



NSP

Network Services Platform

Release 23.11

System Architecture Guide

3HE-18993-AAAC-TQZZA
Issue 1
December 2023

© 2023 Nokia.

Use subject to Terms available at: www.nokia.com/terms

Legal notice

Nokia is committed to diversity and inclusion. We are continuously reviewing our customer documentation and consulting with standards bodies to ensure that terminology is inclusive and aligned with the industry. Our future customer documentation will be updated accordingly.

This document includes Nokia proprietary and confidential information, which may not be distributed or disclosed to any third parties without the prior written consent of Nokia.

This document is intended for use by Nokia's customers ("You"/"Your") in connection with a product purchased or licensed from any company within Nokia Group of Companies. Use this document as agreed. You agree to notify Nokia of any errors you may find in this document; however, should you elect to use this document for any purpose(s) for which it is not intended, You understand and warrant that any determinations You may make or actions You may take will be based upon Your independent judgment and analysis of the content of this document.

Nokia reserves the right to make changes to this document without notice. At all times, the controlling version is the one available on Nokia's site.

No part of this document may be modified.

NO WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF AVAILABILITY, ACCURACY, RELIABILITY, TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, IS MADE IN RELATION TO THE CONTENT OF THIS DOCUMENT. IN NO EVENT WILL NOKIA BE LIABLE FOR ANY DAMAGES, INCLUDING BUT NOT LIMITED TO SPECIAL, DIRECT, INDIRECT, INCIDENTAL OR CONSEQUENTIAL OR ANY LOSSES, SUCH AS BUT NOT LIMITED TO LOSS OF PROFIT, REVENUE, BUSINESS INTERRUPTION, BUSINESS OPPORTUNITY OR DATA THAT MAY ARISE FROM THE USE OF THIS DOCUMENT OR THE INFORMATION IN IT, EVEN IN THE CASE OF ERRORS IN OR OMISSIONS FROM THIS DOCUMENT OR ITS CONTENT.

Copyright and trademark: Nokia is a registered trademark of Nokia Corporation. Other product names mentioned in this document may be trademarks of their respective owners.

© 2023 Nokia.

Contents

About this document	5
1 NSP product overview	7
1.1 Introduction to the Network Services Platform	7
1.2 What the NSP provides	7
1.3 Related documentation	8
2 NSP product offerings	11
2.1 NSP software selection	11
2.2 Planning your deployment.....	13
2.3 Network Operations	15
2.4 Resource Control	18
2.5 Network Programming	20
3 NSP architecture fundamentals	23
3.1 System design.....	23
3.2 NSP graphical user interfaces.....	24
3.3 NSP APIs	25
3.4 NSP network mediation.....	25
3.5 Classic management architecture	26
4 Platform programmability	29
4.1 NSP programming frameworks	29
4.2 MDM framework.....	29
4.3 Workflow management framework.....	31
4.4 Intent management framework	32
4.5 Synchronization framework.....	32
4.6 Operation, administration, and maintenance (OAM) framework.....	32
4.7 RESTCONF gateway	32
4.8 IETF standard models.....	33
5 Deployment fundamentals	35
5.1 Overview	35
5.2 NSP Kubernetes implementation	36
5.3 VSR-NRC.....	37
5.4 NSP analytics servers	37
5.5 NSP Flow Collectors and Flow Collector Controllers	38

5.6	Auxiliary database	38
5.7	Classic management components	39
5.8	WS-NOC	40
6	Security architecture	41
6.1	NSP system security	41
6.2	User security and session management	42
6.3	Firewall support.....	43
6.4	NSP User Access Control	43
6.5	Activity logging	44
6.6	NFM-P system security	44
6.7	NSP software security summary	47
7	NSP communication	49
7.1	Overview	49
7.2	External NSP communication.....	50
7.3	Internal NSP communication.....	52
7.4	NFM-P communication.....	54
8	System redundancy and fault tolerance	57
8.1	Redundancy mechanisms.....	57
8.2	NSP system failure and recovery scenarios.....	59
8.3	MDM fault tolerance	60
8.4	NFM-P fault tolerance	61
8.5	VSR-NRC fault tolerance	62
8.6	NSP Flow Collector and Flow Collector Controller fault tolerance	64
8.7	Analytics server fault tolerance	65
A	NSP technology standards	67
A.1	NSP technology standards.....	67
A.2	Classic management technology standards	69

About this document

Purpose

The *NSP System Architecture Guide* describes the Network Services Platform architecture and interoperation with other systems from a high-level perspective. The audience is a technology officer, network planner, or system administrator who requires a broad technical understanding of the NSP system structure and design methodology.

Scope

The guide scope is limited to a description of the integral elements that are common to NSP components. For information about the architecture of a specific NSP component, or a product or appliance that integrates with the NSP, see the component, product, or appliance documentation.

Document support

Customer documentation and product support URLs:

- [Documentation Center](#)
- [Technical support](#)

How to comment

Please send your feedback to [Documentation Feedback](#).

1 NSP product overview

1.1 Introduction to the Network Services Platform

1.1.1 Product description

The Network Services Platform, or NSP, is a distributed system that provides comprehensive infrastructure, service management, and resource control within and among multiple network domains.

The NSP manages multi-vendor physical and virtual network infrastructure, and supports a number of management mechanisms and protocols.

A variety of interfaces are available for management functions such as programmable multi-layer service provisioning, rollout, and activation. NSP service management extends to offerings that employ multiple technologies to:

- accelerate the creation and rollout of on-demand network services
- enable real-time service optimization and flow steering
- extend assurance capabilities and automate assurance functions

1.1.2 Functional highlights

The NSP enables and automates network management using:

- an integrated suite of functions that are adaptable to your network management and service rollout use cases
- resource-control functions for real-time network control and traffic optimization
- an open, programmable platform for the automation of network operations and integration with in-house or third-party support systems

1.2 What the NSP provides

1.2.1 Service rollout and management

The NSP facilitates rapid service provisioning and activation, and provides timely service performance, usage, and fault information.

1.2.2 Traffic optimization

The NSP automates traffic management by controlling and steering flows as required.

1.2.3 Data collection and analysis

The NSP monitors network KPIs for immediate and trend-based reporting. You can view near-real-time NE KPIs, and identify network trends using NSP analytics reports that are based on collected flow statistics or NE telemetry data.

1.2.4 Network administration

You can define groups of network objects for access by specific user groups, and create policies that define NSP operator access to objects and functions.

The NSP provides the ability to create network intents and workflows that define the scope and flow of network management operations.

1.2.5 Programming

The NSP programming functions enable the design of NSP network intents and workflows to automate routine and repetitive tasks in a specifically prescribed manner. Using vendor-agnostic intent and workflow frameworks for equipment deployment and management, the NSP performs network mediation based on YANG models that support network synchronization, OAM test execution, and event reporting.

1.2.6 Inventory management

An NSP classic management deployment dynamically maintains a network equipment data store for SNMP-managed devices. See [Chapter 2, “NSP product offerings”](#) for information about NSP deployment options.

1.3 Related documentation

1.3.1 NSP platform guides

The following guides are fundamental to the planning, deployment, and administration of an NSP system, and are intended to be consumed in the order shown:

- *NSP Installation and Upgrade Guide*—describes NSP deployment options and platform configuration; includes software installation, upgrade, and integration procedures
- *NSP Planning Guide*—describes the required NSP environment and platform resources, provides network and security specifications, and includes scaling and performance guidelines
- *NSP System Administrator Guide*—describes administrative responsibilities such as NSP security, system control and configuration, NSP system configuration, and data backup and restore
- *NSP System Architecture Guide*—this guide; acquaints the reader with the NSP product functions, main system elements, and platform programmability; describes structural and design aspects that include system security, networking, and fault tolerance

1.3.2 Other NSP guides

The NSP documentation suite also includes the following:

- *NSP Release Description*—describes the features and functions in an NSP release; includes information such as functional enhancements and resolved product issues
- *NSP Release Notice*—contains special or notable information about an NSP release, such as deployment limitations or product enhancements not covered in the core documentation
- *NSP User Guide*—describes NSP access and use for operators; introduces the NSP UI functions, and describes how to obtain NSP software and documentation

-
- *function-specific management guides*—multiple guides that each describe how to use an NSP facility, for example, service management, which is described in the *NSP Service Management Guide*; example use cases are provided
 - *NSP utility help*—multiple guides that each describe the function and operation of an NSP utility; example use cases are provided
 - *NSP Glossary*—defines acronyms, initialisms, and other terms used in the NSP product interfaces and documentation

2 NSP product offerings

2.1 NSP software selection

2.1.1 Introduction

The NSP software offerings are logically organized using the following terminology:

- software suite—defines the available software for a general network management realm such as operations, resource management, or automation
- package (base package or feature package)—a defined a set of functions within a software suite that are purchased and licensed together
- installation option—enables features within a feature package during containerized NSP installation

[2.2 “Planning your deployment” \(p. 13\)](#) lists the available NSP software suites, feature packages, and installation options.

NSP software suites

An NSP software suite is a grouping of feature packages that are associated with a broad aspect of network management. Feature packages other than the mandatory NSP Platform feature package are in one of the following software suites:

- [2.3 “Network Operations” \(p. 15\)](#)
- [2.4 “Resource Control” \(p. 18\)](#)
- [2.5 “Network Programming” \(p. 20\)](#)

NSP feature packages

The NSP functions in a software suite are organized as feature packages that are selectively available for purchase. Each feature package contains a set of related functions; for example, basic or enhanced capabilities, or management capacities related to network type or scope. You can combine multiple feature packages from different software suites in an NSP deployment.

The NSP base packages are feature packages that provide core platform functionality and management features that are common across all product suites.

i **Note:** A feature package may require the functions of one or more other feature packages in the same software suite; however, no such dependencies exist between feature packages in separate software suites.

i **Note:** A feature package must be purchased to receive Nokia support for the associated NSP functions.

NSP installation options

The containerized software components associated with a feature package are installed by enabling one or more installation options in the NSP configuration during system deployment.

Selectively enabling installation options allows you to highly customize your NSP deployment and reduce the platform requirements of the deployment.

The *NSP Installation and Upgrade Guide* describes how to enable and configure installation options in a containerized NSP deployment.

2.1.2 Platform

The Platform feature package provides a core set of services that are used by all other feature packages, including authentication and authorization, databases, centralized logging and system health monitoring, service registration, and a message bus.

The Platform feature package includes the following installation options:

- Base Services
- Logging and Monitoring
- Pluggable Network Adaptation
- REST Event Forwarder

Base Services and Logging and Monitoring are enabled by default. Pluggable Network Adaptation and REST Event Forwarder are optional.

Two mediation types are included in the Platform feature package:

- Classic Mediation
- Model-Driven Mediation

You must select either one or both of these mediation types when using the Platform Sizing Tool.

Classic Mediation comprises the following functions:

- mediation services for SNMP-managed Nokia NEs via the Network Function Manager - Packet (NFM-P) component
- control-plane management services for network infrastructure management and service assurance functions via the Control Plane Assurance Manager, or CPAM

The Platform feature package also includes NSP Flow Collectors and NSP Flow Collector Controllers that direct the flow-collection activity. An NSP Flow Collector is a scalable component that collects AA Cflowd or System Cflowd statistics directly from NEs. The statistics records are stored in a database for reporting by NSP functions such as Analytics, or forwarded to one or more remote target servers for processing by in-house systems or third-party tools.

Base Services


Base Services is a mandatory installation option within the Platform feature package that provides the minimum level of services used by all other feature packages. The embedded services include:

- browser-based graphical UI that provides operator access to NSP functions
- user access, management, and monitoring functions
 - single sign-on user authentication for system access
 - role- and resource-based user authorization and access control
 - user activity tracking and logging

- system and operator resources
 - persistent storage using relational and graph databases
 - distributed event streaming
 - subscription-based event forwarding
 - REST/RESTCONF API gateway
 - central online help
- security functions such as TLS certificate monitoring and expiry management

Logging and Monitoring

Logging and Monitoring is an installation option of the Platform feature package that provides centralized collection and viewing of application logs for troubleshooting the NSP deployment. Logging and Monitoring also includes visualization of various NSP operational metrics using System Health.

 **Note:** Logging and Monitoring is enabled by default. At the direction of Nokia in rare scenarios, the option may be disabled.

You can optionally enable the forwarding of NSP log entries to a Splunk or syslog server; see “Centralized logging” in the *NSP Installation and Upgrade Guide* for information.

Pluggable Network Adaptation

Pluggable Network Adaptation is an optional Platform installation option that provides vendor-agnostic mediation services to network devices over the following management interface types:

- NETCONF
- SNMP
- CLI
- gRPC

Mediation is managed through run-time loadable adaptors that provide backward- and forward-compatible interfaces to devices as the NSP evolves. The Pluggable Network Adaptation mediation services are provided by the NSP Model-Driven Mediation (MDM) framework, and are available to other NSP functions.

REST Event Forwarder

REST Event Forwarder is an optional Platform installation option that provides functionality to manage on-the-fly notifications coming from external controllers. Enabling this option is required for managing T-BTS via Mantaray NM without installing Simplified RAN Transport.

2.2 Planning your deployment

2.2.1 Feature packages and installation options

The following table correlates the feature packages in each software suite with the installation options you can specify during NSP deployment.

Table 2-1 NSP software suites, feature packages, and installation options

Software suite	Feature package	Installation option
Base (common/mandatory)	Platform	platform-baseServices
		platform-loggingMonitoring
		platform-pluggableNetworkAdaptation
		platform-restForwarder
Network Operations	Network Infrastructure Management	networkInfrastructureManagement-basicManagement
		networkInfrastructureManagement-deviceConfig
		networkInfrastructureManagement-performanceIndicatorsAndAlerts
		networkInfrastructureManagement-igpTopology
		networkInfrastructureManagement-simplifiedRanTransport
		networkInfrastructureManagement-gnmiTelemetry
		networkInfrastructureManagement-gNMIonChange
	Service Assurance	serviceAssurance
	Service Activation and Configuration ¹	serviceActivationAndConfiguration-originalServiceFulfillment
		serviceActivationAndConfiguration-intentBasedServiceFulfillment
Network Operation Analytics	networkOperationsAnalytics-baselineAnalytics	
Transport Slice Operations	transportSliceOperations	
Resource Control	Control & Visualization Starter	controlAndVisualizationStarter
	Path Control & Optimization	pathControlAndOptimization
	Enhanced Optimization	enhancedOptimization
	Simulation	simulation
	Multi-Layer Discovery and Visualization	multilayerDiscoveryAndVisualization
	Multi-Layer Control and Coordination	multilayerControlCoordination
Network Programming	Software Development Kit	Not applicable ²
	Intent Based Networking Framework	intentBasedNetworkingFramework
	Workflow Automation Engine	workflowAutomationEngine

Notes:

1. In addition to the installation options in the table, the Service Activation and Configuration feature package also enables support of Service Manager features in classic management, which is deployed as an ancillary component and has no associated NSP installation option.
2. See the *NSP Network Automation Guide* for more information.
3. It is recommended that the networkInfrastructureManagement-igpTopology installation options is not deployed alongside any of the following installation options: serviceActivationAndConfiguration-originalServiceFulfillment, controlAndVisualizationStarter, pathControlAndOptimization, enhancedOptimization, or resourceControlAssurance. The functions of these installation options will supersede those of networkInfrastructureManagement-igpTopology.

