NSP
Network Services Platform
Release 25.11

System Architecture Guide

3HE-21467-AAAC-TQZZA
Issue 4
June 2026

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.

Contents

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 (NSP) is a distributed system that provides comprehensive infrastructure, service management, and resource control within and among multiple network domains.

The NSP manages multivendor 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 multilayer 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 through real-time LSP path control
- 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, as well as timely service performance, usage, and fault information.

1.2.2 Traffic optimization

The NSP automates traffic management by controlling and steering flows through LSP path control as required.

1.2.3 Data collection and analysis

The NSP monitors network key performance indicators (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 NE statistics or 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 deployment dynamically maintains a network equipment data store. 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:

- *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, NSP system configuration, and data backup and restore
- *NSP System Architecture 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 Notice* – contains special or notable information about an NSP release, such as deployment limitations or product enhancements not covered in the core documentation

  This document is available from the Nokia Doc Center only. It is not included in the on-product NSP Help Center with the other documentation.
- *NSP Release Description* – describes the features and functions in an NSP release; includes information such as functional enhancements and resolved product issues
- *NSP User Guide* – describes NSP access and use for operators, introduces the NSP UI functions, and describes how to obtain NSP software and documentation

- *NSP Glossary* – defines acronyms, initialisms, and other terms used in the NSP product interfaces and documentation
- Multiple product- and component-level guides and tools that each describe how to use an aspect of the NSP – see “Documentation architecture” in the *NSP User Guide* for more information about the NSP documentation suite

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

2.2 “Planning your deployment” (p. 14) 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.

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.

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

NSP installation options

The 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 system platform requirements.

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

2.1.2 NSP Platform

The NSP 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
- Certificate Manager
- Logging and Monitoring
- Pluggable Network Adaptation
- Flow Stats Collection

Base Services, Certificate Manager, and Logging and Monitoring are enabled by default.

Pluggable Network Adaptation is required to discover and manage NEs in NSP for all mediation types.

Two mediation types are included in the Platform feature package:

- classic mediation

  Classic mediation provides mediation services for SNMP-managed Nokia NEs via the underlying Network Function Manager - Packet (NFM-P) component.
- model-driven mediation (MDM)

  MDM provides mediation services for NETCONF- and YANG-compatible Nokia NEs and third-party vendor (multivendor) NEs via the Pluggable Network Adaptation option components and the specific MD adaptors.

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

Certificate Manager

Certificate Manager is an installation option of the Platform feature package that provides management of certificates within the cluster. It is mandatory in most deployments but may be optional in some cloud environments.

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 MDM framework and are available to other NSP functions.

This framework can additionally provide REST and RESTCONF interfaces towards domain controllers. The domain controllers can have domain-specific network and service models, for example, Open Networking Foundation (ONF) Transport API (T-API), that enable the multi-domain coordination capabilities in NSP.

Flow statistics collection

The Platform feature package also includes the flowCollection installation option, which enables scalable flow statistics collection. The function employs one or more NSP Flow Collector nodes in an NSP cluster to collect AA cflowd or system cflowd flow records directly from NEs. The flow records can be stored for reporting by NSP functions, such as Analytics, and forwarded to one or more remote target servers for processing by in-house systems or third-party tools.

2.2 Planning your deployment

2.2.1 Feature packages and installation options

An NSP deployment is composed of the NSP Base/Platform feature package, which includes the mediation options Model-Driven (Pluggable Network Adaptation) and/or Classic management, along with the selected feature packages included with your NSP license.

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 (mandatory)	Platform	platform-baseServices
		platform-loggingMonitoring
		platform-certificateManager
Platform (optional)	Platform	platform-flowCollection
		platform-pluggableNetworkAdaptation
Network Operations	Network Infrastructure Management	networkInfrastructureManagement-basicManagement
		networkInfrastructureManagement-deviceConfig
		networkInfrastructureManagement-performanceIndicatorsAndAlerts
		networkInfrastructureManagement-igpTopology
		networkInfrastructureManagement-gnmiTelemetry
		networkInfrastructureManagement-gNMIonChange
		networkInfrastructureManagement-accountingTelemetry
	Service Assurance	serviceAssurance
	Service Activation and Configuration [1]	serviceActivationAndConfiguration-intentBasedServiceFulfillment
	Network Operations Analytics	networkOperationsAnalytics-baselineAnalytics
		networkOperationsAnalytics-analyticsReporting
	Transport Slice Operations	transportSliceOperations

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

Software suite	Feature package	Installation option
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.

