-P main server station.

Note: In a redundant NFM-P system, you must perform the steps on the primary main server station first.

1. Log in as the root user, open a console window, and execute the following command:

   ```
   # samconfig -m main ↲
   ```

   The following is displayed:

   ```
   Start processing command line inputs...
   <main>
   ```

2. Execute:

   `<main>` **`configure registry ip-list <registry_IP_1>; <registry_IP_2>`** ↲

   where *registry_IP_1* and *registry_IP_2* are the public IP addresses of the NSD and NRC servers

   The prompt changes to `<main configure registry>`.

3. Execute:

   `<main configure registry>` **`exit`**↲

   The prompt changes to `<main>`.

4. Enter the following:

   `<main>` **`apply`** ↲

   The configuration is applied.

5. Enter the following:

   `<main>` **`exit`** ↲

   The samconfig utility closes.

30

Close the open console windows, as required.

31

Start the NFM-P module.

END OF STEPS

5.10 To retrieve an NFM-P custom SSL certificate

5.10.1 Purpose

Use this procedure to retrieve a NFM-P custom SSL certificate for use with your NSD and NRC system.

5.10.2 Steps

1

Retrieve the cacerts.trustStore file from the */opt/nsp/nfmp/server/nms/config/ssl /trustStore/* directory on the NFM-P server.

2

Extract the certificate in from the trustStore using the java keytool utility. Execute the following command:

```
/opt/nsp/os/jre/bin/keytool keytool -exportcert -keystore
cacerts.trustStore -alias <cert_alias> -storepass <trustStore_
password> -rfc -file nfmp.pem
```

where

cert_alias is the alias of the certificate in the NFM-P trustStore

truststore_password is the password for the trustStore container

3

Place the generated nfmp.pem file in the *ssl/nfmp/* folder where the NSD and NRC installer bundle was extracted.

END OF STEPS

5.11 To retrieve an NFM-T custom SSL certificate

5.11.1 Purpose

Use this procedure to retrieve a NFM-T custom SSL certificate for use with your NSD and NRC system.

5.11.2 Steps

1

Retrieve the sslcert.pem file from the */alu/AAA-repository/1350OMS-Public /SSLCertificate/* directory on the 1350 OMS server.

2

Place the sslcert.pem file in the *ssl/nfmt/* folder where the NSD and NRC installer bundle was extracted.

END OF STEPS

5.12 To generate a keystore

5.12.1 Purpose

Keystores provide identity verification and encryption on all northbound and internal interfaces. A keystore is automatically generated by the NSD and NRC installer, however, this procedure can be used to manually generate a keystore. Keystores are required to be in the Java KeyStore (JKS) format. A keystore that contains a self-signed security certificate can be generated using the Java Keytool that ships with any Java Development Kit (JDK) or Java Runtime Environment (JRE).

5.12.2 Steps

1

Execute the following Keytool command:

```
./keytool –genkeypair –keystore <file name> –keypass <key
password>  -storepass <store password> –keyalg rsa –alias
<alias name> –dname "CN=<common name>, OU=<organizational
unit>, O=<organization>, L=<location>, ST=<state>, C=<country>
" –validity <days> -ext bc=ca:true -ext san=<SAN string>
```

where

file name is the absolute path to the Java KeyStore file that will hold the public /private key pair that is generated

key password is the password that is used to access the private key stored within the keystore

store password is the password to access the contents of the keystore

alias name is the human-readable identifier for the key pair that is used to differentiate between different keys in a keystore

common name is the name of the keystore owner

organizational unit is the name of the organizational unit to which the keystore owner belongs

organization is the name of the organization to which the keystore owner belongs

location is the name of the city in which the keystore owner resides

state is the name of the state or province in which the keystore owner resides

country is the name of the country in which the keystore owner resides

days is the integer value for the number of days for which the keys should be considered valid

SAN string is a list of all interfaces on the NSD and NRC server(s), pre-pended with the "IP:" string. This list must contain the loopback (127.0.0.1) interface.For example, a redundant NSD and NRC deployment with 2 servers having the IPs 10.0.0.1 and 10.0.0.2 would use: *-ext san=IP:127.0.0.1,IP:10.0.0.1,IP:10.0.0.2*. If hostnames were used during installation, they must be included, pre-pended with the "DNS:" string. For example, *-ext san=IP:127.0.0.1,DNS:<hostname>.nokia.com*.

2

Use the custom_keystore_path parameter, under the ssl section, to point to the generated keystore file. You should also set the other ssl values to match the parameters specified in the command listed above.

END OF STEPS

5.13 To retroactively add a license to the NSD and NRC

5.13.1 Purpose

Use this procedure to add a license file to an NSD and NRC server after the install script has been run.

5.13.2 Steps

1

Copy the appropriate license file(s) into the *license/* folder where the NSD and NRC installer bundle was extracted.

2

Run the install script to re-configure the NSD and NRC with the new license(s). Execute:

```
cd bin

./install.sh
```

3

Restart the Tomcat instance to activate the new license file. As root user, execute:

```
systemctl restart nsp-tomcat
```

Note: For redundant NSD and NRC systems, this step must be performed on both servers.

END OF STEPS

5.14 To retroactively enable SSL communication to the NFM-P

5.14.1 Purpose

Use this procedure to enable SSL communication to the NFM-P *after* NSD and NRC installation has been completed.

5.14.2 Steps

1

Copy the NFM-P certificate into the *ssl/nfmp/* folder where the NSD and NRC installer bundle was extracted.

2

Ensure that your NSD and NRC configuration file has been modified so as to enable SSL on NFM-P. For example:

```
nfmp:

  cert_provided: true
```

3

Run the install script to re-configure the NSD and NRC with NFM-P SSL configured. Execute:

```
cd bin

./install.sh
```

END OF STEPS

5.15 To install required NFM-P templates

5.15.1 Purpose

Use this procedure to install required NFM-P templates on the NFM-P server that is being used with the NSD and NRC modules.

5.15.2 Steps

1

Navigate to */opt/nsp/configure* in the Linux host environment.

2

Copy the entire samTemplates directory at this location to the NFM-P server that is being used with the NSD and NRC modules.

3

On the NFM-P server, navigate to the samTemplates directory and follow the instructions in the README file to install the required NFM-P Templates.

END OF STEPS

5.16 To restore the PostgreSQL and Neo4j databases

5.16.1 Purpose

Use this procedure to restore the PostgreSQL and Neo4j databases from backups following a catastrophic system failure.

5.16.2 Before you begin

Prior to restoring the databases, backups must be created using the **nspdctl backup** CLI command, or using the POST /backup/trigger/ REST API method. See the *NSP API Programmer Guide* for more information.

5.16.3 Steps

1

Stop the SDN and nspOS services. As root user, execute the following command:

```
nspdctl stop
```

Note: This command should be executed on both servers in a redundant NSD and NRC deployment.

2

To restore the PostgreSQL database, perform the following steps on a standalone NSD and NRC server, or on the primary server in a redundant deployment:

1. Stop the nspd agent. On all NSD and NRC servers, execute:

   ```
   systemctl stop nspos-nspd
   ```

2. Start the nsp-postgresql service. Execute:

   ```
   systemctl start nsp-postgresql
   ```

3. Extract the nsp-postgresql backup set. As nsp user, execute:

   ```
   tar -xv -C /tmp -f /opt/nsp/backup/backupset_1/nsp-
   postgresql_backup_<time stamp>.tar.gz
   ```

   Where *time stamp* is the date and time at which the backup was performed.

4. Run the database restore. Execute:

   ```
   /opt/nsp/scripts/db/pgsql/pg-restore.sh -f /tmp/nspdb.custom
   ```

3

If the NSD and NRC system was deployed in a redundant configuration, execute the following commands as the nsp user to restore the PostgreSQL database on the standby server:

```
/opt/nsp/server/pgsql/repmgr-standby-bootstrap.sh

/opt/nsp/server/pgsql/bin/repmgr -f /opt/nsp/server/pgsql/
conf/repmgr.conf --force standby register
```

4

To restore the Neo4j database, perform the following:

1. Extract the nspos-neo4j backup set. Execute:

   ```
   mkdir /tmp/nspos-neo4j-backup

   tar -xv -C /tmp/nspos-neo4j-backup -f /opt/nsp/backup/
   backupset_1/nspos-neo4j_backup_<time stamp>.tar.gz
   ```

   Where *time stamp* is the date and time at which the backup was performed.

2. Restore the nspos-neo4j backup set. Execute:

   ```
   /opt/nsp/os/neo4j/bin/neo4j-admin restore --from=/tmp/nspos-
   neo4j-backup/graph.db --database=graph.db --force
   ```

3. Extract the nsp-tomcat backup set. Execute:

   ```
   mkdir /tmp/nsp-tomcat-backup

   tar -xv -C /tmp/nsp-tomcat-backup -f /opt/nsp/backup/
   backupset_1/nsp-tomcat_backup_<time stamp>.tar.gz
   ```

   Where *time stamp* is the date and time at which the backup was performed.

4. Restore the nsp-tomcat backup set. Execute:

   ```
   /opt/nsp/scripts/db/neo4j/bin/neo4j-admin restore --from=/
   tmp/nsp-tomcat-backup/graph.db --database=graph.db --force
   ```

Note: These commands should be executed on both servers in a redundant NSD and NRC deployment.