2.3 Network Operations

2.3.1 Description

The Network Operations software suite enables operators to manage multi-vendor networks using mechanisms such as SNMP or MDM. The feature packages enable configuration, supervision, and assurance functions, and provide telemetry and analytics reporting.

The Network Operations feature packages are:

- Network Infrastructure Management
- Service Assurance
- Service Activation and Configuration
- Network Operations Analytics
- Transport Slice Operations

2.3.2 Network Infrastructure Management

Network Infrastructure Management is a mandatory feature package that provides mediation-agnostic device management. Network Infrastructure Management uses the Platform feature package with either the Base Services or Pluggable Network Adaptation installation option for communication with the network.

The Network Infrastructure Management functions are enabled using the installation options described below.

Basic Management

The Basic Management installation option includes network management capabilities such as network mediation, network infrastructure and service management, and fault management, which are spread among various NSP functions:

Basic Management also includes various KPI dashboards and provides API access to NSP functions.

The Basic Management internal services use the NSP data synchronization framework to ensure that the NSP network infrastructure view is current and consistent with the actual network configuration. The NSP OAM framework manages the execution of various infrastructure tests.

Device Config

The Device Config installation option provides zero-touch provisioning (ZTP) and infrastructure configuration management (ICM) for model-driven device management..

Performance Indicators And Alerts

The Performance Indicators and Alerts installation option enables you to define and monitor network key performance indicators (KPIs), and create threshold-crossing alerts based on the KPIs.

IGP topology

The IGP Topology installation option provides the IGP topology information that is required to populate the IGP topology map in Network Map and Health and the IGP layer in the Object Troubleshooting multi-layer map.

Simplified RAN Transport

The Simplified RAN Transport (SRT) installation option provides SRT dashboard views in the Network Health application. Specifically, the SRT-specific network health summary displays the health of T-BTS nodes and their associated radio plane bindings. See the *NSP Simplified RAN Transport Solution* for information about the installation options that are required for T-BTS management.

gnmiTelemetry

The gNMI telemetry installation option enables gNMI telemetry collection for classic and model-driven devices.

gNMI onChange

The gNMI onChange installation option enables NE mediation using gNMI.

2.3.3 Service Assurance

The Service Assurance feature package provides mediation-agnostic monitoring and troubleshooting of multi-vendor IP/MPLS and Ethernet services.

The Service Assurance functions use the NSP OAM framework to invoke and manage service-level tests, and the NSP data synchronization framework to ensure that the NSP view of the network infrastructure is consistent with the actual network configuration.

2.3.4 Service Activation and Configuration

The Service Activation and Configuration feature package licenses customers to use any one of the following:

- NSP service management
- NSP classic management component, NFM-P

The mediation-agnostic, multi-vendor service provisioning and activation features within this feature package use operator-defined policies to guide dynamic network resource selection and automated service provisioning, and can execute pre- and post-activation workflows using the NSP Workflows framework. Service Activation and Configuration also includes a service management REST API.

Service management

The NSP service management function adds the programmability of the intent management framework to service provisioning and activation. The programmability supports flexible service constructs and hot-deployable models, and extends to augmentation that supports new NSP service offerings without an NSP upgrade. Nokia provides a set of predefined service models (intent types) for a variety of services. See the *NSP Service Management Guide* for more information.

2.3.5 Network Operations Analytics

The Network Operations Analytics feature package provides a pre-integrated solution that enables operators to better understand how their network works. The feature package provides full L0-L7 analytics with detailed visualizations and also real-time analysis from hardware inventory to L2 and L3 services by collecting, warehousing and aggregating event, performance and volumetric data.

Analytics

Prepackaged reports and dashboards are integrated with the Nokia SR portfolio regardless of the mediation layer and mode the network elements are running in. Drag-and-drop and web-based tools are provided to create insightful ad-hoc reports and dashboards to enable self-service reporting. Analytics is installed via RPM.

Baseline analytics

The Baseline Analytics installation option is used to enable baselining and anomaly detection that are ideally suited for advanced event-based and closed-loop automation use cases. Baselines automatically learn the normal behavior pattern for selected network and service resources. Anomalies can be triggered when a deviation from the normal historical pattern is observed. This advanced capability can provide an early warning of unforeseen network issues and provide additional time for mitigation and resolution. Baselines are configured and visualized using Data Collection and Analysis.

2.3.6 Transport Slice Operations

Transport Slice Operations is a feature package that enables the 5G Transport Slice Controller dashboard. The dashboard provides a summary of the overall health of all transport slices in the network, as well as allowing drill-down into health views on a per-slice basis. The TSC dashboard displays details about L0/L1/L2/L3 services and tunnels/paths used during the realization of transport slices, and provides pro-active monitoring telemetry data and reports on transport slices.

i **Note:** The Service Activation and Configuration feature package is required with the Transport Slice Operations feature package in all use cases.

Depending on your use case, you may require feature packages in addition to Service Activation and Configuration:

If your deployment uses FlexAlgo - a nodal capability for optimal path selection - then no additional packages are needed.

If your deployment uses the centralized NSP PCE (NRC-P) for manual or automatic [LSP] traffic re-routing, then it requires either:

- Control and Visualization Starter FP + Path Control and Optimization FP, or
- Control and Visualization Starter FP + Path Control and Optimization FP + Enhanced Optimization FP (if your deployment uses the extra precision/options of telemetry-based optimization and/or traffic-engineered ECMP)

2.4 Resource Control

2.4.1 Description

The Resource Control software suite provides IP path control, visualization, and optimization applications, as well as multi-layer IP/optical discovery, visualization, and coordination.

i **Note:** The Network Infrastructure Management or Service Assurance feature package enables the NSP to discover the network IGP topology.

The Resource Control feature packages are:

- Control and Visualization Starter
- Path Control and Optimization
- Simulation
- Enhanced Optimization
- Multi-layer Discovery and Visualization
- Multi-layer Control and Coordination

2.4.2 Control and Visualization Starter

The Control and Visualization Starter feature package provides visual representations of the IGP and MPLS topologies, and enables traditional MPLS, control-plane, and brownfield LSP configuration.

For this function, the Virtual Service Router - Network Resources Controller (VSR-NRC) acts in a Virtual Network Function (VNF) capacity to perform topology discovery. The VSR-NRC implements the southbound Path Computation Element (PCE) function using the PCEP, BGP-LS and IGP protocols and an OpenFlow Controller (OFC).

2.4.3 Path Control and Optimization

The Path Control and Optimization feature package enables a stateful or stateless PCE to provide paths to requesting LSPs via PCEP/BGP based on a path profile. The feature package also provides the ability to invoke maintenance practices either manually or automatically based on event notifications.

i **Note:** To deploy Path Control and Optimization, you must also deploy the Control and Visualization Starter feature package.

2.4.4 Simulation

The Simulation feature package is a traffic-engineering tool for network engineers. The tool enables the design of highly optimized traffic flows, and enables the offline simulation of operational assurance changes to assist in capacity planning.

i **Note:** To deploy Simulation, you must also deploy the Path Control and Visualization feature package.

2.4.5 Enhanced Optimization

The Enhanced Optimization feature package offers automated closed-loop optimization for live network deployments using analytics and telemetry to trigger automated actions.

The functions provided by the feature package use the MDM and OAM frameworks to accomplish the required tasks.

i **Note:** To deploy Enhanced Optimization, you must also deploy the following feature packages:

- Control and Visualization Starter
- Path Control and Optimization

i **Note:** If your network includes 7250 IXR gen 1 devices, Enhanced Optimization requires an NSP Flow Collector and Flow Collector Controller in the NSP deployment.

2.4.6 Multi-layer Discovery and Visualization

The Multi-layer Discovery and Visualization feature package provides IP/optical topology discovery, path diversity verification, and bottom-up and top-down navigation. The MDM framework is used for communication with a service deployed in IP/Optical Coordination.

i **Note:** The feature package requires no other dependencies to view the transport topology, including the LAGs; however, to view the IP/MPLS (LSP) topology, please contact Nokia Professional Services for guidance; minimally, the Control and Visualization Starter feature package would also be required.

Optical management

An NSP deployment that requires optical management functions must include the WS-NOC product, which provides end-to-end optical management that includes service provisioning over multi-technology transport networks.

See the WS-NOC product documentation for more information.

2.4.7 Multi-layer Control and Coordination

The Multi-layer Control and Coordination feature package supports single-step LLI establishment, maintenance co-ordination, cross-domain connection management, and floating port restoration. The multi-layer control and coordination service is deployed on the NSP cluster.

i **Note:** To deploy Multi-layer Control and Coordination, you must also deploy the Multi-layer Discovery and Visualization feature package.

i **Note:** For multi-layer optimization use cases, you also require the following feature packages:

- Control & Visualization Starter + Path Control and Optimization (for multi-layer optimization only), or
- Control & Visualization Starter + Path Control and Optimization + Enhanced Optimization (for the extra precision/options of telemetry-based optimization and/or traffic-engineered ECMP)

2.4.8 The IGP Topology Discovery Service

The IGP Topology Discovery Service (nsp-topo-sync-app) synchronizes with VSR-NRC and populates the PostgreSQL database with topology information in the same way that NSP's path control function does, but uses less resources.

There is no UI for this service. Instead, REST APIs can be used to delete links, routers, and prefixes that are operationally down (this is equivalent to path control's clean-up references function). Visit the [Network Developer Portal](#) for more information.

The IGP Topology Discovery Services is installed when the networkInfrastructureManagement-igpTopology feature package is included in a deployment. Only one entity can integrate with the VSR-NRC in any deployment. As such, only the IGP Topology Discovery Service *or* NSP's path control function can be installed — not both. Regardless of which is installed, the same VSR-NRC configuration is required in the config.yml file. See the *NSP Installation and Upgrade Guide* for more information about populating this file.

2.5 Network Programming

2.5.1 Description

The Network Programming software suite consists of software for developers and network architects. The software employs the programmable NSP platform to enable the automation of network operations through open, model-driven APIs that allow integration with third-party devices, OSS, and orchestrators.

The Network Programming feature packages are:

- Software Development Kit
- Intent Based Networking Framework
- Workflow Automation Engine

Network Developer Portal

The [Network Developer Portal](#) provides information and an environment to support the development of the programmable artifacts used by the NSP.

2.5.2 Software Development Kit

The Software Development Kit feature package enables the use of an SDK to facilitate adaptor development in the Model Driven Mediation environment. The NSP SDK is a separately installable package that is available for download by registered customers from the NSP software delivery site.

2.5.3 Intent Based Networking Framework

The Intent Based Networking Framework feature package enables developers to manage the lifecycle of network and service intents using Network Intents.

2.5.4 Workflow Automation Engine

The Workflow Management Framework feature package enables developers and administrators to manage the lifecycle of workflows using NSP Workflows.

3 NSP architecture fundamentals

3.1 System design

3.1.1 Development methodology

The NSP architecture incorporates design considerations that include:

- open standards that promote interworking and integration with in-house or third-party systems; see [Appendix A, “NSP technology standards”](#) for information
- flexible internal model that accommodates product evolution
- deployment agility for adaptation to changing network scope or complexity
- programmability for dynamic management of network operations
- SSO access
- IPv4 and IPv6 support on internal, external, and mediation interfaces
- cloud-native distributed processing for efficiency and horizontal scalability
- fault-tolerance safeguards that include local and geographic redundancy
- stringent internal security among components
- highly secure local and remote client access

3.1.2 Core system elements

The various components of the modular NSP architecture work together as a customized management solution designed to meet your current and changing network or business requirements. Components and functions can be readily added, updated, or removed, as required.

Depending on the installation options that you choose, ancillary components in an NSP system may provide the following:

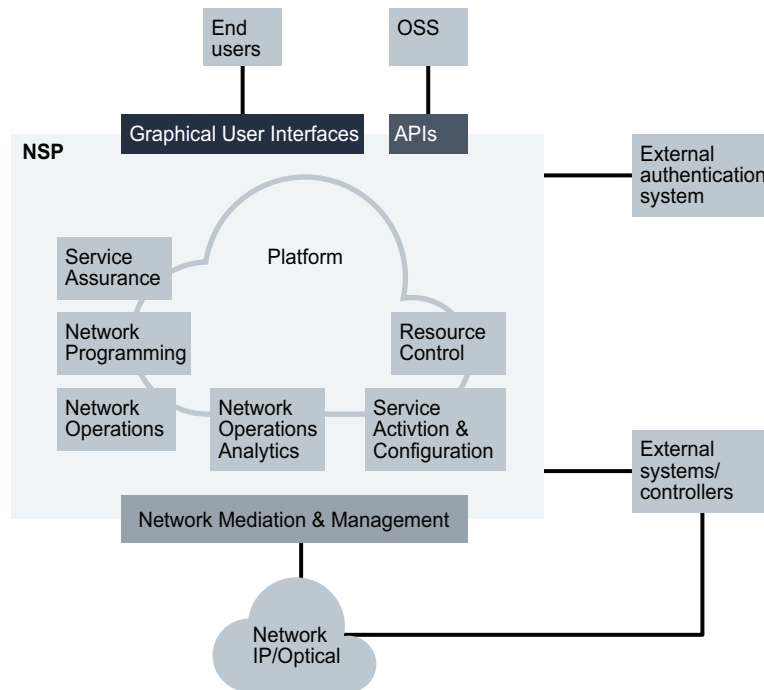
- graphical user interfaces
- public APIs
- management domain or business logic
- mediation services

The NSP is designed to interface or integrate with external systems such as the following:

- remote authentication agents for user access
- other EMS or network controllers

The following figure shows an abstract representation of the NSP services and interfaces.