2.3 Network Operations

2.3.1 Description

The Network Operations software suite enables operators to manage multivendor networks. 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 enables device discovery, device management, zero-touch provisioning (ZTP), and intent-based device configuration.

Performance Indicators And Alerts

The Performance Indicators and Alerts installation option enables you to define and monitor network KPIs and create threshold-crossing alerts based on the KPIs.

The use of indicators requires telemetry to be configured, which has additional prerequisites; see “What are the best practices for telemetry data collection?” in the *NSP Data Collection and Analysis Guide*.

IGP topology

The IGP Topology installation option deploys the IGP Topology discovery service, which provides the IGP topology information required to populate the IGP topology map in Network Map and Health and the IGP layer in the Object Troubleshooting multilayer map.

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.

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 fewer 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 service is not deployed if any of the following installation options are present: controlAndVisualizationStarter, pathControlAndOptimization, or enhancedOptimization. The functions of these installation options supersedes those of networkInfrastructureManagement-igpTopology.

gNMI onChange

The gNMI onChange installation option enables alarm management via gNMI on change.

gnmiTelemetry

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

Accounting Telemetry

The Accounting Telemetry installation option enables collection of accounting statistics using NSP.

NSP supports accounting telemetry collection using FTP or SFTP for classic NEs and SFTP only for model-driven NEs.

2.3.3 Service Assurance

The Service Assurance feature package provides mediation-agnostic monitoring and troubleshooting of multivendor IP/MPLS and Ethernet services. It also ensures continuous SLA compliance.

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 and the related license enable users to perform create, read, update, and delete operations in NSP';s Service Management views and APIs.

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 functions. The feature package enables the collection, storage, and aggregation of event, performance and volumetric data to present valuable information such as:

- full L0-L7 analytics with detailed visualizations
- real-time hardware inventory and L2/L3 service analysis

Network operations analytics requires telemetry to be configured, which has additional prerequisites; see "What are the best practices for telemetry data collection?" in the *NSP Data Collection and Analysis Guide*.

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 detected as deviations from baselines can act as early warnings of unforeseen network

issues that allow additional mitigation and resolution time. Baselines are configured and visualized using Data Collection and Analysis.

Analytics Reporting

The networkOperationsAnalytics-analyticsReporting installation option enables NSP Analytics, which provides prepackaged reports and dashboards, regardless of the mediation layer and NE operating mode. The reports and dashboards in the Analytics GUI provide insightful immediate and trend-based views of management aspects such as network capacity and performance. Operators can also create new ad-hoc reports and dashboards on demand.

2.3.6 Transport Slice Operations

Transport Slice Operations is a feature package that enables the 5G Transport Slice Controller (TSC) 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 L2/L3 services and tunnels/paths used during the realization of transport slices and provides proactive monitoring telemetry data and reports on transport slices.

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

The Service Assurance feature package is required if you want to use the assurance features on transport slices.

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

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
- Control and Visualization Starter FP + Path Control and Optimization FP + Enhanced Optimization FP (if your deployment uses the extra precision and 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 functions, as well as multilayer IP/optical discovery, visualization, and coordination.

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

Note: High availability is provided for the NSP functions by default with feature packages from the resource control suite.

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.

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.

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.

2.4.5 Enhanced Optimization

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

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

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

- Control and Visualization Starter
- Path Control and Optimization

Note: If your network includes Generation 1 7250 IXR devices, Enhanced Optimization requires the flowCollection installation option in the Platform feature package.

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 to

establish T-API based interfaces towards an optical domain controller. This then enables the IP/optical coordination capabilities.