5

Restart the nspd agent. As root user, execute:

```
systemctl start nspos-nspd
```

Note: This command should be executed on both servers in a redundant NSD and NRC deployment.

END OF STEPS

5.17 To disable websocket event notifications

5.17.1 Purpose

Websocket-based events are used by the NSD and NRC applications and are exposed only to the tenant who owns the resource in question, as well as to the admin GUI. This procedure can be used to disable websocket event notifications.

Note: The websocket connection used by the NSD and NRC modules may not work if the browser, or any client, is behind a proxy. Websocket communication through any entity that is positioned between the websocket client and server (such as proxies, firewalls, or load balancers) is dependent on how those entities are configured.

5.17.2 Steps

1

As nsp user, navigate to the following directory: */opt/nsp/configure/config*

2

Open the wsc-security.conf file.

3

Modify the section below as follows:

```
websocket{

 enableEvents=false

}
```

4

Restart the NSD and NRC modules. Execute:

```
systemctl restart nsp-tomcat
```

END OF STEPS

5.18 To uninstall an NSD and NRC system

5.18.1 Purpose

Use this procedure to uninstall either a standalone NSD and NRC system, or an HA NSD and NRC system.

5.18.2 Steps

1

Perform one of the following:

a. Modify the hosts file in the installer directory so as to contain the IP addresses of the systems from which the NSD and NRC software will be uninstalled.

b. Create a new hosts file, as described in 5.3 “To install or upgrade a standalone NSD and NRC system” (p. 63), that contains the IP addresses of the systems from which the NSD and NRC software will be uninstalled.

2

Execute the following commands:

```
cd bin/
```

```
./uninstall.sh
```

The NSD and NRC software is removed from all hosts declared in the hosts file.

END OF STEPS

6 Tenancy and roles

6.1 Introduction

6.1.1 Tenancy

While using the NSD and NRC REST API or applications, each user is associated with a tenant. A tenant is a logical group that allows the assigning of network resources. A user can only view the resources that are assigned to their tenant.

For information about creating, deleting, and assigning tenants, see the *NSP API Programmer Guide*.

6.1.2 Roles

While using the NSD and NRC REST API or applications, each user is assigned a role. A role specifies the type of access that a user has to their tenant. A user can only perform the operations that are authorized by their role.

A user can be assigned multiple roles, but will assume the assigned role with the highest priority. The roles are prioritized as follows:

Role	Permissions	Priority
Admin	Can modify and manipulate any object within NSD and NRC modules	1
Operator	Can perform read/write operations on assigned network resources	2
User	Can perform read only operations on assigned network resources	3

6.1.3 Openstack Keystone

The NSD and NRC modules maintain a generic model of users, tenants, and roles, which can be mapped to external identity provider systems . For example, the NSD and NRC modules are integrated with Openstack Keystone, which authenticates a user's tenancy whenever an operation is performed from the NSD and NRC REST API or applications.

Note: Openstack Keystone must be configured using the method and parameters described in 5.3 “To install or upgrade a standalone NSD and NRC system” (p. 63). The use of any other methods or parameters is not supported.

6.2 To manage tenants

6.2.1 Purpose

Use this procedure to specify which tenants are able to configure a given resource.

6.2.2 Steps

1

From the Service Fulfillment application, click on the All Services tab or the Endpoints tab.

2

Select one or more resources in the list, right-click, and choose Manage Tenants from the contextual menu. The Resource Tenancy Management form opens.

3

Perform one of the following:

a. To allow tenants to configure the specified resource(s), select a tenant in the Tenant drop-down menu and click on the Add button. The tenant is added to the list of authorized tenants. Repeat as required.

b. To prevent tenants from configuring the specified resource(s), select a tenant from the list of authorized tenants and click on the Delete button. The tenant is removed from the list of authorized tenants. Repeat as required.

END OF STEPS

Part III: Services and Templates

Overview

Purpose

This volume describes the services, templates, policies, and tasks that can be created using the NSD and NRC applications.

Contents

7 Services

Service provisioning

7.1 Service description

7.1.1 Introduction

This section describes each of the service types that can be provisioned from the Service Fulfillment application. To display a specific service when opening the Service Fulfillment application, the service's unique ID must be provided as part of the URL:

https://<*server*>:8543/nsd/?map=service&serviceId=<*service ID*>

Where

server is the hostname or IP address of your installed NSD and NRC server

service ID is the unique service ID of the service to be displayed

Note: The user must already be logged in to the Service Fulfillment application prior to modifying the URL as described.

For information about provisioning services using the NSP's REST API, see the *NSP API Programmer Guide*

7.1.2 Object life cycle

Object Life Cycle (OLC) is used to manage state transitions of objects inside the NSD as they go from the planning phase to the deployment phase. The planning phase includes four states:

- Planned
- Routing
- Routing Failed
- Routed and Save

The deployment phase includes five states:

- Waiting for Deployment
- Deploying
- Partially Deployed
- Deployment Failed
- Deployed

Note: Services that were not created by the NSD cannot be modified by the NSD. When a service is discovered or re-synchronized from NFM-P, its state is Deployed.

7.1.3 Service CAC

The NSD can perform bandwidth CAC and validation on access ports. Every port available for use in E-Line, C-Line, E-LAN, and L3 VPN services will have their available ingress bandwidth and available egress bandwidth displayed as read-only properties in the Service Fulfillment application and the NSP's REST APIs. When any of these ports are discovered, available bandwidth is initialized to port speed. In some cases, such as the 60-port 10/100 card when the port is operationally down, the port speed is zero. On fixed port speed cards, the port speed is populated, allowing services with bandwidth to be configured even when the port is down.

Note: Service CAC is not available on the variable-speed SFP-based cards.

Any changes to port speed will be reflected in the displayed available ingress bandwidth and available egress bandwidth. This may result in these fields displaying a negative value. No alarms or notifications will occur but a WARN level log will be generated.

A formula is used to calculate both the ingress and egress aggregate bandwidth of all endpoints used by E-Line, C-Line, ELAN, and L3 VPN services. The formula yields the sum of the CIR values, which is based on each of the configured queues and the scheduler policy of the QoS. This same value is used for E-Line service tunnel bandwidth calculation. No overbooking is applied to the formula. When the NSD creates a service on one of these endpoints, the validation code will make sure that the sum of the formula is less than, or equal to, the current available bandwidth on the port, otherwise the service will not be created and an error is returned.

The bandwidth is only booked after the traversal operation is run to match with the current behavior of the core bandwidth. It is possible that between the validation check and the traversal operation, the port bandwidth was consumed by another service. In this case, the OLC state is changed to Routing Failed, and the user is told that either the access port ingress or egress bandwidth was exceeded. Modifying the CIR will reinitiate the traversal operation. Similar operations occur when adding endpoints to an existing service and modifying endpoints. In the latter case, it is the bandwidth delta which is applied to the available ingress or egress bandwidth. Upon deletion of an endpoint or service, the available ingress or egress bandwidth is increased by the bandwidth of the endpoints.

Service CAC is available on both access and hybrid ports. If there are network interfaces on hybrid ports, these are not tracked as part of the available ingress or egress bandwidth. When an upgrade is performed, the available ingress and egress bandwidths will be calculated based on all existing services within the NSD. This may result in negative values. When in an overbooked state, any request that will not cause a change to bandwidth reservation, or that will cause a shrink in bandwidth reservation, will be permitted.

Service CAC is disabled by default. For information about enabling service CAC, see the 7.2 “To enable service CAC” (p. 105) procedure.

Note: Service CAC is supported on services originating from the NFM-P that have had their 'NSD-managed' flag enabled.

Note: Service CAC is not supported on multi-vendor services for which there is no access port bandwidth tracking.

7.1.4 E-Line services

E-Line services connect two customer Ethernet ports over a WAN. The NSD supports the creation of E-Line services over both IP and optical networks (L0). Whether IP or optical, when an E-Line service is deployed, the selection of the endpoints automatically utilizes the requisite technology (MPLS or L0 WDM) tunnels. For example, when the tunneling technology is MPLS, a service tunnel with a single LSP satisfying the service-specified constraints and objectives is automatically selected. The service is then bound to that LSP via the service tunnel. The LSP's available bandwidth is tracked by the NSD and is automatically adjusted to accommodate the E-Line service, which reserves bandwidth on the LSP.

If an existing E-Line service is modified (for example, to increase bandwidth), the service tunnel is resized to accommodate it, if permitted by policy. If the service tunnel resizing fails, the service tunnel may be rerouted onto links that cannot accommodate the resized service tunnel. If the reroute fails, then a new service tunnel is created. It is possible for E-Line services to use service tunnels that were not created using the NSD.

Note: Policies for service-to-tunnel binding dictate the rules associated with the service binding. If no service tunnel meets all the constraints, and this is a new E-Line service, a new service tunnel is created.

Other parameters of the E-Line service are obtained from the specific templates referenced in the abstract API definition. The service definition in the abstract API, the detailed configuration in the service templates, and other network and tunnel parameters form the complete service definition, which is represented in the normalized model for E-Line. Specific configurations based on the devices are then constructed and deployed using the NFM-P.

For information about provisioning E-Line services from the Service Fulfillment application, see the 7.4 “To provision E-Line services” (p. 108) procedure.