Figure 3-1 NSP architecture, conceptual view



36826

3.2 NSP graphical user interfaces

3.2.1 Operator interfaces

The following NSP graphical user interfaces are available:

- browser-based UI—provides access to all NSP functions
- Java-based—provides classic management access to the NFM-P

Each GUI supports cross-launch to other applications and management systems, and provides direct access to the product documentation.

3.2.2 NSP GUIs

The *NSP User Guide* describes the NSP browser-based GUI, and provides general access and usage information. The available functions depend on which NSP installation options you specify during system deployment.

Classic management GUI

The NFM-P Java-based client GUI enables operators to perform policy-based classic device management. GUI client deployment is supported on multiple platforms; see the *NSP Planning Guide* for information.

The *NSP NFM-P User Guide* describes how to use the Java-based client GUI to perform classic management operations.

3.3 NSP APIs

3.3.1 External / public APIs

The NSP architecture publicizes the following APIs for Online Support System (OSS) clients:

- NSP REST—for HTTP CRUD (Create, Retrieve, Update, Delete) operations on NSP carrier SDN, NFM-P and NSP objects
- NSP RESTCONF—for communication with the NSP system and state notifications
- NSP Kafka—for event and alarm notifications about the managed network

See the [Network Developer Portal Documentation](#) for information.

i **Note:** The available API functions depend on which NSP installation options you specify during system deployment.

Classic management APIs

The NFM-P has the following APIs:

- NFM-P REST—provides access to network management functions for use cases that the NSP APIs do not fulfill
- NFM-P XML SOAP/JMS—for SNMP network management; see the *NSP NFM-P XML API Developer Guide* for information

3.4 NSP network mediation

3.4.1 Description

The NSP interacts with the network using MDM, and through optional classic SNMP management using protocol providers such as the VSR-NRC. NSP mediation provides:

- network data access
- object provisioning and modification mechanisms
- network change and event notifications

i **Note:** The NSP supports NE mediation using IPv4 and IPv6 concurrently; see “IP version support” in the *NSP Installation and Upgrade Guide* for information.

The NSP includes the following network mediation mechanisms and components:

- MDM, in a deployment that includes Pluggable Network Adaptation, which provides mediation services for:
 - devices managed using YANG model-based interfaces
 - multi-vendor IP devices managed using SNMP
- NSP Flow Collectors, which collect AA Cflowd or System Cflowd statistics directly from NEs
- NFM-P, in a classic management deployment for legacy SNMP management

-
- VSR-NRC, which mediates with IP NEs for PCE-PCEP and OpenFlow communication
 - WS-NOC, for optical transport network mediation

Classic management

The NFM-P provides role-based FCAPS functions for classically managed networks. In addition to managing Nokia devices, you can obtain NFM-P adaptors for managing third-party equipment.

LLDP link discovery

The NSP provides an internal LLDP application that discovers MDM managed LLDP links. The LLDP application is deployed by default and no user configuration is required. The LLDP application requires that all appropriate MDM LLDP v2 adaptors for your NEs are configured on your system.

i **Note:** Because MDM adaptors may not support events on link objects, there may be a delay of up to 60 minutes before link updates appear in the NSP. Any other updates to the LLDP state that are not triggered by the port can take up to 24 hours to appear in NSP.

3.5 Classic management architecture

3.5.1 NFM-P system

The NFM-P provides the classic management's mediation functions for SNMP-managed devices, and includes a local database for NSP user authentication and authorization.

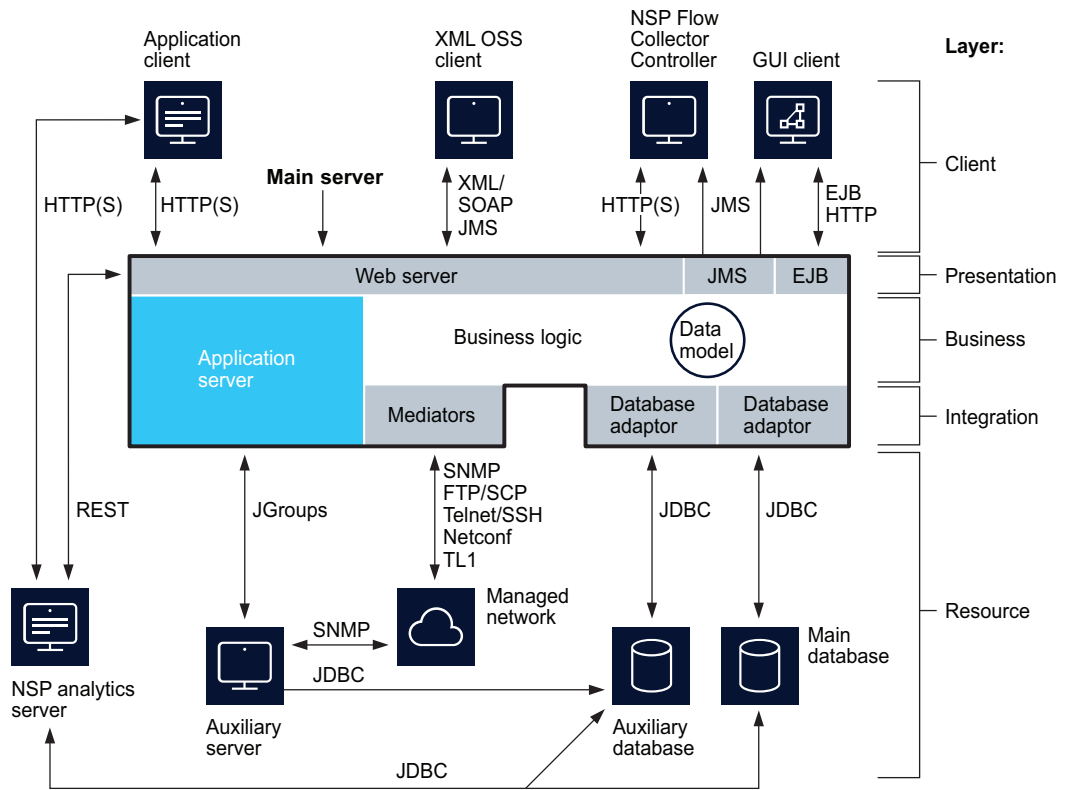
The NFM-P supports SNMP mediation functions that include:

- traditional device management
- traditional L2 and L3 service management
- IP/MPLS network infrastructure management
- IP/MPLS network and service assurance

The NFM-P is deployed on multiple server and database stations, and supports redundant deployment using a warm standby model. The NFM-P system elements use proprietary and third-party software, and are logically organized in a framework that has the following layers, as illustrated in [Figure 3-2, "NFM-P multi-layer model" \(p. 27\)](#):

- resource
- integration
- business
- presentation
- client

Figure 3-2 NFM-P multi-layer model



28672

Resource layer

The resource layer includes the network of managed NEs, the main database, and optional components like auxiliary servers and an auxiliary database. The available resources include, for example, NE configuration backups and software images, network topology information, customer service configurations, and statistics.

Integration layer

The integration layer buffers resource-layer elements from the business layer. This layer contains the mediators, which communicate with equipment in the managed network, and the database adaptor. The mediator components translate messages from the business layer into the commands that are sent to the managed network. Messages from the network are processed by the mediator components and passed to the business layer. The database adaptor translates business logic requests into JDBC commands, and translates JDBC responses into Java business model objects.

Business layer

The business layer contains the logic and data model for NFM-P functions. The business logic processes client requests, SNMP traps from managed NEs, and internal server events, and

performs the appropriate actions on the managed network, clients, and data model. The data model maintains information about network objects and their relationships. To support the business layer, an application server provides Java EE services.

Presentation layer

The presentation layer buffers the application logic from the client layer, and includes the following:

- web server that receives messages from OSS clients and passes them to the business layer
- application server that processes EJB method invocations from GUI clients and returns the responses from the business layer; the application server also forwards JMS messages from the business layer to GUI and OSS clients

Client layer

The GUI, OSS, and browser-based application clients comprise the client layer.

- A Java VM on a GUI client sends EJB RMI to a main server.
- The OSS clients send XML/SOAP, or REST messages to a main server.
- Web clients use JNLP for portal access.

4 Platform programmability

4.1 NSP programming frameworks

4.1.1 Introduction

The NSP platform programmability functions employ a number of frameworks, some of which can be customized, as described in the [Network Developer Portal](#):

- **MDM**
Provides a pluggable, programmable model-driven interface for equipment from any vendor, and enables the NSP to manage new devices and device versions without the need for an NSP system enhancement or upgrade
- **Workflow management**
Enables the creation of workflows to automate routine or repetitive tasks
- **Intent management**
Translates an operational intent into configuration artifacts that can be executed automatically and continually, as required; the framework engine enforces an intent, which is a desired network state or configuration
- **Synchronization**
Ensures that the NSP YANG-model view of the network infrastructure is consistent with the actual network configuration and state
- **Operations, Administration, and Maintenance (OAM)**
Manages the execution of OAM tests against the network infrastructure; the tests can be run on demand, or configured to provide continuous monitoring
- **RESTCONF gateway**
Provides APIs to YANG models that support a variety of functions, and allows for the addition of new YANG models

4.2 MDM framework

4.2.1 Description

The MDM component is a mediation layer that uses runtime loadable adaptors to interface with NEs and southbound controllers. MDM translates between device-specific data models and internal data models.

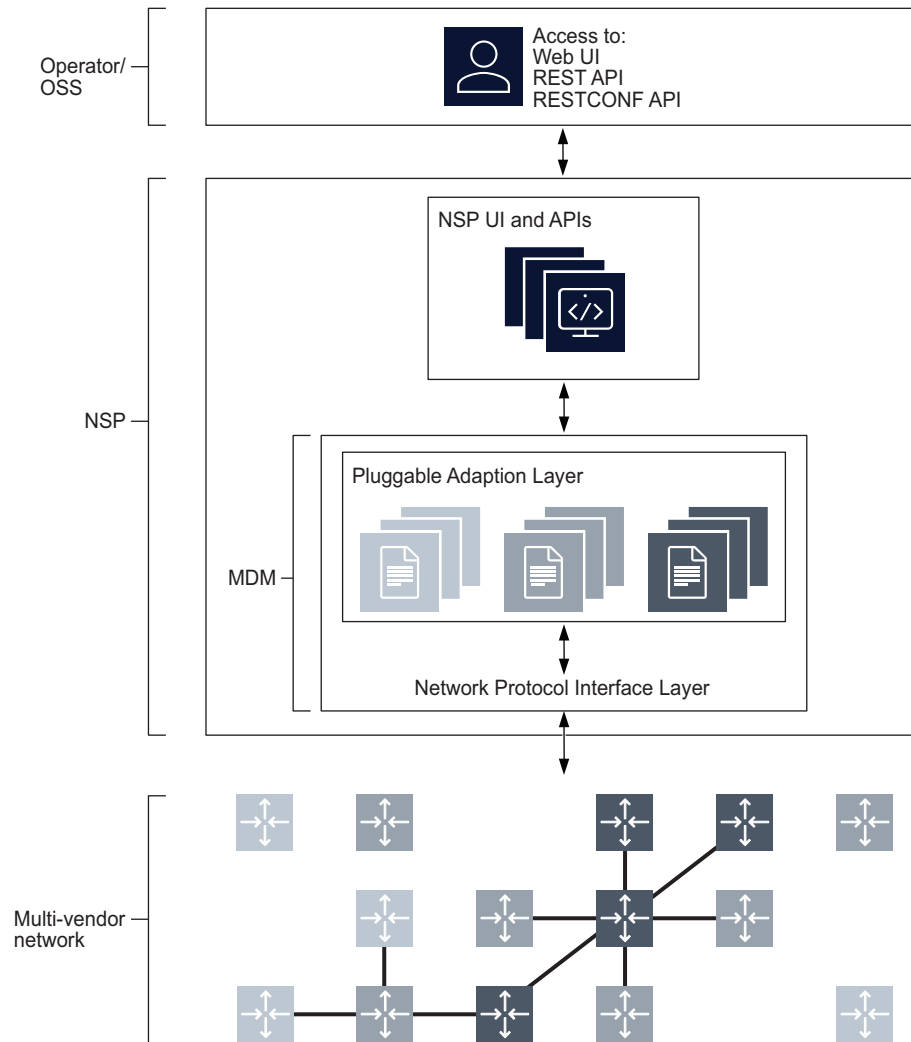
In MDM, the data objects that make up an NE and its capabilities are defined using YANG models. MDM provides the translation and abstraction required for automated functions to interact with the YANG model and manage the NEs.

MDM translates NE network protocol inputs for NSP functions, and creates protocol inputs to manage the NE configuration, state, and telemetry.

An MDM server is installed as part of the NSP deployment. Within the MDM server, network protocol inputs from devices are adapted to create inputs to NSP, and vice versa. Adaptation is performed by MDM adaptor files. Adaptors are installed on the MDM server according to network requirements.

The following figure illustrates the basic concepts of MDM. MDM incorporates a network protocol interface layer, pluggable adaptation layer, and application interface layer to allow NSP to manage Nokia and multi-vendor devices using Netconf/YANG.

Figure 4-1 Model-driven mediation in NSP



38909

4.2.2 MDM in NSP

MDM is a component of NSP enabled by the following feature packages:

- Multi-layer Discover and Visualization
- Multi-layer Control Coordination

MDM is provided notably for model-driven Nokia and multi-vendor NE management.

Model-driven management interfaces are based on a common infrastructure that uses YANG models as the core definition for configuration, state, and operational actions. All model-driven interfaces take the same common underlying YANG modules and render them for the management interface.


NSP supports NE discovery, management, and configuration for MDM managed NEs. The availability and level of functionality depends on the adaptors. Adaptors are developed continuously, and updated on the software download site regularly, meaning that application functionality can expand at any time, without the need to upgrade the NSP or the device.

4.3 Workflow management framework

4.3.1 Description

The NSP workflow management framework provides programmable network and service operational automation. The framework enables orchestration among the managed NEs, nspOS infrastructure, NSP functions, and external systems.

The programmable automation of workflow management enables fine control over a wide array of operational responsibilities that include network migration, configuration management, performance testing, security management, and software management. A workflow can be manually initiated, scheduled, or triggered by network events or the outcomes of other NSP operations. An existing workflow can easily be adapted to your specific procedural requirements, and to support a new Nokia or third-party device release.

 **Note:** NSP workflow management functions are available for OSS via programmable APIs.

As an example, the workflow management use cases include:

- NE software upgrades
- service activation tests
- service fulfillment that invokes pre- and post-deployment workflows
- customizing policy logic for other NSP functions
- mass service migration from one tunnel type to another
- one place scheduling for all NSP REST APIs and NE CLI operations
- centralized orchestration for Nokia NSP and third-party APIs

4.4 Intent management framework

4.4.1 Description

The NSP Network Intents function enables you to create and execute NSP intents. Using intents, you can implement your planning and design at the network level rather than at the NE level.