Note: Although the feature package requires no other dependencies to view the transport topology, including the LAGs, the Control and Visualization Starter feature package is also required to view the IP/MPLS (LSP) topology; contact Nokia Professional Services for guidance.

2.4.7 Multi-layer Control and Coordination

The Multi-layer Control and Coordination feature package supports single-step Link Layer Interconnect (LLI) establishment, maintenance coordination, cross-domain connection management, and floating port restoration. The multilayer control and coordination service is deployed on the NSP cluster.

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

Note: For multilayer optimization use cases, you also require one of the following feature packages:

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

2.5 Network Programming

2.5.1 Description

The Network Programming software suite consists of software for developers and network architects. Using declarative and imperative programming, 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, Operations Support System (OSS), and orchestrators

The Network Programming feature packages are:

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

Note: The Intent Based Networking Framework and Workflow Automation Engine feature packages are required if you intend to modify existing Nokia-provided workflows or intent types, or plan on creating your own workflow or intent types from scratch.

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 (SDK) feature package enables the use of an SDK to facilitate adaptor development in the MDM environment. The NSP SDK is a separate package that is available for download and installation by registered customers from the NSP software delivery site.

2.5.3 Intent Based Networking Framework

The Intent Based Networking Framework feature package enables NSP users to create and manage network and service intents using Network Intents.

2.5.4 Workflow Automation Engine

The Workflow Management Framework feature package enables NSP users to create and manage 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:

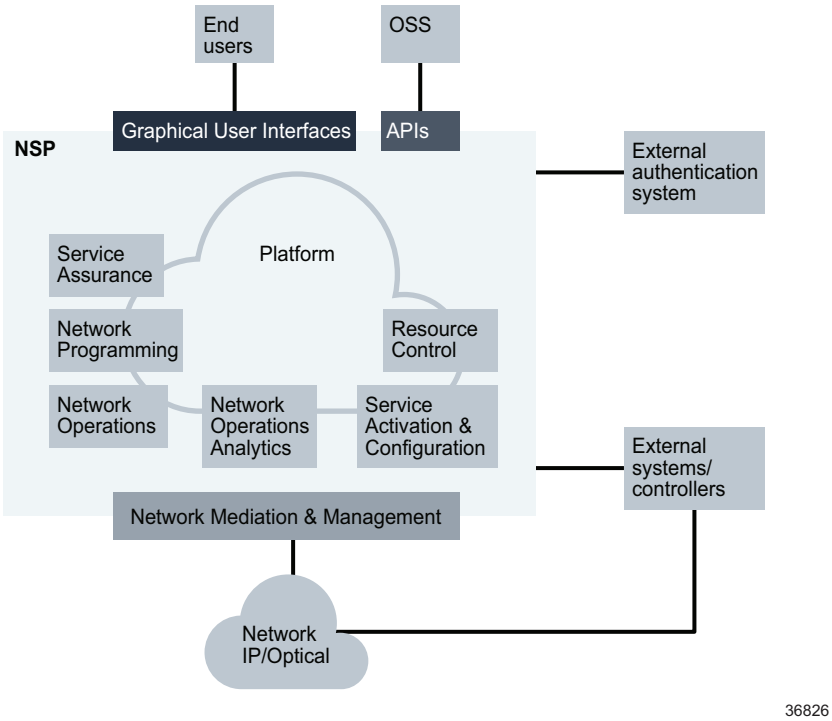
- graphical user interfaces
- public APIs
- management domain or business logic
- mediation services

NSP is designed to interface or integrate with:

- remote authentication agents for user access
- other external systems and controllers

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

Figure 3-1 NSP architecture, conceptual view



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 functions 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 and public APIs

The NSP architecture publicizes the following APIs for OSS clients:

- NSP REST/RESTCONF—for HTTP Create, Retrieve, Update, Delete (CRUD) operations on NSP objects
- NSP Kafka—for event and alarm notifications about the managed network

See the Network Developer Portal documentation for information.