Note: SAP-to-SAP E-Line services can be provisioned, provided different ports are used for each endpoint.

Multi-domain E-Line service provisioning

The NSD supports multi-domain E-Lines that span any mix of MPLS and non-MPLS domains. The service tunnels must be already created in the MPLS domains. The non-MPLS domains can consist of only peer-to-peer Ethernet links.

In addition to the SAP-to-SAP and SAP-to-SDP service sites, the multi-domain E-Line service also supports SDP-to-SDP connections through the use of pseudowire switching. However, the NEs eligible for SDP-to-SDP pseudowire switching must be pre-configured with a pw-switching flag that is enabled on the NE.

The NSP calculates an optimal end-to-end path that traverses existing service tunnels, including VLAN handoff. Only strictly-routed RSVP-based service tunnels have calculations for the number of hops and accumulated IGP metric and latency. The VLAN handoffs have hard-coded hops, IGP metric and latency to 1. Other service tunnels have very large numbers for hops, IGP metric and latency and are usually less preferred. The non-RSVP service tunnels have zero bandwidth.

The multi-domain E-line service provisioning allows you to create the following types of E-Lines:

- vc-switched—in addition to the two terminating sites, an E-Line can include one or more switching sites
- composite—a composite E-Line consists of multiple component services connected through VLAN handoff to provide end-to-end connectivity

  A composite E-Line can include a vc-switched e-line.

To support the multi-domain functionality, the E-Line service template provides the VC Type parameter, which allows you to specify the type of pseudowire for the E-Line service.

Optical E-Line services

The NSD supports the creation of optical E-Line services. Using the NFM-T as an intermediary, the NSD will deploy the VPLS service, along with the required Network /Access SAPS, and the appropriate Eth-CFM settings to the 1830 PSS network element.

CaCing, Access QoS, and Network Port Bandwidth tracking are supported. Null, dot1Q, QinQ, and all LAG types can be used as Access ports. Network SAPs will be deployed as Q.* type network SAPs. E-Line service provisioning is supported on 11QPE24, 11QCE12X, and 11OPE8 cards. These cards must be of the *Provider-Bridge* variety.

Note: For 11OPE8 cards, backplane switching M ports are not supported. Only X and C ports are valid E-Line SAPs.

Deploying these services requires that service templates be deployed via NFM-T. The two possible template types are:

- **EPL template** — to be used when access ports are of encapsulation type *null*
- **EVPL template** — to be used when access ports are of encapsulation type *dot1Q* and/or *QinQ*

Note: Access ports of encapsulation type *null* should not be mixed with access ports of encapsulation type *dot1Q*.

Deploying a VPLS services also requires that a Customer be created on the NFM-T server. By default, the NSD is configured to use templates with the following names:

EVPL: "NSP_EVPL_E-Line"

EPL: "NSP_EPL_E-Line"

Customer Name: "nsp"

Note: Unless the system.conf file is modified to permit alternatives, these naming conventions must be used.

Brownfield E-Line services

E-Line services created within the NFM-P can be managed by the NSD. In order for the NSD to discover these services, their "NSD-managed" flag must be enabled within the NFM-P. Once discovered by the NSD, these services will function the same as E-Line services created within the NSD itself, provided that they meet the NSD requirements. Any change made to these services within NFM-P after discovery will be propagated to the NSD, provided the change impacts the topology of the service.

Note: E-Line services created within the NFM-P have an “Auto-delete” flag. When enabled, services without service sites are automatically deleted. This flag should not be enabled on services being managed by the NSD, as the “NSD-managed” flag is disabled upon service deletion, and remains so even if the service is recreated and resynchronized into the NSD.

E-Line multi-vendor support

The NSD supports the following multi-vendor endpoint combinations for E-Line services:

- Cisco-Nokia
- Juniper-Nokia
- Cisco-Juniper
- Cisco-Cisco
- Juniper-Juniper

Note: Cisco LSP names must be in the format of Tunnel<*number*>, where <*number*> is an integer between 0 and 65535.

Note: Standby paths are not supported by Cisco or Juniper, only secondary paths. Therefore, in instances where Cisco or Juniper endpoints are used and the Tunnel Creation Template has the Protection Type set to *Standby*, secondary paths will be created instead.

Note: Cisco LSP-Path Bindings contain a property called Path Option. This property will be set to 1 for primary and 2 for secondary.

When creating an E-Line service on multi-vendor nodes, the NSD will attempt to find a tunnel based on the criteria specified in the Tunnel Selection Profile (TSP). If no tunnel exists, and the TSP specifies that new tunnels should be created, the NSD will create MPLS RSVP-TE tunnels, including the Dynamic LSP and LSP-Path Bindings.

7.1.5 C-Line services

C-Line services connect two SAPs that can be defined on SONET/SDH, DS3/E3,T1/E1 ports or TDM channels. The NSD supports the creation of C-Line services over IP and optical networks. Whether IP or optical, when a C-Line service is deployed, the selection of the endpoints automatically utilizes the requisite technology (MPLS or L0 WDM) tunnels.

It is possible for C-Line services to use service tunnels that were not created using the NSD.

Note: Policies for service-to-tunnel binding dictate the rules associated with the service binding. If no service tunnel meets all the constraints, and this is a new C-Line service, a new service tunnel is created.

Other parameters of the C-Line service are obtained from the specific templates referenced in the abstract API definition. The service definition in the abstract API, the detailed configuration in the service templates, and other network and tunnel parameters form the complete service definition, which is represented in the normalized model for C-Line. Specific configurations based on the devices are then constructed and deployed using the NFM-P.

For information about provisioning C-Line services from the Service Fulfillment application, see 7.5 “To provision C-Line services” (p. 110).

Note: The SAP-to-SAP C-Line services can be provisioned if different ports are used for each endpoint.

Brownfield C-Line services

C-Line services created within the NFM-P can be managed by the NSD. In order for the NSD to discover these services, their "NSD-managed” flag must be enabled within the NFM-P. Once discovered by the NSD, these services function the same way as C-Line services created within the NSD, provided that they meet the NSD requirements. Any change made to these services within NFM-P after discovery is propagated to the NSD if the change impacts the topology of the service.

Note: The C-Line services created within the NFM-P have an “Auto-delete” flag. When enabled, services without service sites are automatically deleted. This flag must not be enabled on services managed by the NSD, as the “NSD-managed” flag is disabled upon service deletion, and remains so even if the service is recreated and resynchronized into the NSD.

C-Line NE support

For C-Line creation, the NSD supports the 7x50 and 7705 NE types. Third-party vendor NEs are not supported.

VC types

The C-Line service creation requires you to specify a type of VC (pseudowire). The options are:

- SAToP T1 (unstructured DS1)
- SAToP E1 (unstructured E1)
- CESoPSN (structured)
- CESoPSN CAS (structured with CAS)

7.1.6 E-LAN services

E-LAN services are configured with the same parameters that are used for E-Line service creation. Objectives/constraints are enforced for the LSPs. The default endpoint QoS template is applied to all endpoints. Zero bandwidth is reserved in the core. E-LAN services can use service tunnels that were not created using the Service Fulfillment application.

For information about provisioning E-LAN services from the Service Fulfillment application, see the 7.3 “To provision E-LAN services” (p. 105) procedure.

E-LAN multi-vendor support

E-LAN services can be created on Cisco nodes. When this is done, the NSD configures a property called bridgeDomainId during site creation.

E-LAN services are not supported on Juniper nodes.

7.1.7 L3 VPN

The NSD supports the creation of L3 VPN services. L3 VPN services utilize layer 3 VRF (VPN/virtual routing and forwarding) to routing tables for each customer utilizing the service. The customer peers with the service provider router and the two exchange routes, which are placed into a routing table specific to the customer. Multiprotocol BGP (MP-BGP) is required to utilize the service.

The RD and RT is auto-generated as per policy direction and the topology type selected. Other parameters specified in the referenced template complete the service definition. Other parameters of the L3 VPN service are obtained from the specific templates referenced in the abstract API definition. The service definition in the abstract API, the detailed configuration in the service templates, and other network and tunnel parameters form the complete service definition, which is represented in the normalized model for L3

VPN. Specific configurations based on the devices are then constructed and deployed using NFM-P. L3 VPN services can use service tunnels that were not created using the Servic Fulfillment application.

Note: Before provisioning L3 VPN services using the NSD, the user must have MP-BGP configured and working between the PE nodes to support IP VPN. The Peer CE nodes also need to be well configured. Only one AS is supported per provider.

For information about provisioning L3 VPN services from the Service Fulfillment application, see the 7.6 “To provision L3 VPN services” (p. 113) procedure.

Multi-domain L3 VPN service provisioning from L2 endpoints

Multi-domain L3 VPN services from L2 metro areas are supported. These services are created between PE routers on metro areas, however, because some PE routers are not L3 capable, the NSD performs the path search across the network, from L2 metro areas to L3 core, and finds the best exiting routers from metro to core. Then, the NSD provisions L2 E-Line services on all metro areas and L3 VPN services in the core. Finally, the services are stitched together by the NSD using VLAN hand-off.

The intra-domain tunnels must be created in advance, and all metro domains are interconnected via Ethernet links (VLAN handoff) to the core. Since none of the routers on L2 metro domains are L3 VPN capable, the NSD uses this property to run the path search algorithm. This property can be set using the NSP's REST APIs.