The intent management framework translates the high-level goal defined in an intent to the required network configuration. The NSP subsequently generates and validates the configuration, and continually verifies the state of the objects within the scope of the intent against the intent itself.

4.5 Synchronization framework

4.5.1 Description

The synchronization framework extracts data from lower mediation layers to update the NSP data models. The framework uses a combination of static and programmable mechanisms.

Data synchronizers use programmable data mapping definitions to update YANG data stores such as persisted device models and the NSP normalized model.

4.6 Operation, administration, and maintenance (OAM) framework

4.6.1 Description

The OAM framework is a data-driven framework that manages OAM tests and templates, and enables test-suite creation, configuration, execution, and result collection. The MD OAM framework employs data-driven relationships between OAM tests and NSP equipment or service objects, and includes metadata for associating NSP Telemetry records with defined OAM tests. The framework also enables the publication of test results to a Kafka topic and persistent storage, and allows the inclusion of additional information. Test results can also be classified according to dynamically updated classifications.

The OAM Tests function supports test creation, editing, and deletion, and presents the OAM test results for viewing.

The OAM framework is configured and managed using OAM metadata submitted to a RESTCONF API. The programmable elements include:

- result classification mappings—define how result data is classified
- MDM OAM adaptor—enables mediation between the RESTCONF API and the managed NEs for MD OAM content

4.7 RESTCONF gateway

4.7.1 Description

The RESTCONF gateway provides API access to YANG-defined device and NSP data models. RESTCONF uses persisted and network data stores.

See the [Network Developer Portal](#) for information about using the RESTCONF API.

4.8 IETF standard models

4.8.1 Framework

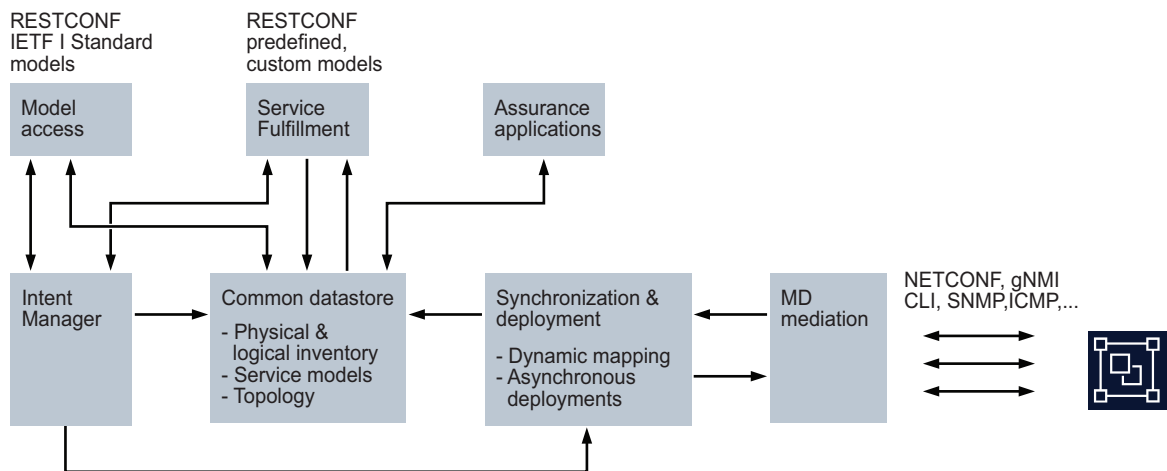
The Internet Engineering Task Force (IETF) creates standard models to support use cases in the northbound Interfaces (NBIs) of IP controllers. These models focus primarily on L2NM and L3NM network services, IETF network definitions, and IETF TE functions.

The NSP features an IETF framework that functions as a programmable northbound mediation layer, allowing for the import of IETF-based models into the NSP. CRUDL operations are automatically supported, with necessary invocations to NSP functions, as required. Customization is supported for overriding the CRUDL operations.

All IETF configurations are performed using the NSP RESTCONF API. Imported IETF service or network YANG models are exposed via the API, which is compliant with RESTCONF RFC 8040.

4.8.2 Architecture

The following diagram shows the architecture of the NSP's IETF framework.



37505

In order to use the NSP IETF framework, the Network Infrastructure Management and Service Activation and Configuration licensed packages must be part of your deployment. Additionally, several artifacts must be installed, including IETF artifacts, logical inventory adaptors, datasync mapping files, and predefined intent types. Documents delivered with these artifacts provide installation and compatibility information. These can be obtained from the Nokia [Support Portal](#) at the following locations:

1. `NSP/release/Adaptors/vendor` — where *vendor* is the supported device type.

NOTE: Customers should refer to the relevant adaptor documentation for more details about the adaptor suites. These documents are available from the Nokia Support Portal alongside the adaptor suites.

2. `NSP/release/Artifacts` — This directory hosts artifacts available for production NSP

deployments. It contains the predefined NSP service intents collection that is required for IETF services: `NSP_Predefined_Service_Intents_release_version.zip`. This directory also contains NSP ZTP intents, and ICM intents, which are unrelated to the NSP IETF framework.

3. `NSP/release/Beta_Artifacts` — This directory hosts artifacts that come with certain limitations. Most IETF artifacts can be found here, as well as the associated MoPs.

See the *NSP Network Automation Guide* for information about installing artifacts.

4.8.3 Topology

NSP supports the use of RESTCONF to retrieve IETF-compliant topology models, including layer 2 and 3 topology aspects, and traffic engineering topology aspects. This is in accordance with RFC8345, RFC8346, RFC8944, and draft-dbwb-opsawg-sap-02, and allows for the retrieval of service attachment points. IETF-compliant topology models are retrieved using the NSP's RESTCONF APIs. Visit the [Network Developer Portal](#) for more information.

4.8.4 Logical inventory

NSP supports the use of RESTCONF to retrieve IETF-compliant and OpenConfig-compliant logical interfaces, as defined by the TIP MUST IP SDN Controller NBI Technical requirements document. This is in accordance with RFC8345, RFC8528, and OpenConfig specifications. IETF-compliant logical inventory models are retrieved using the NSP's RESTCONF APIs. Visit the [Network Developer Portal](#) for more information.

4.8.5 Path control

NSP supports the use of RESTCONF to create, delete, update, and read IETF-compliant LSPs according to draft-ietf-teas-yang-te, including regular RSVP LSPs and PCC-Initiated RSVP/SR LSPs for deployment through NFM-P or MDM. The LSPs may be configured with a number of constraints - including hops and bandwidth - and may also be protected via redundancy. IETF-compliant LSPs are configured using the NSP RESTCONF API. Visit the [Network Developer Portal](#) for more information.

4.8.6 Services

NSP supports the use of RESTCONF to create, delete, update, and read IETF-compliant Layer 2 and Layer 3 services (L3 VPN and E-Line). IETF service models are stored and updated to reflect network changes. IETF service CRUD requests are translated prior to deployment. Additional services and service attributes can be added later via customization. IETF artifacts must be obtained from the Nokia [Support Portal](#). IETF-compliant services are configured using the NSP's RESTCONF APIs and IETF models. Visit the [Network Developer Portal](#) for more information.

5 Deployment fundamentals

5.1 Overview

5.1.1 Introduction

The NSP is a highly distributed system that requires a number of hosts, where a host is defined as a physical or virtual processing entity that has a discrete OS instance. The functions provided by the NSP software are installed on the NSP hosts, which collectively constitute an NSP deployment.

Many different NSP deployment scenarios that employ various levels of redundancy are supported; see [Chapter 8, “System redundancy and fault tolerance”](#), the *NSP Planning Guide*, and the *NSP Installation and Upgrade Guide* information.

5.1.2 Deployable NSP platform elements

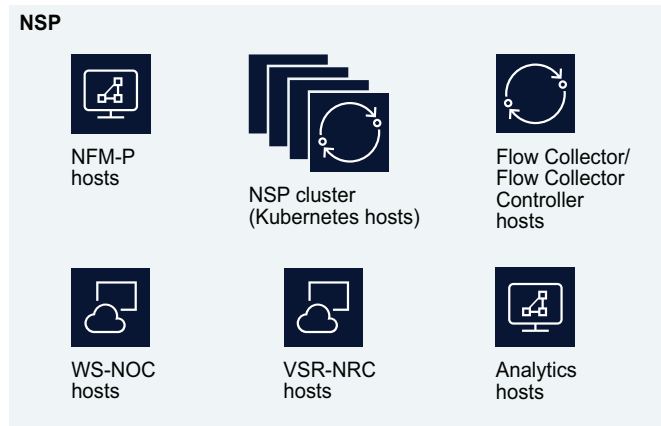
The core elements of the NSP platform are deployed in a Kubernetes container environment that Nokia provides. Other components of an NSP system are deployed outside the Kubernetes environment.

The following elements, which are described in subsequent topics, comprise an NSP system:

- NSP cluster—group of containerized VMs managed by a Kubernetes orchestration layer that co-ordinates the deployment of NSP system services
- depending on the system requirements, one or more of the following, which are deployed outside the Kubernetes environment:
 - NFM-P
 - WS-NOC
 - VSR-NRC
 - NSP analytics servers
 - NSP auxiliary database
 - NSP Flow Collectors and Flow Collector Controllers

The following figure shows a high-level conceptual diagram of an NSP deployment. A component in the diagram may require one or multiple hosts.

Figure 5-1 NSP deployment, abstract view



38429

5.2 NSP Kubernetes implementation

5.2.1 Environment

A small-profile VM called the NSP deployer host holds the required NSP container runtime environment for NSP cluster instantiation. The required NSP software instances are deployed in Kubernetes pods to the cluster members, which are called NSP cluster VMs.

The NSP container environment uses only the storage local to each NSP cluster member; no cluster-wide network storage is required. The NSP service pods are deployed only to specific cluster members to ensure that each service has access to fixed portions of the local storage.

NSP deployment supports various levels of system redundancy; see [Chapter 8, “System redundancy and fault tolerance”](#), the *NSP Planning Guide*, and the *NSP Installation and Upgrade Guide* for information.

5.2.2 Deployment profiles

Based on the deployment scope and required level of fault tolerance, you specify one of the following profiles during NSP cluster deployment, based on the response to your Nokia Platform Sizing Request:

- node-labels-basic-1node.yml—single-VM deployment
- node-labels-standard-3nodes.yml—standard deployment of three VMs
- node-labels-standard-4nodes.yml—standard deployment of four VMs
- node-labels-enhanced-6nodes.yml—enhanced deployment for HA

As required to accommodate system or network growth, you can later add instances of functions such as MDM to an NSP cluster that has sufficient resources. See the *NSP Installation and Upgrade Guide* for information.

5.2.3 NSP cluster

The NSP cluster, a containerized set of Kubernetes hosts, is the central component of any NSP deployment. The cluster hosts the NSP Platform functions, and a significant number of NSP framework and application functions.

An NSP cluster may consist of only one member, or three or more that each host a portion of the installed NSP software. Each NSP cluster member is a designated Kubernetes worker node on which NSP services run. One or more members are also designated control nodes.

NSP functions hosted by NSP cluster members

The following installation options are deployable only in an NSP cluster:

- Platform - Base Services
- Platform - Logging and Monitoring
- Platform - Pluggable Network Adaptation
- Model Driven Mediation
- Network Infrastructure Management - Basic Management
- Network Infrastructure Management - Device Configuration
- Network Infrastructure Management - Performance Indicators And Alerts
- Network Operations - Baseline Analytics
- Service Assurance
- Service Activation and Configuration - Intent Based Service Fulfillment
- Intent Based Networking Framework
- Workflow Automation Engine
- Simulation

5.3 VSR-NRC

5.3.1 Description

The VSR-NRC is required in order for NSP to establish PCE-PCEP and OpenFlow communication with SNMP-managed NEs. The VSR-NRC is a virtual SR OS instance that uses the software image from the matching SR OS release; the VSR-NRC license enables additional code for NSP interaction. For platform requirements and deployment information, see the *Virtualized 7750 SR and 7950 XRS Simulator (vSIM) Installation and Setup Guide*.

5.4 NSP analytics servers

5.4.1 Description

The software associated with the Analytics installation option is deployed on one or more NSP analytics servers, which run business intelligence software to analyze raw and aggregated NE telemetry data collected via legacy SNMP management or MDM.

Like other NSP components, NSP analytics servers support redundant deployment; see [8.7.1 “Description” \(p. 65\)](#) for information about NSP analytics server redundancy and other fault-tolerance mechanisms.

5.5 NSP Flow Collectors and Flow Collector Controllers

5.5.1 Description

An NSP Flow Collector is a horizontally scalable component that collects AA Cflowd or System Cflowd statistics directly from NEs. The statistics records can be forwarded to remote servers or kept in persistent storage, and are made available for processing by third-party tools or NSP Analytics.

An NSP Flow Collector Controller is required in any deployment that includes NSP Flow Collectors. A Flow Collector Controller extracts the NFM-P network data model for use as a statistics-collection framework by each Flow Collector, and updates the model as the NFM-P sends JMS notifications about model updates.

For a small-scale deployment, you can collocate an NSP Flow Collector Controller and an NSP Flow Collector on one station. A small-scale deployment has a maximum of two stations, and supports the following:

- standalone—one station that hosts a Flow Collector Controller and Flow Collector, and a second station that hosts only a Flow Collector
- redundant—two stations that each host a Flow Collector Controller and Flow Collector

Redundancy

Like other NSP components, NSP Flow Collectors and NSP Flow Collector Controllers support redundant deployment; see [8.6.1 “Description” \(p. 64\)](#) for information about NSP Flow Collector and Flow Collector Controller redundancy and other fault-tolerance mechanisms.

5.6 Auxiliary database

5.6.1 Description

An auxiliary database is an optional, horizontally scalable database that expands the NSP storage capacity for demanding operations such as statistics collection, and performs the data aggregation required for NSP Analytics.

The database is deployed on one station, or in a cluster of three or more stations, depending on the scale requirement. In a multi-station auxiliary database, load balancing and data replication among the stations provide high performance and robust fault tolerance.

Auxiliary databases support DR deployment consists of two matching auxiliary database clusters that are deployed in separate data centers.

5.7 Classic management components

5.7.1 NFM-P system components

A basic NFM-P system consists of the components described in the following topics. Some deployment types may require additional components, as described in the *NSP Planning Guide*.