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 Network Developer Portal 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. NSP mediation provides:

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

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, MDM provides mediation services for
  - Nokia and multi-vendor devices managed using YANG model-based interfaces
  - Nokia and multivendor IP devices managed using SNMP
  - T-API
- 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

  VSR-NRC also provides BGP-LS protocol termination from network to NSP and BGP SR policy protocol termination from NSP to SR OS.
- WS-NOC, for optical transport network mediation

Classic management

The NFM-P provides role-based network and service management for classically managed networks.

Note: In addition to managing Nokia devices, you can obtain NFM-P adaptors for managing third-party equipment; contact Nokia for information.

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.

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 shown in Figure 3-2, "NFM-P multi-layer model" (p. 28):

- resource
- integration
- business
- presentation
- client

Resource layer

The resource layer includes the network of managed NEs, the main database, and optional components such as auxiliary servers and an NSP 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 Java Database Connectivity (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 Enterprise Java Bean (EJB)-method invocations from GUI clients and returns the responses from the business layer; the server also forwards Java Message Service (JMS) messages from the business layer to GUI and OSS clients

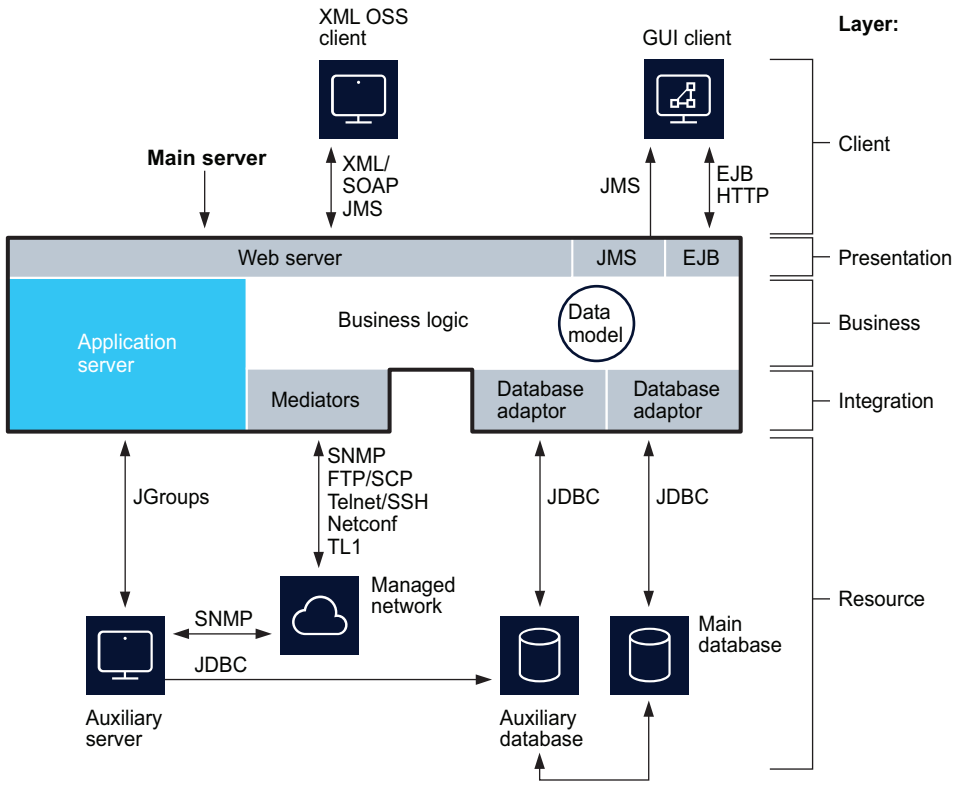
Client layer

The GUI, OSS, and browser-based clients comprise the client layer. A Java VM on a GUI client sends EJB remote method invocations to a main server. The OSS clients send XML, SOAP, or REST messages to a main server.

Multilayer model illustration

The following figure shows the logical organization of NFM-P system elements.

Figure 3-2 NFM-P multi-layer model



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-based networking framework

  Abstracts the intricacies of network configurations by providing network operators with the ability to initiate infrastructure and service configurations using intents.

- 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.