The NSD uses L2 and L3 service templates to define the common attributes for the auto-created services. Profiles are used for QoS and the auto-assignment of L3 RD/RT. The NSD also uses the tunnel selection profile to include and exclude specific tunnels during path search. The path search objectives (such as minimizing hop or cost) and other values specific to the VPN (such as the IP addresses of the L3 access points) are defined either from the Service Fulfillment application or the NSP's REST APIs. The NSD uses the QoS CIR values to book the bandwidth on tunnels.

L3 VPN multi-vendor support

For L3VPN services, the NSD supports the RSVP-TE option, since multi-vendor nodes do not support SDP tunnels. As a result, if an L3 VPN service is created on a multi-vendor node, the NSD's algorithm will try to find or create RSVP-TE tunnels and always set the auto-bind property to RSVP-TE on the multi-vendor nodes.

7.1.8 LAG

A LAG service is an NSP construct for managing and monitoring Nokia SR-based services over underlying LAG connectivity, including multi-technology services such as IP/Ethernet or IP/Optical. The NSP supports the addition and cross-connection of ports in LAG across an IP and optical backbone. The LAG configuration is deployed manually

via CLI or an EMS. The NSD dynamically adds ports as required. LAG services may be created on Ethernet ports on IP line cards on routers or Ethernet ports on Optical line cards.

Endpoints with a valid LAG configuration are listed automatically in the endpoint list. The bandwidth specified automatically adds the necessary operational ports into the LAG. For example, if the LAG configuration is based on 1G ports and the bandwidth specified is 1.2G, then two 1G ports are automatically added. When the monitor bandwidth option is selected, the LAG service demonstrates elasticity by automatically adding and deleting ports based on the traffic demand. LAG services created on either IP or Optical line cards automatically create the OCH services using the applicable LAG service constraints. Adding and deleting ports manually, or via bandwidth monitoring, also adds and deletes the corresponding OCH services.

For information about provisioning LAG services from the Service Fulfillment application, see the 7.7 “To provision LAG services” (p. 117) procedure.

7.1.9 OCh

The NSD supports the creation of OCHk (k = 0,1,2,3) services across an optical domain.

Note: In NSP Release 2.0 R2, OCh services can only be provisioned using GMPLS configuration.

For information about provisioning OCh services from the Service Fulfillment application, see the 7.9 “To provision OCh services” (p. 119) procedure.

7.1.10 ODU

The NSD supports the creation of ODUk (k = 0,1,2,3) services across an optical domain. The endpoints and bandwidth specified automatically pick the size of the ODU container. An ODU service automatically creates the relevant OCH service, which defaults to the physical characteristics of the port. ODU muxing is supported implicitly if needed, depending on the type of selected endpoint and the OCh type supported by the OT card hosting the endpoint.

Note: In NSP Release 2.0 R2, ODU services can only be provisioned using GMPLS configuration.

For information about provisioning ODU services from the Service Fulfillment application, see the 7.8 “To provision ODU services” (p. 118) procedure.

7.2 To enable service CAC

7.2.1 Steps

1

On your NSD and NRC server, navigate to the following directory: */opt/nsp/server /tomcat/webapps/sdn/WEB-INF/config/*

2

Modify the system.config file as follows:

```
algo

{

 serviceCAC="on"

 multiVendorServiceCAC = "on"

}
```

End of steps

7.3 To provision E-LAN services

7.3.1 Steps

1

In the Service Fulfillment application, click on the Services tab, then click on the Add button. The service creation panel opens.

2

Choose E-LAN from the Service Type drop-down menu and configure the required parameters:

Parameter	Description
Service	The name of the service
Application ID	Specifies the custom Application ID for the service
Tenant	Specifies the tenant to whom the service belongs

Parameter	Description
Template	Specifies an Endpoint QoS template to apply
Path Profile	Specifies the identifier of the path profile to apply to the service
Tunnel Selection Profile	Specifies the Tunnel Selection profile to apply to the service
Admin State	Specifies the current administrative state of the service
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Objective (Optimize on)	Specifies the primary goal when identifying resources and/or paths for service creation
MTU	Specifies the MTU for the service. The range is 0 to 9194.
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
Topology	Specifies the type of topology used for the service

3

Use the Add and Remove buttons on the service creation form to add or remove endpoints for the service. For each endpoint, configure the required parameters:

Parameter	Description
End Point	Specifies the endpoint to be used for the service
Admin State	Specifies the current administrative state of the endpoint
Inner VLAN ID	Specifies the inner tag. Applicable to Dot1Q or QinQ ports.

Parameter	Description
Outer VLAN ID	Specifies the outer tag. Applicable to Dot1Q or QinQ ports.
QoS Profile	Specifies the Generic QoS Profile to be used

4

Perform one of the following:

a. If a Generic Qos Profile was selected in the previous step, expand the Ingress and Egress tabs in each endpoint panel and configure the required parameters:

Parameter	Description
Scheduler CIR	Specifies the scheduler CIR override in either kbps or percentage (0..100), where -3 is sum of CIR, -2 is no override, and -1 is maximum.
Scheduler PIR	Specifies the scheduler PIR override in either kbps or percentage (1..100), where -2 is no override, and -1 is maximum.
Scheduler Type	Specifies the type of scheduler override. Default is Scheduler.

b. If a Generic Qos Profile was not selected in the previous step, expand the Ingress and Egress tabs in each endpoint panel and configure the required parameters:

Parameter	Description
CIR	Specifies the Committed Information Rate in Kbps.
PIR	Specifies the Peak Information Rate in Kbps.
CBS	Specifies the Committed Burst Size in KB.
MBS	Specifies the Maximum Burst Size in KB.

5

Click Submit. The E-LAN service is created.

END OF STEPS

7.4 To provision E-Line services

7.4.1 Steps

1

In the Service Fulfillment application, click on the Services tab and click on the Add button. The service creation panel opens.

2

Choose E-Line from the Service Type drop-down menu and configure the required parameters:

Parameter	Description
Service	The name of the service
Application ID	Specifies the custom Application ID for the service
Tenant	Specifies the tenant to whom the service belongs
Template	Specifies an Endpoint QoS template to apply
Path Profile	Specifies the identifier of the path profile to apply to the service. Only applicable to Optical E-Line services
Tunnel Selection Profile	Specifies the Tunnel Selection profile to apply to the service
Admin State	Specifies the current administrative state of the service
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Objective (Optimize on)	Specifies the primary goal when identifying resources and/or paths for service creation
MTU	Specifies the MTU for the service. The range is 0 to 9194
Max Hops (Span)	Specifies the maximum number of hops to consider

Parameter	Description
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider

3

If required, enable Schedule BW Modification and configure the required parameters. See 9.2 “To schedule bandwidth modification tasks” (p. 147) for more information.

4

Configure the required parameters for both Endpoint 1 and Endpoint 2:

Parameter	Description
Endpoint	The name of the Endpoint QoS template
Admin State	Specifies the current administrative state of the endpoint
Inner VLAN ID	Specifies the inner tag. Applicable to Dot1Q or QinQ ports
Outer VLAN ID	Specifies the outer tag. Applicable to Dot1Q or QinQ ports
QoS Profile	Specifies the Generic QoS Profile to be used

5

Perform one of the following:

a. If a Generic Qos Profile was selected in the previous step, expand the Ingress and Egress tabs in each endpoint panel and configure the required parameters:

Parameter	Description
Scheduler CIR	Specifies the scheduler CIR override in either kbps or percentage (0..100), where -3 is sum of CIR, -2 is no override, and -1 is maximum.

Parameter	Description
Scheduler PIR	Specifies the scheduler PIR override in either kbps or percentage (1..100), where -2 is no override, and -1 is maximum.
Scheduler Type	Specifies the type of scheduler override. Default is Scheduler.

b. If a Generic Qos Profile was not selected in the previous step, expand the Ingress and Egress tabs in each endpoint panel and configure the required parameters:

Parameter	Description
CIR	Specifies the Committed Information Rate in Kbps.
PIR	Specifies the Peak Information Rate in Kbps.
CBS	Specifies the Committed Burst Size in KB.
MBS	Specifies the Maximum Burst Size in KB.

6

Click Submit. The E-Line service is created.

END OF STEPS

7.5 To provision C-Line services

7.5.1 Steps

1

In the Service Fulfillment application, click on the Services tab and click on the Add button. The service creation panel opens.