Internal components

Internal NFM-P components such as Java modules, database software, and web server software are represented by license files in the following directory on a main server:

```
/opt/nsp/nfmp/server/nms/distribution/licenses
```

5.7.2 Main server

A main server is the central Java-based processing engine in an NFM-P system. A main server can be deployed on a dedicated station, or collocated on a station with a main database. A main server hosts an application server, JMS server, web server, protocol stack, and database adaptor. Functions like statistics collection are performed by a main server only in a deployment that does not include any auxiliary servers.

5.7.3 Auxiliary server

An auxiliary server, like a main server, is a Java-based processing engine, but is an optional, scalable component that extends the system capacity for collecting SNMP statistics. An auxiliary server collects data directly from NEs, and is controlled and directed by a main server.

5.7.4 Main database

An NFM-P main database is a relational database that provides persistent storage to hold the network data repository. A main database can be deployed on a dedicated station, or collocated on a station with a main server.

5.7.5 Single-user GUI client

A single-user GUI client is a Java-based graphical interface for network operators. Single-user GUI client deployment is supported on multiple platforms.

5.7.6 Client delegate server

A client delegate server supports simultaneous GUI sessions using one client software installation. A client delegate server can host local and remote user sessions, and supports the use of a third-party remote access tool such as a Citrix gateway. Client delegate server deployment is supported on multiple platforms.

A GUI session that is opened through a client delegate server is functionally identical to a single-user client GUI session. The client delegate server locally stores the files that are unique to each user, such as the client logs and GUI preference files.

5.7.7 Internal subcomponents

Internal subcomponents, for example, Java modules, database software, and web server software,

are represented by license files in the following directory on a main server:
`/opt/nsp/nfmp/server/nms/distribution/licenses`

5.8 WS-NOC

5.8.1 Description

An NSP deployment that requires optical management functions must include the WS-NOC product, which provides unified end-to-end optical management and support functions such as service provisioning and service assurance.

The WS-NOC can manage multi-technology optical transport networks that include SDH/SONET, carrier Ethernet, WDM, ROADM, OTN, and packet. Browser-based fault management provides network and service assurance functions, and an API enables OSS integration.

See the *WS-NOC Getting Started Guide* for more information.

6 Security architecture

6.1 NSP system security

6.1.1 TLS

The NSP supports the use of Transport Layer Security (TLS) throughout the NSP system. The NSP installation software includes a utility called a Public Key Infrastructure (PKI) server that you can use to automate the distribution of TLS artifacts for NSP components. A PKI server can generate, sign, and distribute a self-signed TLS certificate, or use a certificate from another source.

i **Note:** The NSP supports only TLS v1.2; however, you can enable older TLS versions for compatibility with OSS or external systems that do not support TLS v1.2.

TLS ensures secure external communication between NSP clients and the NSP cluster, and among NSP components. The NSP supports the use of external TLS certificates signed by a trusted public Certificate Authority (CA), and self-signed certificates.

You determine the source and signing authority of the external TLS certificate in an NSP system. The internal certificate, however, is automatically created and signed by an internally generated private CA on the PKI server, so no certificate from any external CA is trusted for internal system access.

Each NSP cluster serves as the central store of the following certificates for the other NSP components in the local datacenter:

- external, customer-provided or generated by PKI server
- internal, generated by PKI server

Other external security mechanisms

In addition, session credentials and messaging can be protected using mechanisms and protocols such as the following:


- NAT between system components
- HTTPS at the application layer for API clients
- SNMPv3 for communication with network devices
- SMTPS or STARTTLS for secure e-mail notifications

You can also enable HTTP Strict-Transport-Security, or HSTS, during system deployment, which enforces the use of HTTPS by any browser that connects to the NSP. See the *NSP Installation and Upgrade Guide* for information about enabling HSTS.

6.1.2 SELinux

The deployment of SELinux in permissive or enforcing mode to log user operations is supported on the RHEL OS of all NSP components, with the exception of an auxiliary database, which supports SELinux only in permissive mode.

The NSP supports the upgrade of SELinux-enabled components; however, SELinux must be in permissive mode during the upgrade. Switching to SELinux enforcing mode is done only after a component installation or upgrade.

 **Note:** SELinux is enabled in permissive mode on an NSP RHEL OS disk image, but must be manually enabled after a manual RHEL OS installation.

See “What is SELinux?” in the *NSP System Administrator Guide* for information about deploying and managing SELinux for the NSP.

6.2 User security and session management

6.2.1 Single-sign-on

The NSP single-sign-on (SSO) mechanisms enable a common security framework for all supported NSP functions and services. NSP supports authentication against local and external authentication agents such as LDAP, RADIUS, or TACACS+ servers.

In addition to user access control, the NSP provides user session management and activity logging. See [6.5 “Activity logging” \(p. 44\)](#) and the *NSP System Administrator Guide* for more information.

OAuth2 user authentication

OAuth2 is a Single Sign-On (SSO) authentication mode based on the Keycloak open-source identity and access management solution using standard OAuth 2.0 protocol. The OAuth2 mode supports local user management and external authentication agents, but does not support authentication against an NFM-P local user database. The NSP includes a utility for importing users from an NFM-P local user database for the migration from CAS to OAuth2.

CAS user authentication

CAS, the legacy NSP authentication module, is an open-source SSO solution that supports local user authentication against an NFM-P local user database, if the NSP deployment includes the NFM-P. CAS also supports external authentication agents.

6.2.2 Kafka authentication

All Kafka communication is secured by default using TLS. Additionally, you can enable authentication for internal and external Kafka clients. Internal and external Kafka authentication are independent of each other, and are enabled and configured separately.

Internal Kafka authentication for communication among NSP subsystems uses mTLS two-way authentication.


External Kafka authentication requires NSP user credentials and OAuth2 authentication mode.

See the *NSP Installation and Upgrade Guide* for information about configuring internal and external Kafka authentication.

6.3 Firewall support

6.3.1 Description

The NSP supports firewall deployment on all NSP host interfaces, although firewall support among system components may vary. A component such as the NFM-P or WS-NOC that has multiple system elements may have additional firewall requirements.

 **Note:** Firewall deployment between the members of an NSP cluster is not supported.

See the *NSP Planning Guide* and any specific component planning documentation, as required, for firewall port requirements and restrictions.

6.4 NSP User Access Control

6.4.1 Description

NSP User Access Control (UAC) is a mechanism that enables an NSP administrator to define user access rights to NSP functions and data. If UAC is disabled, each NSP user has access to all NSP objects.

The NSP supports the creation of local user groups and roles. A role, which is assigned to a user group, can be given read, write, or execute permission to network inventory and data. Consequently, an NSP user has the access defined in a role to only the resources in the associated resource groups defined by an NSP administrator.

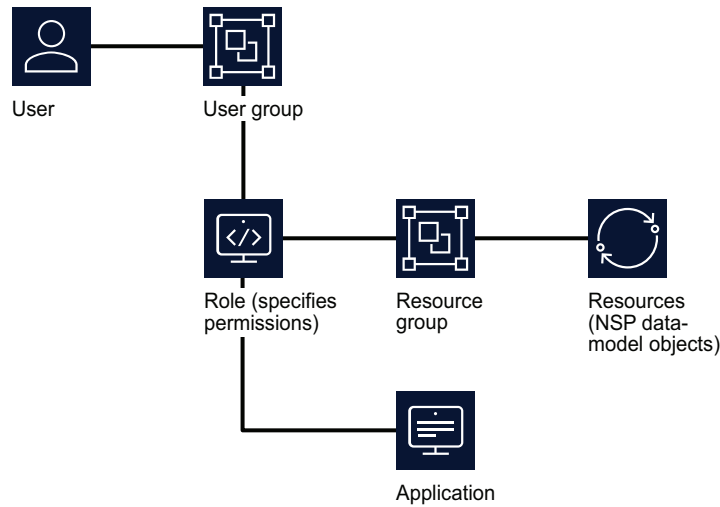
Resource groups

To restrict equipment or service access to specific operators, you can define resource groups that contain NSP objects such as equipment and service components, and can specify the user access to each group.

A UAC resource group can include objects such as the following:

- service—IP links, services, service sites, service endpoints, service bindings
- equipment—equipment, chassis, LAGs, devices
- KPI—NE and service KPIs
- Analytics—Analytics resources

Figure 6-1 UAC roles and resource groups



36825

6.5 Activity logging

6.5.1 Description

The NSP performs centralized user-activity logging for audit-trail creation and analysis. The logging creates records for the following activities:

- all direct user-initiated object modification calls to the REST API
- all identification and authentication attempts

A log record identifies the user, the called API, and any affected objects. For audit-trail creation, you can filter the records by user or user group.

6.6 NFM-P system security

6.6.1 Platform security

The RHEL OS is common to all NSP components, including the NFM-P. The OS is protected by firewalls and stringent internal security measures that restrict access to files and functions. The NFM-P supports platform-wide mechanisms such as SELinux, which records RHEL user actions, and HTTPS Strict-Transport-Security (HSTS), which restricts client browser access. NE management communication can be secured using protocols such as SNMPv3, as well as the strong security associated with the Federal Information Processing Standards (FIPS).

See the *NSP Security Hardening Guide* for more information about NSP platform security.

6.6.2 Communication security

The NFM-P employs strict security at the session and other communication layers. Interfaces between a main server and other system components are secured using Transport Layer Security, or TLS.

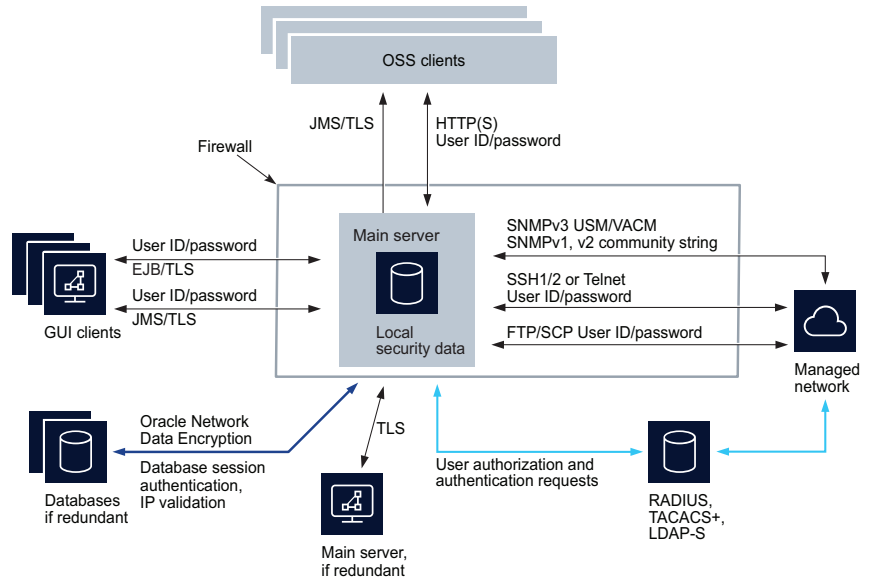
Communication with a main database is secured using Oracle Network Data Encryption.

A GUI, application, or OSS client must provide user credentials for access to the NFM-P. Session credentials and messages are protected using mechanisms and protocols that include the following:

- HTTPS, as the application-layer transport for clients
- SSH, SCP, and SNMPv3 with USM or VACM, at the application layer for communication between a main server and the managed network
- NAT, at the network layer, between the following:
 - main server and single-user GUI client or client delegate server
 - main or auxiliary server and OSS client
 - main or auxiliary server and managed network
- IP validation, at the network layer, between the main or other server components and each main database

The following figure shows the NFM-P components and the security mechanisms.

Figure 6-2 NFM-P security mechanisms



6.6.3 NFM-P session management

An NFM-P operator can configure authentication, accounting, and administrative (AAA) functions using the local NFM-P security mechanisms, a third-party server, or both.

- Local NFM-P authentication is performed using a local database of users and a local security scheme.
- Supported remote authentication servers are RADIUS, TACACS+, LDAP and LDAP-S, and CSA, which have separate user lists and administration processes.

NFM-P user accounts consist of a user name, password, and an associated user group, scope of command, and span of control over network objects. User groups define user authorization levels, and control the level of access to objects such as equipment, customers, services, and alarms. An NFM-P administrator can limit the type of user access per managed NE; for example, allowing FTP access but denying console or SNMP access.

Client sessions

Client sessions use the following authentication mechanisms.

- A GUI client EJB session is authenticated using the client username and password.
- An OSS client session is authenticated using cached information from an authorization server.
- A JMS session is authenticated using the client username and password.

Database sessions

A main database is accessible through a connection that is secured by a user name and password. After each database update in response to a GUI or OSS client request, the client activity log records the request information, which includes the user name.

Secure communication between a main server and main database is available using IP validation, which is typically configured on a main database station during installation or upgrade.

Managed NE sessions

A main or auxiliary server opens CLI, FTP, SFTP and SCP sessions on managed NEs. A managed NE uses a local security database, or a third-party service such as RADIUS or TACACS+, to perform AAA functions.

SNMPv3 message authentication and authorization are handled by the USM and VACM mechanisms, which define the user authorization permissions. Older SNMP versions are authenticated using community strings. Each SNMP message is individually authenticated.

6.6.4 Network transport security

Transport-layer security is available to the network protocols that carry messages between NFM-P components.

Main server and clients

Communication between a main server and clients is performed using messaging such as the following.

- XML API clients use HTTP or HTTPS to send XML/SOAP messages, and receive notifications using JMS, which can be secured using TLS.
- REST API clients use HTTPS.
- GUI clients use the EJB interface, which can be secured using TLS.

Servers and managed NEs

When SNMPv3 is used, an authentication key using current algorithms and ciphers is included in each message and checked against the shared encryption key. SSH provides the security for a CLI session between a GUI client and a managed NE.

RSA encryption is available for communication between auxiliary servers and managed NEs; contact customer support for information.

Firewall support

The NFM-P supports firewall deployment on all server interfaces; for example, between a main server and the auxiliary servers, GUI, and OSS clients, and between a main or auxiliary server and the managed network. See the *NSP Planning Guide* for firewall and reserved TCP port information.

6.7 NSP software security summary

6.7.1 DFSEC requirement implementation

The NSP follows Nokia's Design For Security (DFSEC) process to ensure the security of the NSP product software. The DFSEC defines a framework for delivering secure products based on providing defence-in-depth, while utilizing a continuous secure delivery process based on industry-recognized standards and best practices.

The DFSEC requirements have been assembled by Nokia over several years and are based on Nokia involvement in global standards development, engagement with customers and regulators, participation in industry forums, and the collective experiences of many product development teams. The DFSEC requirements cover general software security, network security, operating system security, information security, database security, application/web server security, virtualization security, as well as authentication, authorization, and accounting. Requirements compliance is verified at various points throughout the release lifecycle from initial design to delivery.

NSP security testing occurs every release and utilizes tools recommended in the DFSEC process. The security testing includes the execution of vulnerability scans, web application scans, port scans, targeted robustness (aka "fuzzing") and DoS testing. Security testing is carried out using a mix of commercial and internal tools. The DFSEC also specifies the use of the Nokia Software Vulnerability Management tool, which is used to manage any vulnerabilities that have been found in the third-party libraries used within the NSP product; these include:

- the identification and notification of any new vulnerabilities declared against the third-party libraries used,

-
- tracking the vulnerability assessment and severity assignment, and
 - tracking the mitigation activity necessary to eliminate the vulnerability in cases where the vulnerability is not a false positive.

Mitigation of third-party library vulnerabilities is managed through the regular release planning process with priority given to the vulnerabilities that have a critical CVSS v3 score after the vulnerability assessment.

7 NSP communication

7.1 Overview

7.1.1 NSP networking options

The NSP has various internal and external communication requirements. Unless otherwise configured, an NSP system uses one management network for all communication. For greater security, the NSP supports the use of multiple interfaces on each component that define separate networks for the client, mediation, and internal traffic, as required.

i **Note:** The NSP supports the use of IPv4 and IPv6 in the client, internal, and external networks; see “IP version support” in the *NSP Installation and Upgrade Guide* for more information.

An NSP component may communicate over one or more of the networks, depending on the services that the component provides. For example, a component that communicates only with other NSP components uses only the internal network; a component that provides mediation and a client API uses all three networks.

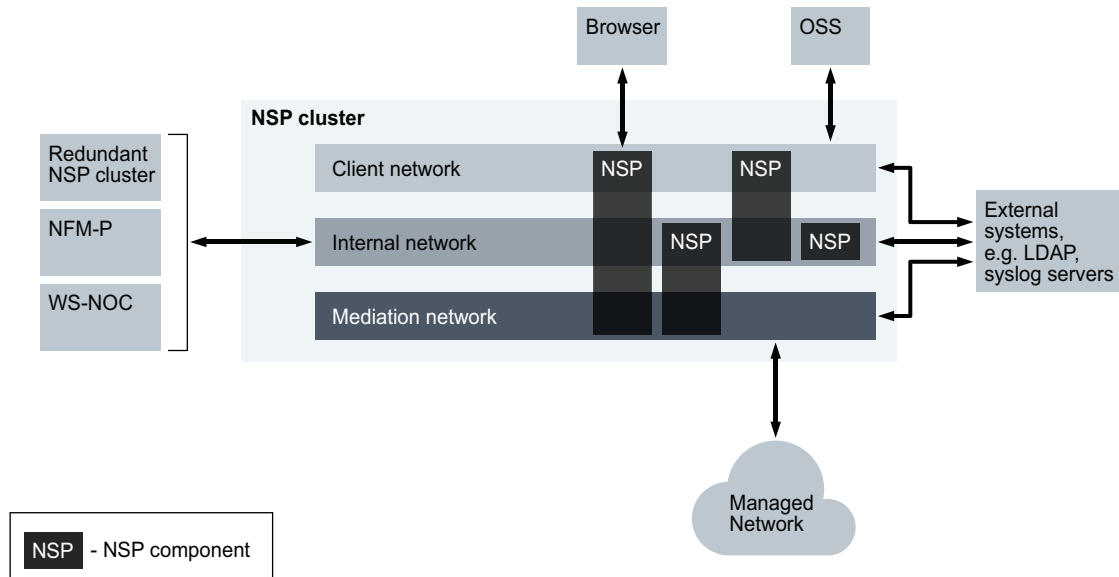
During system deployment, you can enable the use of separate networks by specifying IP addresses for the client, mediation and internal traffic, as required. If you opt to use only one interface per component, only one network is used.

The following are the networks that you can establish in the NSP management domain:

- client network—for NSP client browser, GUI, or API access
- mediation network—for direct communication with managed NEs
- internal network—for communication among NSP components

The following figure shows a high-level view of the NSP multi-interface implementation.

Figure 7-1 NSP multi-interface communication



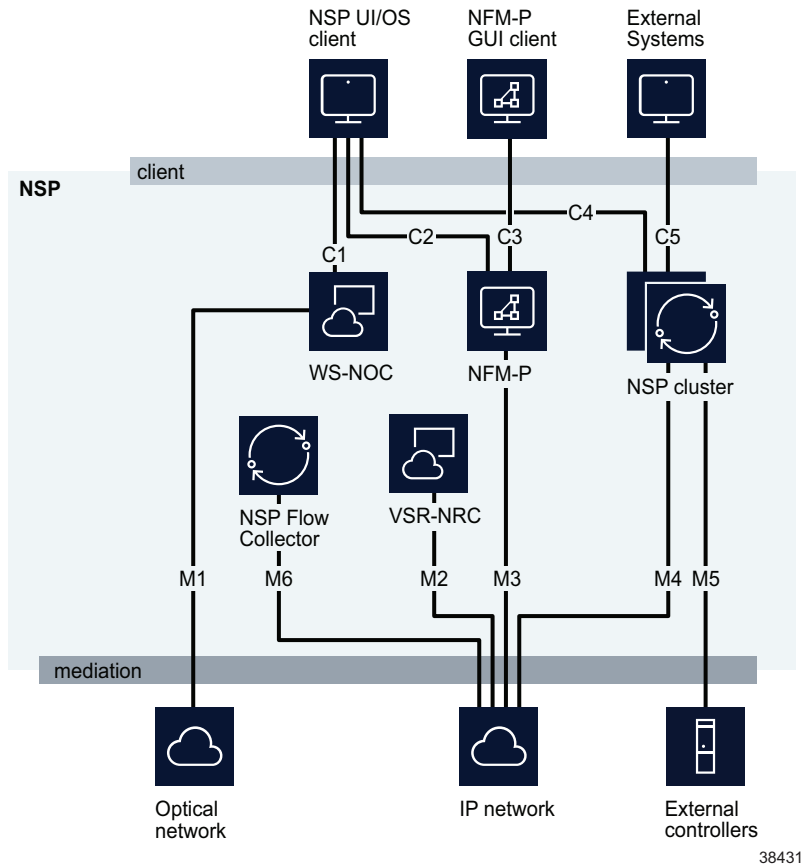
38430

7.2 External NSP communication

7.2.1 Introduction

The NSP provides a number of external interfaces to the client and mediation networks, as shown in the following figure. Each external interface is secured using TLS.

Figure 7-2 External NSP interfaces



7.2.2 Client network

The following table lists the consumers associated with each NSP client interface shown in [Figure 7-2, “External NSP interfaces”](#) (p. 51).

Table 7-1 NSP client interfaces and consumers

Interface	Consumers
C1, C2, C4	Web clients NSP OSS clients NFM-P OSS clients
C3	NFM-P GUI clients

Table 7-1 NSP client interfaces and consumers (continued)

Interface	Consumers
C5	External Syslog server External authentication source External e-mail server

7.2.3 Mediation network

The following table lists the function of each NSP mediation interface shown in [Figure 7-2, “External NSP interfaces”](#) (p. 51).

Table 7-2 NSP mediation interfaces and functions

Interface	Function
M1	Optical network mediation using SNMP, TL1
M2	VSR-NRC network mediation using BGP, PCEP, PCEPS
M3	Legacy network mediation using SNMP, TL1
M4	IP NE mediation using gRPC/gNMI, SNMP, NETCONF, SSH NE communication using CLI via SSH
M5	Mediator communication with external controller REST/RESTCONF API
M6	AA Cflowd or System Cflowd statistics using secure file transfer

7.3 Internal NSP communication

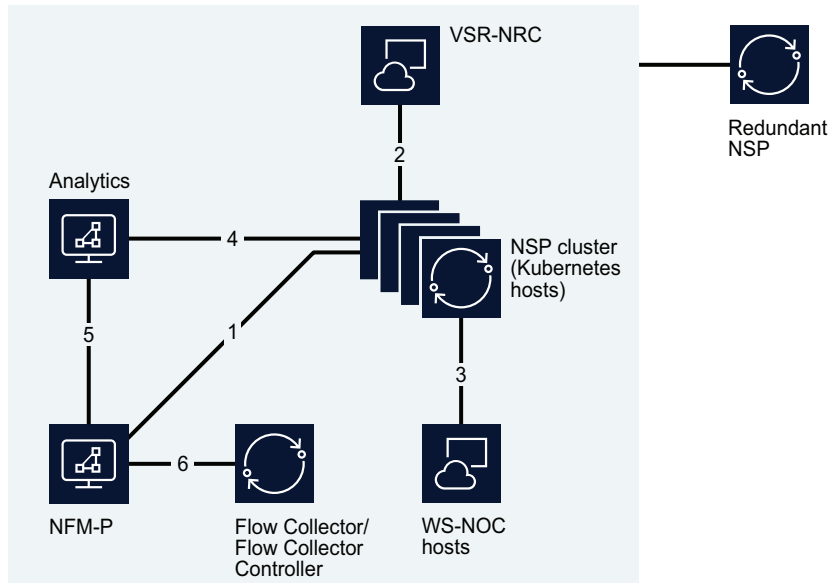
7.3.1 Introduction

Communication among NSP components in a data center is performed over the internal network. Each internal interface is secured using TLS.

i **Note:** TLS is also applied to the communication among internal NSP system services and processes. For additional robustness, the system-level communication uses a certificate from an internal root CA that is inaccessible to external agents.

Each NSP cluster member has a unique IP address; however, the cluster members share one virtual IP address.(VIP) that is assigned to the cluster for access by other components.

Figure 7-3 Internal NSP interfaces



36822

The following table describes each interface shown in Figure 7-3, “Internal NSP interfaces” (p. 53).

Table 7-3 Internal NSP connections

Interface	Endpoints	Traffic type
1	NFM-P NSP cluster	SSO authentication. Zookeeper registration, Kafka messaging, NFM-P API
2	NSP cluster VSR-NRC (BOF interface)	Data connection – CROTO (non-secure)
3	WS-NOC NSP cluster	SSO authentication, Zookeeper registration (non-secure), REST over HTTPS, proprietary WS-NOC HTTP communication
4	NSP cluster Analytics hosts	SSO authentication, Zookeeper registration, Kafka messaging, PostgreSQL transactions
5	NFM-P Analytics hosts	TLS-secured JDBC and REST
6	NFM-P NSP Flow Collectors / Flow Collector Controllers	HTTPS, JMS

7.3.2 Communication between redundant data centers

NSP geographic redundancy, sometimes called geo-redundancy, uses a disaster-recovery (DR) mechanism to engage an identical NSP cluster at a standby site to support service continuity in the event that the primary site suffers an unrecoverable failure. To effectively support a DR NSP deployment, the NSP clusters at the geo-redundant sites require robust, secure communication.

The following NSP components perform DR communication over the internal NSP network:

- NSP cluster
- NFM-P
- WS-NOC

For more information about DR deployments, see [Chapter 8, “System redundancy and fault tolerance”](#).

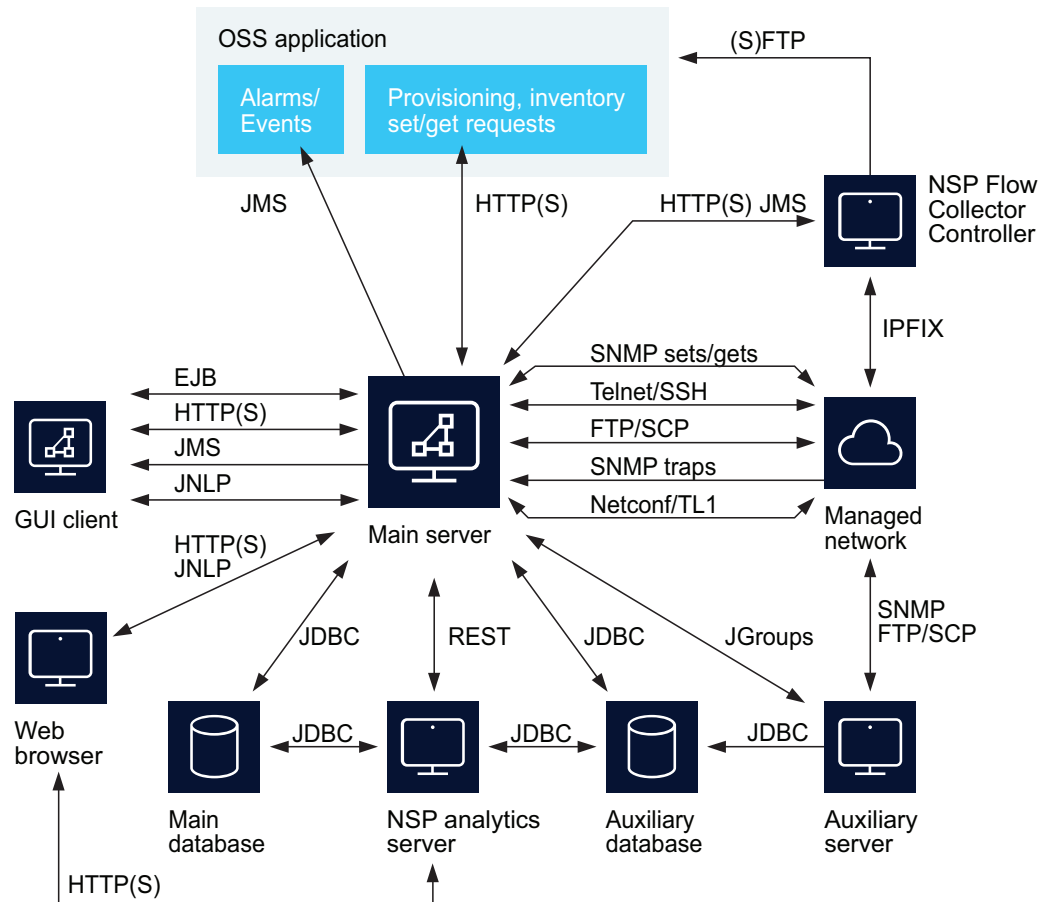
7.4 NFM-P communication

7.4.1 Introduction

Communication among NFM-P components and external systems uses IPv4 or IPv6 exclusively, with the following exceptions:

- The NFM-P can communicate with and manage a network using IPv4 and IPv6 concurrently.
- An NFM-P GUI or browser-based client can connect to the NFM-P using IPv4 or IPv6, regardless of the protocol version in use between the NFM-P server and database components.