2

Choose C-Line from the Service Type drop-down menu and configure the required parameters:

Parameter	Description
Service	The name of the service

Parameter	Description
Application ID	Specifies the custom Application ID for the service
Tenant	Specifies the tenant to whom the service belongs
Template	Specifies an Endpoint QoS template to apply to the service
Tunnel Selection Profile	Specifies the Tunnel Selection profile to apply to the service
VC Type	Specifies the type of pseudowire for the C-Line service
Include RTP header	Enables the inclusion of CEM RTP across the IP/MPLS core network
Admin State	Specifies the current administrative state of the service
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Objective (Optimize on)	Specifies the primary goal when identifying resources and/or paths for service creation
MTU	Specifies the MTU for the service. The range is 0 to 9194
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider

3

Configure the required parameters for both Endpoint 1 and Endpoint 2:

Parameter	Description
Endpoint	The name of the Endpoint QoS template

Parameter	Description
Admin State	Specifies the current administrative state of the endpoint
Inner VLAN ID	Specifies the inner tag. Applicable to Dot1Q or QinQ ports
Outer VLAN ID	Specifies the outer tag. Applicable to Dot1Q or QinQ ports
QoS Profile	Specifies the Generic QoS Profile to be used

4

Perform one of the following:

a. If a Generic Qos Profile was selected in the previous step, expand the Ingress and Egress tabs in each endpoint panel and configure the required parameters:

Parameter	Description
Scheduler CIR	Specifies the scheduler CIR override in either kbps or percentage (0..100), where -3 is sum of CIR, -2 is no override, and -1 is maximum.
Scheduler PIR	Specifies the scheduler PIR override in either kbps or percentage (1..100), where -2 is no override, and -1 is maximum.
Scheduler Type	Specifies the type of scheduler override. Default is Scheduler.

b. If a Generic Qos Profile was not selected in the previous step, expand the Ingress and Egress tabs in each endpoint panel and configure the required parameters:

Parameter	Description
CIR	Specifies the Committed Information Rate in Kbps.
PIR	Specifies the Peak Information Rate in Kbps.
CBS	Specifies the Committed Burst Size in KB.
MBS	Specifies the Maximum Burst Size in KB.

5

Click Submit. The C-Line service is created.

END OF STEPS

7.6 To provision L3 VPN services

7.6.1 Steps

1

In the Service Fulfillment application, click on the Services tab, then click on the Add button. The service creation panel opens.

2

Choose L3 VPN from the Service Type drop-down menu and configure the required parameters:

Parameter	Description
Service	The name of the service
Application ID	Specifies the custom Application ID for the service
Tenant	Specifies the tenant to whom the service belongs
Template	Specifies an Endpoint QoS template to apply
Path Profile	Specifies the identifier of the path profile to apply to the service
Tunnel Selection Profile	Specifies the Tunnel Selection profile to apply to the service
Admin State	Specifies the current administrative state of the service
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Objective (Optimize on)	Specifies the primary goal when identifying resources and/or paths for service creation
Encryption	Specifies whether or not IP VPN encryption is enabled

Parameter	Description
Auto Bind	Specifies the type of tunnel auto bind used for the service
MTU	Specifies the MTU for the service. The range is 0 to 9194
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
Topology	Specifies the type of topology used for the service

Note: The IP topology data of an L3 VPN service provisioned with the Full Mesh topology type may contain an unpopulated list of group connections (or no list of group connections at all), but cannot contain a list of group connections that are themselves empty.
The IP topology data of an L3 VPN service provisioned with the Hub-and-Spoke topology type must contain a list of one group connection with a pair of valid group names.

3

Use the Add and Remove buttons on the service creation form to add or remove endpoints for the service. For each endpoint, configure the required parameters:

Parameter	Description
End Point	Specifies the endpoint to be used for the service
Interface Name	The name of the service interface. Maximum of 32 characters. Must begin with a letter.
Admin State	Specifies the current administrative state of the endpoint
Inner VLAN ID	Specifies the inner tag. Applicable to Dot1Q or QinQ ports
Outer VLAN ID	Specifies the outer tag. Applicable to Dot1Q or QinQ ports

Parameter	Description
Primary IP	Specifies the primary IP address assigned to the service loopback endpoint
QoS Profile	Specifies the Generic QoS Profile to be used

Note: LAGs can be specified as L3 VPN service endpoints, however, ports associated with these LAGs will not be available to the service.

4

Use the Add or Remove buttons in each endpoint panel to add or remove secondary IP addresses for the endpoint. For each secondary IP address, configure the required parameter:

Parameter	Description
IP Addresses	Specifies the secondary IP addresses assigned to the service endpoint

5

Use the Add or Remove buttons in each endpoint panel to add or remove routes for the endpoint. For each route, configure the required parameters:

Parameter	Description
Static Route	Specifies the destination network IP address and subnet mask
Next Hop	Specifies the IP address of the next hop
Preference	Specifies the preference of this route. The default is 5. The range is 1 to 255.

Note: Only one endpoint can exist per site.

6

Use the Add or Remove buttons in each endpoint panel to add or remove eBGP Peers for the endpoint. For each route, configure the required parameters:

Parameter	Description
Peer IP	Specifies the IP address of the eBGP peer.

Parameter	Description
Peer AS	Specifies the Autonomous System number for the eBGP peer.

Note: Only one endpoint can exist per site.

7

Perform one of the following:

a. If a Generic Qos Profile was selected in Step 3, expand the Ingress and Egress tabs in each endpoint panel and configure the required parameters:

Parameter	Description
Scheduler CIR	Specifies the scheduler CIR override in either kbps or percentage (0..100), where -3 is sum of CIR, -2 is no override, and -1 is maximum.
Scheduler PIR	Specifies the scheduler PIR override in either kbps or percentage (1..100), where -2 is no override, and -1 is maximum.
Scheduler Type	Specifies the type of scheduler override. Default is Scheduler.

b. If a Generic Qos Profile was not selected in Step 3, expand the Ingress and Egress tabs in each endpoint panel and configure the required parameters:

Parameter	Description
CIR	Specifies the Committed Information Rate in Kbps.
PIR	Specifies the Peak Information Rate in Kbps.
CBS	Specifies the Committed Burst Size in KB.
MBS	Specifies the Maximum Burst Size in KB.

8

Click Submit. The L3 VPN service is created.

END OF STEPS

7.7 To provision LAG services

7.7.1 Steps

1

In the Service Fulfillment application, click on the Services tab, then click on the Add button. The service creation panel opens.

2

Choose LAG from the Service Type drop-down menu and configure the required parameters:

Parameter	Description
Service	The name of the service
Application ID	Specifies the custom Application ID for the service
Tenant	Specifies the tenant to whom the service belongs
Template	Specifies an Endpoint QoS template to apply
Path Profile	Specifies the identifier of the path profile to apply to the service
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Objective (Optimize on)	Specifies the primary goal when identifying resources and/or paths for service creation
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
BW (Mbps)	Specifies the bandwidth required for the service
Monitor Bandwidth	Specifies whether or not to monitor bandwidth

3

Choose a starting and terminating endpoint for the service and click Submit. The LAG service is created.

End of steps

7.8 To provision ODU services

7.8.1 Steps

1

In the Service Fulfillment application, click on the Services tab, then click on the Add button. The service creation panel opens.

2

Choose ODU from the Service Type drop-down menu and configure the required parameters:

Parameter	Description
Service	The name of the service
Application ID	Specifies the custom Application ID for the service
Group	Specifies the identifier of the group to which this service belongs
Tenant	Specifies the tenant to whom the service belongs
Template	Specifies an Endpoint QoS template to apply
Protection Type	Specifies the provisioned protection type of the connection
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Objective (Optimize on)	Specifies the primary goal when identifying resources and/or paths for service creation
Modulation	Specifies the modulation scheme for the optical signal

Parameter	Description
Phase Encoding	Specifies the encoding type of the optical signal
Wave shape	Specifies the shape of the optical signal to be used
Service Rate	Specifies the bandwidth required for the service
Max Latency (µ secs)	Specifies the maximum latency to consider

3

Choose a starting and terminating endpoint for the service and click Submit. The ODU service is created.

Note: If YCABLE protection type was selected in Step 2, two pairs of Working and Protection endpoints must be chosen.

END OF STEPS

7.9 To provision OCh services

7.9.1 Steps

1

In the Service Fulfillment application, click on the Services tab, then click on the Add button. The service creation panel opens.

2

Choose OCh from the Service Type drop-down menu and configure the required parameters:

Parameter	Description
Service	The name of the service
Application ID	Specifies the custom Application ID of the service
Group	Specifies the identifier of the group to which this service belongs
Tenant	Specifies the tenant to whom the service belongs

Parameter	Description
Template	Specifies an Endpoint QoS template to apply
Path Profile	Specifies the identifier of the path profile to apply
Protection Type	Specifies the provisioned protection type of the connection
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Objective (Optimize on)	Specifies the primary goal when identifying resources and/or paths for service creation
Channel	Specifies the Flex Grid channel to be used
Restoration	Specifies the recovery technique of a path after failure
Modulation	Specifies the modulation scheme for the optical signal
Phase Encoding	Specifies the encoding type of the optical signal
Wave Shape	Specifies the shape of the optical signal to be used
Reversion Mode	Specifies how the switch from the recovery path to the previously-failed path occurs
Facility	Specifies the port facility number to be used
Max Latency (µ secs)	Specifies the maximum latency to consider
Explicit route	Specifies whether or not to use an explicit route

3

If the Explicit route checkbox was enabled in Step 2, perform the following:

a. In the Working Path panel, choose a link from the Include Links drop-down menu

and click on the Add button to include that link in the working path. Repeat as required.

b. In the Working Path panel, choose a link from the Exclude Links drop-down menu and click on the Add button to exclude that link from the working path. Repeat as required.

c. In the Protected Path panel, choose a link from the Include Links drop-down menu and click on the Add button to include that link in the protected path. Repeat as required.

d. In the Protected Path panel, choose a link from the Exclude Links drop-down menu and click on the Add button to exclude that link from the protected path. Repeat as required.

e. As required, select any previously-chosen link from a list of included or excluded links and click on the Delete button to remove that link from the list.

f. Click on the OK button. The Explicit Route form closes.