Figure 7-4 NFM-P component communication



25436

7.4.2 Servers and managed NEs

NFM-P main and auxiliary servers communicate directly with the managed network.

- A main server uses SNMP to monitor and manage network performance, and to deploy configuration changes to NEs.
- An auxiliary server polls NE MIBs for performance statistics. The NEs use asynchronous SNMP messages called traps to notify the NFM-P of events. UDP streaming is used by NEs for some data transfers to the NFM-P.

The CLI of a managed NE is accessible from the client GUI using Telnet or SSH.

The NFM-P uses protocols such as FTP, SFTP, and SCP to back up NE configuration data, collect NE accounting statistics, and download software to NEs.

7.4.3 Main server and clients

Client interfaces provide access to an NFM-P system and the managed network through a main server.

A main server and clients communicate in the following ways:

- GUI clients send requests to the server EJB session beans using Java RMI.
- The GUI client update function uses HTTP or HTTPS for client software updates and file downloads.
- NFM-P browser clients use HTTP or HTTPS to communicate with the web service on a main server.
- A web-based GUI client communicates through a browser using JNLP.
- XML API clients send requests for processing by a main server, and subscribe to JMS topics to receive real-time event notifications. The messages between a main server and an XML API client are in XML/SOAP format, and are sent over HTTP or HTTPS. The JMS and the XML publisher service on a main server run in separate JVMs to support multiple concurrent client connections. See the *NSP NFM-P XML API Developer Guide* for more information about the messaging between XML API clients and main servers.
- REST API clients perform network management functions and receive notifications using the NFM-P REST API. See the online REST API documentation for information.

7.4.4 Main server and main database

A main server communicates with a main database instance using JDBC, a Java API for interworking with SQL relational databases.

7.4.5 Main server and auxiliary servers

A main server sends requests to auxiliary servers. An auxiliary server notifies the main server after it finishes processing a request. If the main server fails to send a request, or all auxiliary servers are unresponsive to a request, the main server raises an alarm.

7.4.6 NFM-P integration with external systems

The NFM-P can be integrated with external network management systems for purposes such as alarm forwarding. Depending on the external system type, you can use a client GUI contextual menu option to open a session on the external system. See the *NSP NFM-P User Guide* for information.


8 System redundancy and fault tolerance

8.1 Redundancy mechanisms

8.1.1 Disaster recovery

A disaster-recovery (DR) NSP deployment consists of identical primary and standby NSP clusters and ancillary components in separate data centers. One cluster has what is called the primary role, and processes all client requests.

The standby NSP cluster in a DR deployment operates in warm standby mode. If a primary cluster failure is detected, the standby automatically initializes as the primary, and fully assumes the primary role.

 **Note:** In a DR deployment, it is strongly recommended that all primary components are in the same physical facility. An NSP administrator can align the NSP component roles, as required.

NSP Role Manager

In a DR NSP deployment, the Role Manager runs in an NSP cluster and acts as a Kubernetes controller. The Role Manager monitors the Kubernetes objects for changes, and updates the objects as required based on the current primary or standby site role.

The Role Manager has the following operation modes:

- standalone—The Role Manager sets the cluster mode to 'active' at initialization time, and does nothing more.
- DR—The Role Manager negotiates the local role with the DR peer.

The Role Manager uses the configuration in the **dr** section of the NSP configuration file to identify the local and peer sites.

The NSP monitors the following NSP base services in a DR deployment:

- Kafka
- Keycloak
- NSP Tomcat
- nspOS Tomcat
- PostgreSQL
- Prometheus
- ZooKeeper

DR fault conditions

If any service in a DR deployment is unavailable for more than three minutes, or two instances of a service in an HA+DR deployment are unavailable for more than three minutes:

- An activity switch occurs; consequently, the peer NSP cluster assumes the primary role.

- An alarm is raised against the service or containing pod to indicate that the service or pod is down.
Note: Such an alarm may not be generated because of a base service disruption, depending on the circumstances.
- A major ActivitySwitch alarm is raised against the former active site, which is now the standby site.

The following are the alarms that the NSP raises against the NmsSystem object in response to such a failure:

- ActivitySwitch—severity Major
- NspApplicationPodDown—severity Critical

i **Note:** If you clear an alarm while the failure condition is still present, the NSP does not raise the alarm again.

The following example describes an alarm condition in a simple DR deployment.

1. An nspOS service at the primary site fails.
2. An activity switch occurs; the standby site consequently assumes the primary role.
3. A major ActivitySwitch alarm is raised against the former primary site, which is now the standby site.

8.1.2 High availability

NSP cluster deployment supports high availability of critical services through replica pods in a containerized deployment. Select pods are deployed with multiple replicas.

The containerized NSP cluster VMs support HA deployment.

i **Note:** In an enhanced/HA deployment, if node4 were to go down due to an ungraceful shutdown (such as a power outage), a switchover would be triggered.

High availability and NSP file service

When an NSP cluster is deployed with HA and the active nsp-file-service pod restarts, or when a switchover to the standby pod occurs, the NSP is not immediately available to service incoming file service requests.

The NSP file service requires several minutes to recover from a pod restart or switchover. Until the primary pod is fully initialized, the NSP rejects incoming file-service requests, which must be retried when the primary pod is available.

In the event of an NSP file-service pod switchover, the NSP raises the following alarm:

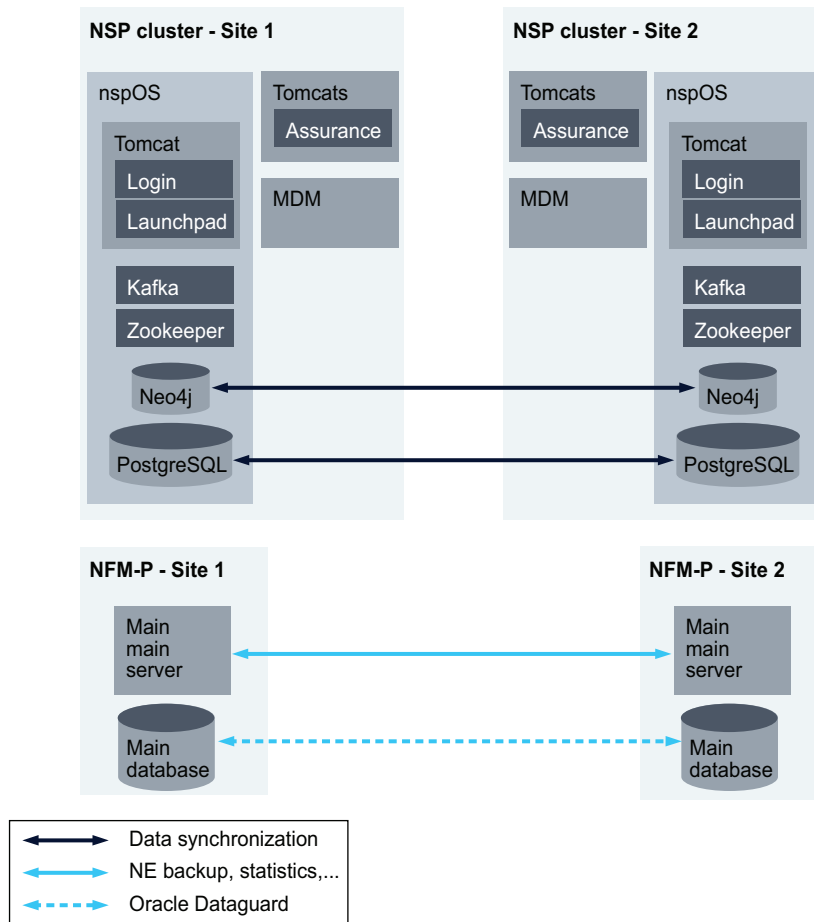
- fileServicePodSwitchOver

8.1.3 Redundancy in shared-mode deployments

To deploy the NSP as a redundant shared-mode system, each component of the system must be redundant. For example, if a redundant NSP deployment includes the NFM-P, the NFM-P must be deployed as a redundant system.

The following figure shows a simple NSP DR deployment.

Figure 8-1 NSP DR deployment



28347

8.2 NSP system failure and recovery scenarios

8.2.1 Introduction

The following topics describe the NSP recovery actions in the event of a redundancy failure; a failure scenario may apply to multiple deployment configurations; the following scenarios are examples only.

8.2.2 Primary NSP cluster failure

The redundant nsp-role-manager agents exchange a heartbeat every five seconds. If the agent on the standby cluster does not receive a heartbeat within 60 seconds, the standby cluster is promoted

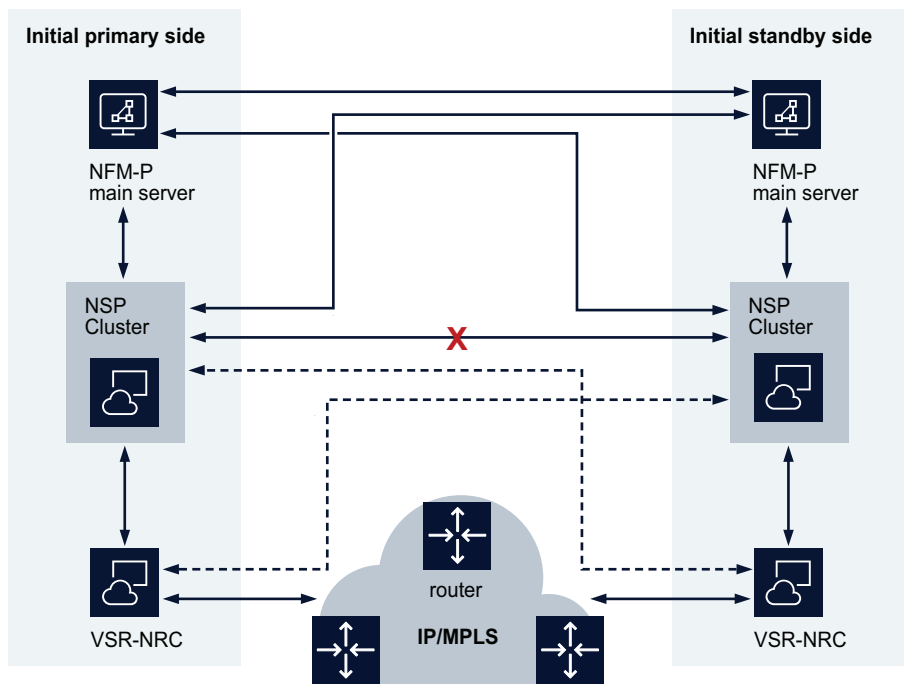
to primary. The new primary cluster subsequently communicates with the NFM-P and the newly active VSR-NRC. Primary cluster base services that stop running can also trigger a switchover.

8.2.3 NSP cluster communication failure

When communication between the NSP clusters fails, each NSP cluster assumes the active role, which creates what is called a split-brain scenario. A 60-second loss of communication between the primary and standby NSP clusters may trigger a switchover.

After communication in a split-brain scenario is restored, the NSP cluster with the higher uptime value assumes the primary role, and the peer cluster assumes the standby role. The assumption is that the cluster running for the longer time was the primary cluster at the time of the loss. In such a scenario, the clients continue to communicate with the same primary cluster.

Figure 8-2 Primary and standby NSP cluster communication failure



28343

8.3 MDM fault tolerance

8.3.1 HA MDM deployment

The NSP supports HA deployment of MDM instances. If an instance fails, the remaining MDM instances assume the mediation functions. An MDM instance is considered failed if both the number of protector pods reaches zero and the number of active MDM pods is fewer than the number provisioned.

A failed MDM instance that is restored to service assumes the standby role.

Note: When an NSP switchover occurs, an MDM switchover also occurs.

8.4 NFM-P fault tolerance

8.4.1 Introduction

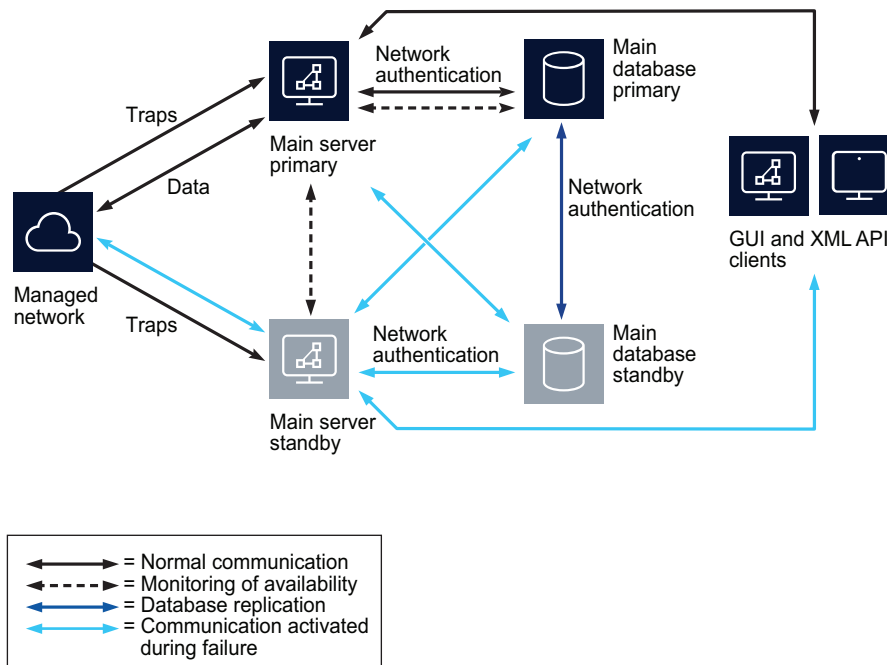
The NFM-P uses component redundancy to ensure that there is no single point of NFM-P system failure.

Redundant physical network interfaces and points of network entry ensure that there is no single point of failure between the NFM-P system and the managed network. Redundant network paths, for example, in-band and out-of-band management, can help to prevent the isolation of a main server from the network in the event of a routing failure.

8.4.2 Main server and database redundancy

A redundant NFM-P system consists of a primary main server and primary main database that actively manage the network, and a second main server and database in warm standby mode. The following figure shows a distributed NFM-P system in a redundant deployment.

Figure 8-3 Redundant NFM-P system



17903

Main server redundancy

Main server redundancy is achieved using clustering technology provided by a JBOSS application server on each main server. The primary and standby main servers regularly poll each other to monitor availability. Traps from the managed network are always sent to both main servers in order to avoid delays in the event that a main server fails.

If the primary server loses visibility of the standby server, it notifies the GUI clients. If the standby server loses visibility of the primary server, the standby server attempts to become the primary server by connecting to the primary database.