4

Choose a starting and terminating endpoint for the service and click Submit. The OCh service is created.

END OF STEPS

Service management

7.10 Service management description

7.10.1 Introduction

This section describes how to perform management task on existing services and service components. The managements tasks include modifying and deleting a service, as well as modifying a service component, such as an endpoint or a service tunnel.

7.11 To modify services using the Service Fulfillment application

7.11.1 Purpose

Created services are displayed on the All Services tab of the Service Fulfillment application. Use this procedure to perform service-related tasks, view service-related resources, and perform service management tasks from the contextual menu that is available from the All Services tab.

7.11.2 Steps

1

Click on the All Services tab and right-click on a service. The contextual menu appears.

2

Choose one of the following options:

a. View Service Map

The service is displayed on a service topology map.

b. Overlay Resources

The service is overlaid on the topology map.

c. View in Service Supervision

The service is displayed in the NFM-P's Service Supervision application.

d. Delete Service

The service is deleted.

e. Edit Service

The service configuration form opens.

f. Manage Tenants

The Resource Tenancy Management form opens. See the 6.2 “To manage tenants” (p. 92) procedure for more information.

END OF STEPS

7.12 To manage service tunnel bandwidth

7.12.1 Bandwidth management

You can modify the parameters of a discovered brownfield service tunnel in the Service Fulfillment application. This enables you to support services with bandwidth booking in the core and to restrict or to allow for consumption, modification and deletion in a different way from how the service tunnels were discovered.

Note: The brownfield service tunnel are tunnels created previously in the NFM-P that you can discover and then use with services created in the NSD and NRC.

7.12.2 Bandwidth management parameters

You can manage the service tunnel bandwidth by modifying the values of the following parameters:

- Available Bandwidth

  This parameter allows you to set how much bandwidth is available to the NSD and NRC to use on a brownfield service tunnel. Then you must use the same bandwidth values when creating a service that uses the service tunnel. When the service is deleted, the available bandwidth on the service tunnel reverts to the previous value.

- Consumable

  This parameter controls whether the tunnel can be used or not for creating services in the NSD and NRC. By default, all greenfield and brownfield service tunnels have the Consumable parameter enabled. Disable the Consumable parameter to prevent the services created in the NSD and NRC from using the service tunnel.

- Auto Modifiable

  This parameter allows you to give the NSD and NRC full control of the available bandwidth on the service tunnel. When you enable the Auto Modifiable parameter, the NSD and NRC calculates the available bandwidth automatically and, as a result, the Available Bandwidth parameter is not modifiable anymore. Now the NSD and NRC treat the brownfield service tunnel as a green field tunnel, except the tunnel cannot be deleted in the NSD and NRC.

7.12.3 Steps

Note: You must perform the bandwidth management tasks on the service tunnel for both tunnel directions.

1

In the Service Fulfillment application, click on the Service Tunnels tab, and then right-click on a service tunnel and choose Manage Service Tunnel. The Manage Service Tunnel form opens.

2

Modify the bandwidth management parameters of the service tunnel, as required.

3

Click Submit. The service tunnel modifications are saved.

End of steps

8 Templates and policies

8.1 Template and policy provisioning

8.1.1 Introduction

This chapter describes the templates and policies that can be created using the Policy Management application.

For information about provisioning templates and policies using the NSP's REST APIs, see the *NSP API Programmer Guide*

8.1.2 E-Line and C-Line service templates

The NSD and NRC modules support the creation of E-Line and C-Line service templates. The configuration of these templates can be applied to the E-Line or C-Line service creation form in the Service Fulfillment application, thereby simplifying service provisioning. If an E-Line or C-Line service uses a template that specifies the same parameters as those specified via the NSP's REST APIs or the E-Line or C-Line service creation form in the Service Fulfillment application, then the template parameters in are overridden.

For information about provisioning E-Line service templates from the Policy Management application, see 8.2 “To provision E-Line service templates” (p. 129) and 8.4 “To provision C-Line service templates” (p. 131).

8.1.3 E-LAN service templates

The NSD and NRC modules support the creation of E-LAN service templates. The configuration of these templates can be applied to the Service Fulfillment application's E-LAN service creation form, thereby simplifying service provisioning. If an E-LAN service uses a template that specifies the same parameters as those specified via the NSP's REST APIs or the Service Fulfillment application's E-LAN service creations form, the template's parameters are overridden.

For information about provisioning E-LAN Service templates from the Policy Management application, see the 8.3 “To provision E-LAN service templates” (p. 130) procedure.

8.1.4 OCh service templates

The NSD and NRC modules support the creation of OCh service templates. The configuration of these templates can be applied to the Service Fulfillment application's

OCh service creation form, thereby simplifying service provisioning. If an OCh service uses a template that specifies the same parameters as those specified via the NSP's REST APIs or the Service Fulfillment application's Och Service creation form, the template's parameters are overridden.

For information about provisioning OCh Service templates from the Policy Management application, see the 8.5 “To provision OCH service templates” (p. 132) procedure.

8.1.5 ODU Service templates

The NSD and NRC support the creation of ODU service templates. The configuration of these templates can be applied to the Service Fulfillment application's ODU service creation form, thereby simplifying service provisioning. If an ODU service uses a template that specifies the same parameters as those specified via the NSP's REST APIs or the Service Fulfillment application's ODU service creation form, the template's parameters are overridden.

For information about provisioning ODU Service templates from the Policy Management application, see the 8.6 “To provision ODU service templates” (p. 133) procedure.

8.1.6 LAG Service templates

The NSD and NRC support the creation of LAG service templates. The configuration of these templates can be applied to the Service Fulfillment application's LAG service creation form, thereby simplifying service provisioning. If a LAG service uses a template that specifies the same parameters as those specified via the NSP's REST APIs or the Service Fulfillment application's LAG service creation form, the template's parameters are overridden.

For information about provisioning LAG Service templates from the Policy Management application, see the 8.7 “To provision LAG service templates” (p. 134) procedure.

8.1.7 L3 VPN service templates

The NSD and NRC modules support the creation of L3 VPN Service templates. The configuration of these templates can be applied to the Service Fulfillment application's L3 VPN service creation form, thereby simplifying service provisioning. If an L3 VPN service uses a template that specifies the same parameters as those specified via the NSP's REST APIs or the Service Fulfillment application's L3 VPN service creation form, the template's parameters are overridden.

For information about provisioning L3 VPN Service templates from the Policy Management application, see the 8.8 “To provision L3 VPN service templates” (p. 135) procedure.

8.1.8 RD/RT range policy

The NSD and NRC modules support the modification of the global RD/RT Range policy. The RD/RT Range policy is a single default policy that applies to all L3 VPN services. In the future, multiple RD/RT Range policies may be supported.

For information about modifying the RD/RT Range policy from the Policy Management application, see the 8.9 “To modify the RD/RT Range policy” (p. 137) procedure.

8.1.9 Tunnel Creation template

The NSD and NRC modules support the modification of the global Tunnel Creation template. The Tunnel Creation template allows the modules to attribute specific behavior to tunnel maintenance, such as allowing the consumption of the tunnel by all services, allowing the automatic deletion of the tunnel when there are no more services attached to it, or allowing modification of tunnel parameters by services.

For information about modifying the Tunnel Creation policy from the Policy Management application, see the 8.10 “To modify the Tunnel Creation template” (p. 138) procedure.

8.1.10 Tunnel Selection policy

The NSD and NRC modules support the modification of the global Tunnel Selection policy, as well as the creation of new Tunnel Selection policies. The Tunnel Selection policy is used to influence the behavior of the algorithm when selecting, creating, or deleting tunnels.

For information about creating and modifying a Tunnel Selection policy from the Policy Management application, see the 8.11 “To provision Tunnel Selection policies” (p. 139) procedure.

Tunnel selection mechanism

The non-rule-based tunnel selection policies do not steer the algorithm to check tunnels in a particular order, but inform the algorithm of what actions are permissible on a selected tunnel candidate. Suitability of a tunnel candidate is determined based on constraints and objectives. A suitable tunnel is identified only as a candidate, as the tunnel selection policy may indicate that the tunnel is unusable if the action that the algorithm would like to apply to the tunnel is deemed impermissible by the policy.

PCC-initiated LSPs

The NSD and NRC support the creation of PCC-initiated LSPs. Whe the NSD receives a service creation request and does not have any LSPs between endpoints, the NSD sends the LSP definition and the LSP creation request to the PCC router. The PCC

sends the LSP request to the NRC-P, which uses its PCE to calculate the path. Then the NRC-P sends an LSP Path Reply to the PCC. The tunnel is created on NSD and attached to the service.

8.1.11 Endpoint QoS policies

The NSD and NRC modules support the creation of Endpoint QoS templates. For information about provisioning Endpoint QoS policies from the Policy Management application, see the 8.12 “To provision Endpoint QoS templates” (p. 141) procedure.