Main database redundancy

NFM-P database redundancy uses Oracle Data Guard Replication in real-time apply mode to keep the standby database synchronized with changes in the primary database. The supported fault-recovery operations are database switchovers and database failovers. A switchover is a manual operation that switches the primary and standby database roles. A failover is an automatic operation that forces the standby database to become the primary database when a primary main database failure is detected.

The primary main server regularly polls each main database. If the primary or standby database is unavailable, the main server notifies the GUI clients. If both main servers lose contact with the primary main database, a failover occurs and the standby main database assumes the primary role.

8.4.3 Auxiliary servers and NFM-P redundancy

Auxiliary servers are passively redundant. They do not cause or initiate main server or database redundancy activities, but if a Preferred auxiliary server ceases to respond to requests from the primary main server and a Reserved auxiliary server is available, the main server directs the current and subsequent requests to the Reserved auxiliary server until the Preferred auxiliary server is available.

An auxiliary server communicates only with the current primary server and database. After an NFM-P redundancy activity such as a database failover, the primary main server directs the auxiliary servers to communicate with the current primary component instead of the former primary component.

8.5 VSR-NRC fault tolerance

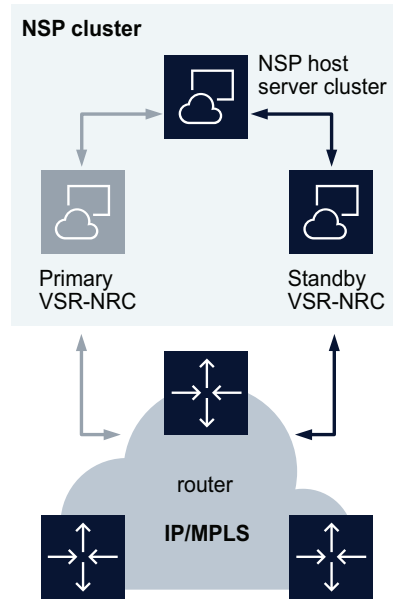
8.5.1 Overview

The NSP supports single-site and dual-site deployments of redundant VSR-NRCs.

Single-site VSR-NRC deployment

The primary VSR-NRC communicates with the NSP cluster. If the primary VSR-NRC is unavailable, the NSP cluster initiates communication with the VSR-NRC in the same data center.

Figure 8-4 Primary VSR-NRC failure in single-site deployment

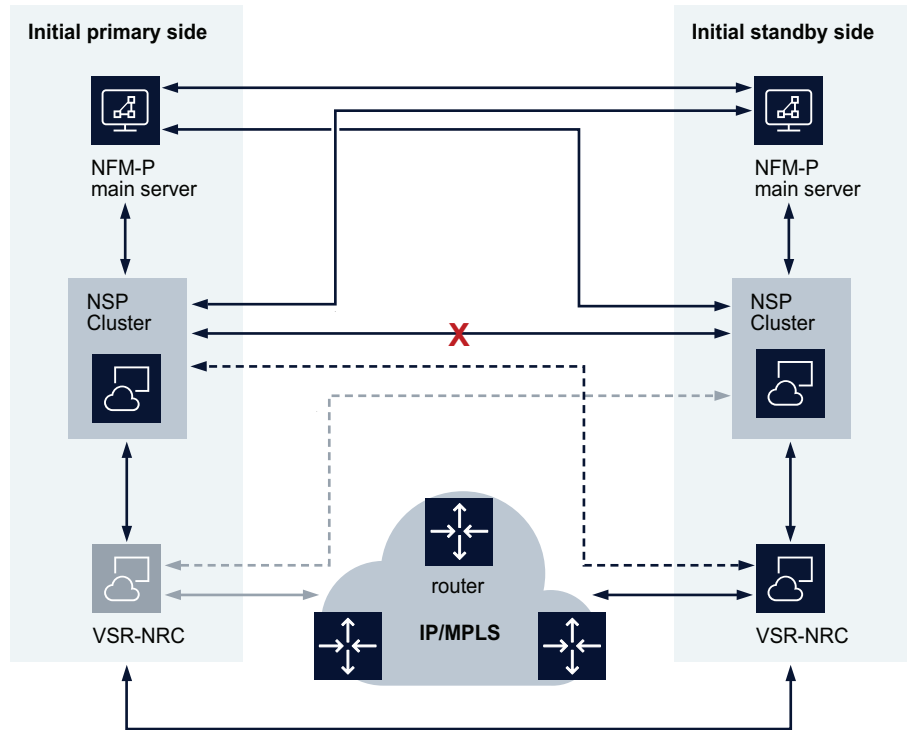


36077

Dual-site VSR-NRC deployment

The primary VSR-NRC communicates with the primary NSP cluster. If the primary VSR-NRC is unavailable, the primary NSP cluster initiates communication with the standby VSR-NRC at the standby site.

Figure 8-5 Primary VSR-NRC failure in a dual-site deployment



28345

The two VSR-NRCs synchronize their databases to allow for faster synchronization after a VSR-NRC failure.

TE-DB and LSP-DB synchronization

Full synchronization with the TE-DB and LSP-DB occurs whenever connection to a primary VSR-NRC is successful. In other words, if the primary VSR-NRC becomes unavailable and the primary NSP cluster begins to communicate with the standby VSR-NRC, the standby VSR-NRC is promoted to primary, and full synchronization with the TE-DB and LSP-DB subsequently occurs.

PCC client communication

PCC clients can define primary and secondary VSR-NRC IP addresses, which enables the PCCs to maintain communication without reconfiguration when the primary VSR-NRC becomes unavailable.

8.6 NSP Flow Collector and Flow Collector Controller fault tolerance

8.6.1 Description

You can deploy redundant NSP Flow Collector Controllers in separate data centers to manage local Flow Collectors, or a common set of Flow Collectors that are deployed in logical proximity to the

managed NEs, and not assigned to a data center. The latter configuration essentially eliminates switchover latency and any resulting data loss in the event of a Flow Collector Controller failure. The configuration also has no dependency on which Flow Collector Controller currently has the primary role.

The NSP Flow Collector Controllers continuously monitor the active nspOS; if the connection is lost or re-established after a connection loss, the Flow Collector Controller primary and standby roles are re-evaluated. In a DR deployment, redundancy is ensured only when the two Flow Collector Controllers are deployed in separate data centers, in order that the failure of one data center does not cause both Flow Collector Controllers to be unreachable.

Redundant NSP Flow Collectors in a data center can collect statistics from the same set of NEs to provide data-collection fault tolerance. The statistics that are stored in a database are de-duplicated beforehand; only one set is stored. If statistics are stored in files, duplicate files are created.

In an HA or HA+DR deployment that includes the NFM-P, the primary Flow Collector Controller connects only to the primary NFM-P. An internal mechanism ensures that the primary Flow Collector Controller in a DR deployment is always in the same data center as the primary NFM-P.

i **Note:** An NSP switchover or failover disrupts the NSP API communication with one or more Flow Collectors in SYS mode. As a result, Flow Collector log entries like the following are recorded at a high frequency:

```
Connection to node node (IP_address) could not be established. Broker  
may not be available
```

The issue is not service-affecting. To stop the recording of the log entries, you must restart the Flow Collector.

Remote statistics transfer

An NSP Flow Collector can transfer collected statistics files to redundant remote servers. For greater fault tolerance, redundant Flow Collectors can collect statistics from the same set of NEs and transfer the files to redundant destination servers. Such a configuration ensures that the statistics collection and transfer continue uninterrupted in the event that an NSP Flow Collector and a transfer destination are each unreachable.

8.7 Analytics server fault tolerance

8.7.1 Description

NSP analytics servers use the following fault-tolerance mechanisms to ensure that there is no single point of failure or server unavailability:

- multiple analytics servers in an active/active configuration, and a load-balancing algorithm that specifies which server responds to a client request
- access to redundantly deployed data sources

A NSP technology standards

A.1 NSP technology standards

A.1.1 Industry standards and open-standard interfaces

The NSP incorporates industry standards and open-standard interfaces that allow interoperation with other network monitoring and management systems. The following table lists and describes the technology standards and interfaces that are represented in the NSP system design.

Table A-1 Industry standards consulted in NSP design

Standard or interface	Description
draft-alvarez-pce-path-profiles-04	PCE path profiles
draft-dbwb-opsawg-sap-02	A network YANG model for Service Access Points (SAPs)
draft-ietf-i2rs-yang-network-topo-20	A data model for network topologies
draft-ietf-idr-bgp-ls-segment-routing-ext-16	BGP link-state extensions for segment routing
draft-ietf-idr-bgp-ls-segment-routing-msd-09	Signaling MSD using BGP-LS
draft-ietf-isis-mi-02	IS-IS Multi-Instance
draft-ietf-isis-segment-routing-extensions-04	IS-IS extensions for segment routing
draft-ietf-opsawg-l2nm	A YANG network data model for layer 2 VPN network
draft-ietf-ospf-segment-routing-extensions-04	OSPF extensions for segment routing
draft-ietf-pce-segment-routing-08	PCEP extensions for segment routing
draft-ietf-pce-stateful-pce-14	PCEP extensions for stateful PCE
draft-ietf-teas-yang-te-29	A YANG data model for traffic engineering tunnels and interfaces
Keycloak	Version 21.1.1
ONF TR-547-TAPI v2.1.3	Reference Implementation Agreement
OpenFlow	OpenFlow Switch Specification version 1.3.1
REST	Representational State Transfer
RFC 3986	URI Generic Syntax

Table A-1 Industry standards consulted in NSP design (continued)

Standard or interface	Description
RFC 4655	Path Computation Element (PCE) based architecture
RFC 5101	Specification of the IP Flow Information Export (IPFIX) Protocol for the exchange of IP traffic flow information
RFC 5102	Information model for IP flow information export
RFC 5440	Path Computation Element Communication Protocol (PCEP)
RFC 5575	Dissemination of flow specification rules
RFC 6020	YANG data modelling language for NETCONF
RFC 6021	Common YANG data types
RFC 6087	Guidelines for YANG Documents
RFC 6241	Network configuration protocol (NETCONF)
RFC 6242	NETCONF over SSH
RFC 6991	Common YANG data types
RFC 7223	A YANG data model for interface management
RFC 7224	IANA interface type YANG model
RFC 7420	PCEP Management Information Base (MIB) model
RFC 7231	HTTP/1.1 Semantics and Content
RFC 7752	North-bound distribution of link-state and Traffic Engineering (TE) information using BGP
RFC 7950	YANG 1.1
RFC 7951	JSON encoding of data modelled with YANG
RFC 7952	Defining and Using Meta Data with YANG
RFC 8040	RESTCONF
RFC 8072	YANG patch media type
RFC 8281	PCEP extensions for PCE-initiated LSP setup in a stateful PCE model
RFC 8321	PCEP extensions for stateful PCE
RFC 8345	A YANG data model for network topologies
REF 8346	A YANG data model for layer 3 topologies
RFC 8476	Signaling MSD using OSPF (node MSD)

Table A-1 Industry standards consulted in NSP design (continued)

Standard or interface	Description
RFC 8491	Signaling MSD using IS-IS (node MSD)
RFC 8525	YANG Library
RFC 8528	YANG schema mount
RFC 8665	OSPF extensions for Segment Routing
RFC 8667	IS-IS extensions for Segment Routing
RFC 8776	Common YANG data types for traffic engineering
RFC 8944	A YANG data model for layer 2 network topologies
RFC 9181	A common YANG data model for layer 2 and layer 3 VPNs
RFC 9182	A YANG network data model for layer 3 VPNs

A.2 Classic management technology standards

A.2.1 Industry standards

The NFM-P incorporates industry standards and open-standard interfaces. The following table lists and describes the technology standards and interfaces that are represented in the NFM-P design.

Table A-2 Industry standards consulted in NFM-P design

Standard	Description
3GPP	3rd Generation Partnership Project IRPs for CORBA R8 and SOAP/XML R8 Solution Sets
draft-grant-tacacs-02.txt	TACACS+ client
draft-ylonen-ssh-protocol-00.txt	SSH
EJB	Java EE Enterprise Java Session Bean version 2.1
HTML5	HyperText Markup Language 5, for NFM-P applications
HTTP(S)	HyperText Transfer Protocol (Secure) version 1.1
ITU-T X.721	SMI
ITU-T X.734	Event report management function
Java SE	Java Standard Edition version 8
JBOSS EAP	Java Bean Open Source Software Enterprise Application Platform version 7
JMS	Java Message Service version 1.1
JSON	ECMA-404 JavaScript Object Notation Data Interchange Format

Table A-2 Industry standards consulted in NFM-P design (continued)

Standard	Description
JS/ECMAScript 5	ECMA-262 ECMA Script Language Specification
M.3100/3120	Equipment and connection models
MTOSI	Compliance of generic network objects, inventory retrieval, and JMS over XML
RFC 0959	FTP
RFC 1213	SNMPv1
RFC 1738	Uniform Resource Locators (URL)
RFC 2138	RADIUS client 2618
RFC 3411-3415	SNMPv3
RFC 3416	SNMPv2c
RFC 5246	The Transport Layer Security (TLS) Protocol
RFC 6241	Network Configuration Protocol (NETCONF)
SOAP	W3C SOAP 1.2
TMF 509/613	Network connectivity model
TR-069	TR-069 (Amendment 1) by way of the Home Device Manager
Velocity	Velocity Engine 2.3
XML	W3C XML 1.0
	W3C Namespaces in XML
	W3C XML schemas

The following standards are considered in the NFM-P GUI design:

- Sun Microsystems, *Java Look and Feel Design Guidelines*, Addison-Wesley Publishing Company, Reading, Massachusetts 1999
- ANSI T1.232-1996, *Operations, Administration, and Provisioning (OAM&P)- G Interface Specifications for Use with the Telecommunications Management Network (TMN)*
- Telcordia (Bell Core) GR-2914-CORE Sept. 98, *Human Factors Requirements for Equipment to Improve Network Integrity*
- Telcordia (Bell Core) GR-826-CORE, June 1994, Issue 1, Section 10.2 of OTGR, *User Interface Generic Requirements for Supporting Network Element Operations*
- ITU-T Recommendation Z.361 (02/99), *Design guidelines for Human- Computer Interfaces (HCI) for the management of telecommunications networks*
- ETSI EG 201 204 v1.1.1 (1997-05), *Human Factors (HF); User Interface design principles for the Telecommunications Management Network (TMN) applicable to the “G” Interface*
- 3GPP 32-series R8 specification, published December, 2009.