8.1.12 Path profile

The NSD and NRC modules support the configuration of path profile templates, which are associated to path requests by PCCs. A default path profile template can also be configured, if required. By default, path profile templates will optimize on metric. Additional behavior can be specified, such as bidirectionality for forward and reverse paths between a pair of sources or destinations, and path disjointness between two paths specifying the same profile. A path request can also contain multiple profiles. Path profile templates can also be specified on the PCC.

When a PCE request contains objects specifying constraints and objectives in addition to the path profile template, the following behavior is observed:

- If a PCC request has an associated path profile template, and also has the specific constraints (B = 1) in the METRIC object (such as bandwidth, IGP metric, and TE metric values), then the path computation will use the PCC-specified values, overriding the constraint values specified in the path profile templates.
- If a PCC request has an associated path profile template and no bounds set on the values in the METRIC object, then the default values specified in the path profile template will be used in the path calculation.

Path profile templates may be applied to both SR TE LSPs, and RSVP TE LSPs. For RSVP TE LSPs, the specification of the path profile template applies to all paths for that LSP.

8.1.13 Router ID Mapping templates

The NRC-P is able to discover and display multiple IGP instances (OSPF and ISIS), which are each discovered as a unique domain. These domains are interconnected on the same routers, which themselves have multiple instances defined. If the Router IDs for these instances are the same, they will be displayed as a single router on the Service Fulfillment application's multi-domain topology maps. If the Router IDs are different, a Router ID Mapping template must be provisioned in order for the instances to be displayed as a single router on the Service Fulfillment application's multi-domain topology maps.

For more information about provisioning Router ID Mapping templates from the Policy Management application, see the 8.15 “To provision Router ID Mapping templates” (p. 144) procedure.

8.2 To provision E-Line service templates

8.2.1 Steps

1

In the Policy Management application, choose E-Line Service from the Template /Policy Type drop-down menu and click Add New. The E-Line Service Template form opens.

2

Configure the required parameters:

Parameter	Description
Name	The name of the E-Line Service template
Admin State	Specifies the administrative state required for the service
Tunnel Selection Template	Specifies a Tunnel Selection template to apply
Endpoint QoS Template	Specifies an Endpoint QoS template to apply
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Optimize on (Objective)	Specifies the primary goal when identifying resources and/or paths for service creation
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
Monitor Bandwidth	Specifies whether or not to monitor bandwidth

Parameter	Description
MTU	Specifies the MTU for the service. The range is 0 to 9194
Description	Describes the E-Line service template
VC Type	Specifies the type of pseudowire for the E-Line service template.

3

Click Submit. The E-Line Service Template form closes.

END OF STEPS

8.3 To provision E-LAN service templates

8.3.1 Steps

1

In the Policy Management application, choose E-LAN Service from the Template /Policy Type drop-down menu and click Add New. The E-LAN Service Template form opens.

2

Configure the required parameters:

Parameter	Description
Name	The name of the E-LAN Service template
Admin State	Specifies the administrative state required for the service
Tunnel Selection Template	Specifies a Tunnel Selection template to apply
Endpoint QoS Template	Specifies an Endpoint QoS template to apply
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined

Parameter	Description
Optimize on (Objective)	Specifies the primary goal when identifying resources and/or paths for service creation
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
Monitor Bandwidth	Specifies whether or not to monitor bandwidth
MTU	Specifies the MTU for the service. The range is 0 to 9194
Description	Describes the E-LAN service template
VC Type	Specifies the type of pseudowire for the E-LAN service template.

3

Click Submit. The E-LAN Service Template form closes.

End of steps

8.4 To provision C-Line service templates

8.4.1 Steps

1

In the Policy Management application, choose C-Line Service from the Template /Policy Type drop-down menu and click Add New. The C-Line Service Template form opens.

2

Configure the required parameters:

Parameter	Description
Name	The name of the C-Line Service template

Parameter	Description
Admin State	Specifies the administrative state required for the service
Tunnel Selection Template	Specifies a Tunnel Selection template to apply
BiDirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Optimize on (Objective)	Specifies the primary goal when identifying resources and/or paths for service creation
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
MTU	Specifies the MTU for the service. The range is 0 to 9194
Description	Describes the C-Line Service template
VC Type	Specifies the type of pseudowire for the C-Line service.

3

Click Submit. The C-Line Service Template form closes.

End of steps

8.5 To provision OCH service templates

8.5.1 Steps

1

In the Policy Management application, choose OCH Service from the Template /Policy Type drop-down menu and click Add New. The OCH Service Template form opens.

2

Configure the required parameters:

Parameter	Description
Name	The name of the OCH Service template
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Optimize on (Objective)	Specifies the primary goal when identifying resources and/or paths for service creation
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
Restoration	Specifies the recovery technique of a path after failure
Reversion Mode	Specifies how the switch from the recovery path to the previously-failed path occurs
Description	Describes the OCH Service template

3

Click Submit. The OCH Service Template form closes.

END OF STEPS

8.6 To provision ODU service templates

8.6.1 Steps

1

In the Policy Management application, choose ODU Service from the Template /Policy Type drop-down menu and click Add New. The ODU Service Template form opens.

2

Configure the required parameters:

Parameter	Description
Name	The name of the ODU Service template
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Optimize on (Objective)	Specifies the primary goal when identifying resources and/or paths for service creation
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
Bandwidth (Mbps)	Specifies the bandwidth required for the service
Reverse Bandwidth (Mbps)	Specifies the bandwidth required for the returning path of the service
Description	Describes the ODU Service template

3

Click Submit. The ODU Service Template form closes.

END OF STEPS

8.7 To provision LAG service templates

8.7.1 Steps

1

In the Policy Management application, choose LAG Service from the Template/Policy Type drop-down menu and click Add New. The LAG Service Template form opens.

2

Configure the required parameters:

Parameter	Description
Name	The name of the LAG Service template
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Optimize on (Objective)	Specifies the primary goal when identifying resources and/or paths for service creation
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
Bandwidth (Mbps)	Specifies the bandwidth required for the service
Reverse Bandwidth (Mbps)	Specifies the bandwidth required for the returning path of the service
Monitor Bandwidth	Specifies whether or not to monitor bandwidth
Description	Describes the LAG Service template

3

Click Submit. The LAG Service Template form closes.

END OF STEPS

8.8 To provision L3 VPN service templates

8.8.1 Steps

1

In the Policy Management application, choose L3 VPN Service from the Template /Policy Type drop-down menu and click Add New. The L3 VPN Service Template form opens.

2

Configure the required parameters:

Parameter	Description
Name	The name of the L3 VPN Service template
Admin State	Specifies the administrative state required for the service
Tunnel Selection Template	Specifies a Tunnel Selection template to apply
Endpoint QoS Template	Specifies an Endpoint QoS template to apply
Bidirectional	Specifies whether or not a return path is required, and if so, what type of return path should be determined
Optimize on (Objective)	Specifies the primary goal when identifying resources and/or paths for service creation
Max Hops (Span)	Specifies the maximum number of hops to consider
Max Latency (µ secs)	Specifies the maximum latency to consider
Max Cost	Specifies the maximum cost to consider
Auto Bind	Specifies the type of autobind to be used for the service
MTU	Specifies the MTU for the service. The range is 0 to 9194
Description	Describes the L3 VPN Service template

3

Click Submit. The L3 VPN Service Template form closes.

END OF STEPS

8.9 To modify the RD/RT Range policy

8.9.1 Steps

1

In the Policy Management application, choose RD/RT Range from the Template /Policy Type drop-down menu. The RD/RT Range policy appears in the list below.

2

Right click on the RD/RT Range policy and choose Edit from the contextual menu. The RD/RT Range Policy form opens.

3

Configure the required parameters:

Parameter	Description
Name	The name of the RD/RT Range policy
Description	Describes the RD/RT Range policy

4

Configure the required parameters in both the Route Target (RT) and Route Distinguisher (RD) panels:

Parameter	Description
Use Provider AS	Specifies whether or not the NSP will use the AS number from the provider's network configuration. If confederation is used, the confederation AS will be used.
Assigned Number (Min)	Specifies the minimum assigned number. For type 0, the value must be between 0 and 4294967295, inclusive. For Type 2, the value must be between 0 and 65535, inclusive
Assigned Number (Max)	Specifies the maximum assigned number. For type 0, the value must be between 0 and 4294967295, inclusive. For Type 2, the value must be between 0 and 65535, inclusive.

5

Click Submit. The RD/RT Range Policy form closes.

END OF STEPS

8.10 To modify the Tunnel Creation template

8.10.1 Steps

1

In the Policy Management application, choose Tunnel Creation from the Template /Policy Type drop-down menu. The Tunnel Creation template appears in the list below.

2

Right click on the Tunnel Creation template and choose Edit from the contextual menu. The Tunnel Creation Template form opens.

3

Configure the required parameters:

Parameter	Description
Name	The name of the Tunnel Creation template
Consumable	Specifies whether or not new services can ride this tunnel
Auto Deletable	Specifies whether or not the tunnel is deleted when the last service is removed from it
Auto Modifiable	Specifies whether or not the tunnel parameters are modifiable due to changes in the services that are riding it
Protected	Specifies whether or not the tunnel has a protection path. The protection path is only signaled after the primary path fails
Description	Describes the Tunnel Creation template

4

If the Protected parameter was enabled in the previous step, configure the following parameter:

Parameter	Description
Protection Type	Specifies the path protection type. The protection path is pre-signaled and available for immediate recovery after a primary path failure

5

Click Submit. The Tunnel Creation Template form closes.

END OF STEPS

8.11 To provision Tunnel Selection policies

8.11.1 Steps

1

In the Policy Management application, choose Tunnel Selection from the Template /Policy Type drop-down menu. The default Tunnel Selection policy appears in the list below.

2

Perform one of the following:

a. Click on the Add New button. The Tunnel Selection Policy form opens.

b. Right click on the default Tunnel Selection policy and choose Edit from the contextual menu. The Tunnel Selection Policy form opens.

3

Configure the required parameters:

Parameter	**Description**
Name	The name of the Tunnel Selection policy
Description	Describes the Tunnel Selection policy
Use existing tunnels	Specifies whether or not services can ride on top of previously-created NSP tunnels

Parameter	Description
Expand existing tunnels	Specifies whether or not services can grow previously-created NSP tunnels without forcing a re-route
Redirect existing tunnels	Specifies whether or not services can force a re-route of previously-created NSP tunnels
Create new tunnels	Specifies whether or not services can create new tunnels
PCC Initiated LSP	Specifies whether or not services can use a tunnel selection profile to achieve a PCC-initiated LSP configuration
Avoid operational state down	Specifies whether or not the tunnel should avoid routers that are operationally down
Strict RSVP	Specifies the priority level for Strict RSVP LSPs
Loose RSVP	Specifies the priority level for Loose RSVP LSPs
BGP	Specifies the priority level for BGP tunnels
LDP	Specifies the priority level of LDP tunnels
GRE	Specifies the priority level of GRE tunnels

4

Click on the Add and Remove buttons to either include or exclude Steering Parameters from the Tunnel Selection policy.

5

Click Submit. The Tunnel Selection Policy form closes.

END OF STEPS

8.12 To provision Endpoint QoS templates

8.12.1 Steps

1

In the Policy Management application, choose Endpoint QoS from the Template /Policy Type drop-down menu and click Add New. The Endpoint QoS Template form opens.

2

Configure the required parameters:

Parameter	Description
Name	The name of the Endpoint QoS template
Description	Describes the Endpoint QoS template
QoS Profile	Specifies the Generic QoS Profile to be used
QoS Profile Description	Describes the Generic QoS Profile that is in use

3

If a Generic Qos Profile was selected in the previous step, expand the Ingress and Egress tabs and configure the required parameters:

Parameter	Description
Scheduler CIR	Specifies the scheduler CIR override in either kbps or percentage (0..100), where -3 is sum of CIR, -2 is no override, and -1 is maximum.
Scheduler PIR	Specifies the scheduler PIR override in either kbps or percentage (1..100), where -2 is no override, and -1 is maximum.
Scheduler Type	Specifies the type of scheduler override. Default is Scheduler.

4

With the Ingress and Egress tabs expanded, configure the remaining required parameters:

Parameter	Description
CIR (Mbps)	Specifies the Committed Information Rate in Mbps.
PIR (Mbps)	Specifies the Peak Information Rate in Mbps.
CBS (KB)	Specifies the Committed Burst Size in KB.
MBS (KB)	Specifies the Maximum Burst Size in KB.

5

Click Submit. The Endpoint QoS Template form closes.

END OF STEPS

8.13 To provision Path Profile templates

8.13.1 Steps

1

In the Policy Management application, choose Path Profile from the Template/Policy Type drop-down menu. The default Path Profile template appears in the list below.

2

Perform one of the following:

a. Click on the Add New button. The Path Profile Template form opens.

b. Right click on the default Path Profile template and choose Edit from the contextual menu. The Path Profile Template form opens.

3

Configure the required parameters:

Parameter	Description
Name	The name of the Path Profile template
Profile ID	The Profile ID of the paths to be included in path computation

Parameter	Description
Reserved Profile ID	The Path Profile template assumes the Name and role of the default Path Profile template
BiDirectional	The bidirectional mode to be used in path computation
Disjoint	The Disjoint mode to be used in path computation
Optimize on (Objective)	Specifies the primary goal when identifying paths for path computation
Max Cost	The Max Cost constraint to be used in path computation
Max Hops (Span)	The Max Hops constraint to be used in path computation
Max TE Metric	The Max TE Metric constraint to be used in path computation
Description	Describes the Path Profile template

4

Click Submit. The Path Profile Template form closes.

END OF STEPS

8.14 To create a Steering Parameter

8.14.1 Steps

1

In the Policy Management application, choose Steering Parameter from the Template/Policy Type drop-down menu and click Add New. The Steering Parameter form opens.

2

Configure the following parameter:

Parameter	Description
Name	The name of the Steering Parameter

3

Click Submit. The Steering Parameter form closes.

END OF STEPS

8.15 To provision Router ID Mapping templates

8.15.1 Steps

1

In the Policy Management application, choose Router ID Mapping from the Template /Policy Type drop-down menu. If a Router ID Mapping template was already created, it appears in the list below.

2

Perform one of the following:

a. Click on the Add New button. The Router ID Mapping Template form opens.

b. Right click on the existing Router ID Mapping template and choose Edit from the contextual menu. The Router ID Mapping Template form opens.

3

Configure the required parameters:

Parameter	Description
Name	Specifies the name of the Router ID Mapping template
System IP Address	Specifies the system IP address of the router
System Name	Specifies the router system name
PCC Address	Specifies the address of the PCC associated with the router
Description	Specifies the router description
Router Info	Specifies the network identifier, the AS number, the BGP-LD topology identifier, the router identifier, and the protocol that the IGP router is using. Click on the Add button to add additional Router Info entries, as required.

4

Click Submit. The Router ID Mapping Template form closes.

END OF STEPS

9 Bandwidth modification

9.1 Bandwidth modification scheduling

9.1.1 Introduction

The Task Scheduler application enables users to schedule bandwidth modification requests/tasks on existing E-Line services. It is assumed that the end user has already created an E-Line service through the NSD and has a valid service-Id. The user can schedule a single, or recurring bandwidth modification request for their E-Line service. After creating a scheduled task, the application allows the user to view, modify, or delete the task. In the case of modification, the user is allowed to change both the start date and the task execution intervals. The user is also able to view all of their current requests and the state of those requests (Scheduled / Running / Disabled). A historical log of all executed tasks, their status, and their results is available.

9.2 To schedule bandwidth modification tasks

9.2.1 Steps

1

In the Task Scheduler application, choose Bandwidth Modification from the Task/Job Type drop-down menu and click Add New. The New Bandwidth Modification Task form opens.

2

Configure the required parameters:

Parameter	Description
Task Name	The name of the task
Start Date	The date and time at which the Bandwidth Modification task begins
End Date	The date and time at which the Bandwidth Modification task ends
Repeats	Specifies at what interval the task repeats, if at all

3

Select a service for which to schedule a bandwidth modification task.

4

Configure the required parameters for both Endpoint 1 and Endpoint 2:

Parameter	Description
QoS Profile	Specifies the Generic QoS Profile to be used
CIR	Specifies the Committed Information Rate in Kbps.
PIR	Specifies the Peak Information Rate in Kbps.
CBS	Specifies the Committed Burst Size in KB.
MBS	Specifies the Maximum Burst Size in KB.

5

Click Submit. The bandwidth modification task is scheduled.

END OF STEPS

Glossary

A

ACL
Access Control List

API
Application Programming Interface

AS
Autonomous System

B

BGP
Border Gateway Protocol

C

C-Line
Circuit Emulation Service

CIR
Committed Information Rate

CSPF
Constrained Shortest Path First

E

E-Line
Ethernet Virtual Private Line

EMS
Element Manager System

G

GUI
Graphical User Interface

H

HTML
HyperText Markup Language

I

IP
Internet Protocol

IGP
Internal Gateway Protocol

K

KPI
Key Performance Indicator

KVM
Kernel-based Virtual Machine

L

LAG
Link Aggregation

LSP
Layered Service Provider

M

MAN
Metropolitan Area Network

MPLS
Multiprotocol Label Switching

MTU
Maximum Transmission Unit

N

NBI
Northbound Interface

NFM-P
Network Functions Manager — Packet

NFM-T
Network Functions Manager — Transport

NRC
Network Resource Controller

NRC-F
Network Resource Controller — Flow

NRC-P
Network Resource Controller — Packet

NRC-T
Network Resource Controller — Transport

NSD
Network Services Director

NSP
Network Services Platform

O

OCh
Optical Channel

ODU
Optical Data Unit

OSPF
Open Shortest Path First

OSS
Operations Support System

P

PCC
Path Computation Client

PCE
Path Computation Element

PCEP
Path Computation Element Protocol

PIR
Peak Information Rate

Q

QoS
Quality of Service

R

RAM
Random-Access Memory

RSVP
Resource Reservation Protocol

S

SDN
Software-Defined Networking

SLA
Service-Level Agreement

T

TPM
Template Provisioning Manager

U

URL
Uniform Resource Locator

V

vCPAA
Virtual Control Plane Assurance Adaptor

VLAN
Virtual Local Area Network

VPLS
Virtual Private LAN Service

VPN
Virtual Private Network

VPRN
Virtual Private Routed Network

W

WAN
Wide Area Network

Y

YANG
Yet Another Next Generation