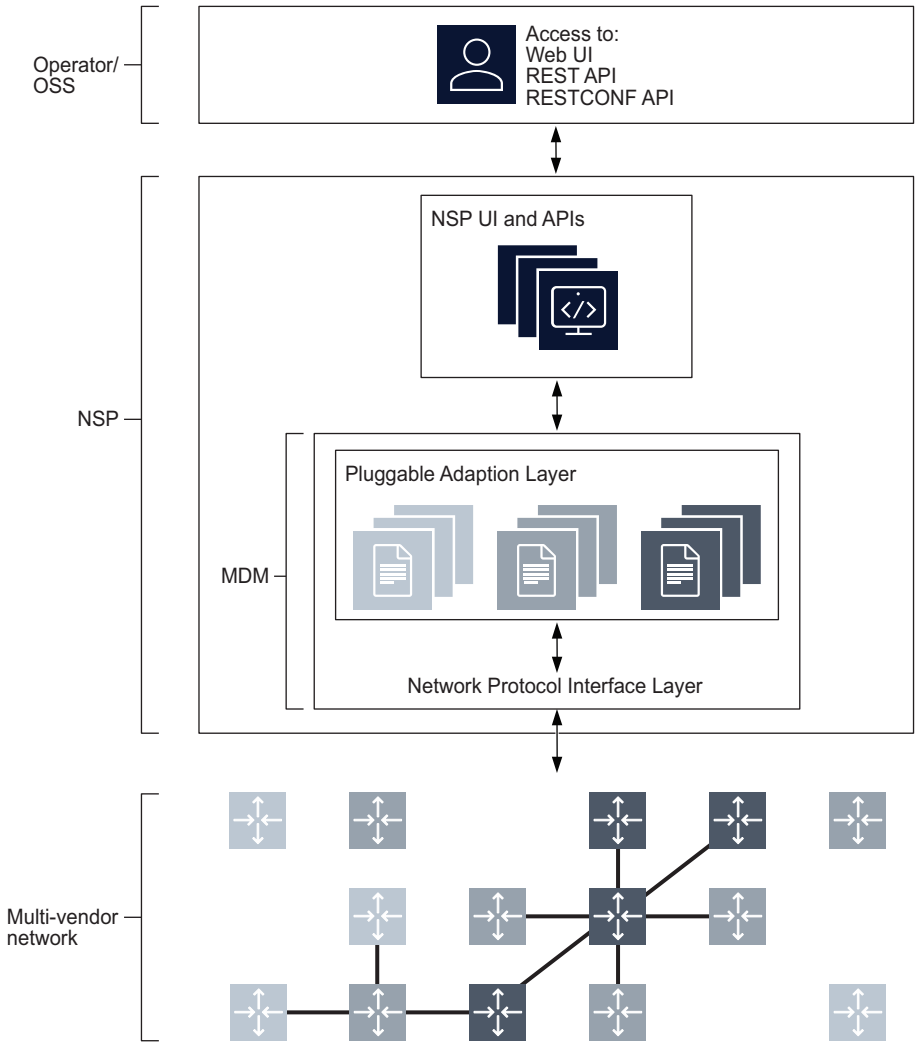
MDM translates network inputs for NSP functions, and creates inputs to manage the NE configuration, state, and telemetry.

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.

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 the functional scope can expand at any time without the need to upgrade the NSP or a device.

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 multivendor devices using Netconf/YANG.

Figure 4-1 Model-driven mediation in NSP



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 based networking framework

4.4.1 Description

NSP's intent-based networking framework provides the capabilities to plan and design configurations at a network level. An intent based system implies a clear separation between expressing the desired end-state (the intent) and the actions to reach that state. The configurations provided in the intent are automatically translated to the network, putting it in the desired state along with policy enforcement.

The intent-based networking framework allows a high level configuration goal to be defined using an intent type. The intent type defines the parameters for the service and provides mappings for device models. The intent type can be saved in the NSP Network Intents views for future use. The Network Intents database can also store service topology and configuration information for later deployment, and schedule audits of the network against the saved configuration requirements.

An intent is an instance of the intent type with any required inputs provided. Intents are customizable and can be deployed during runtime.

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 model-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.

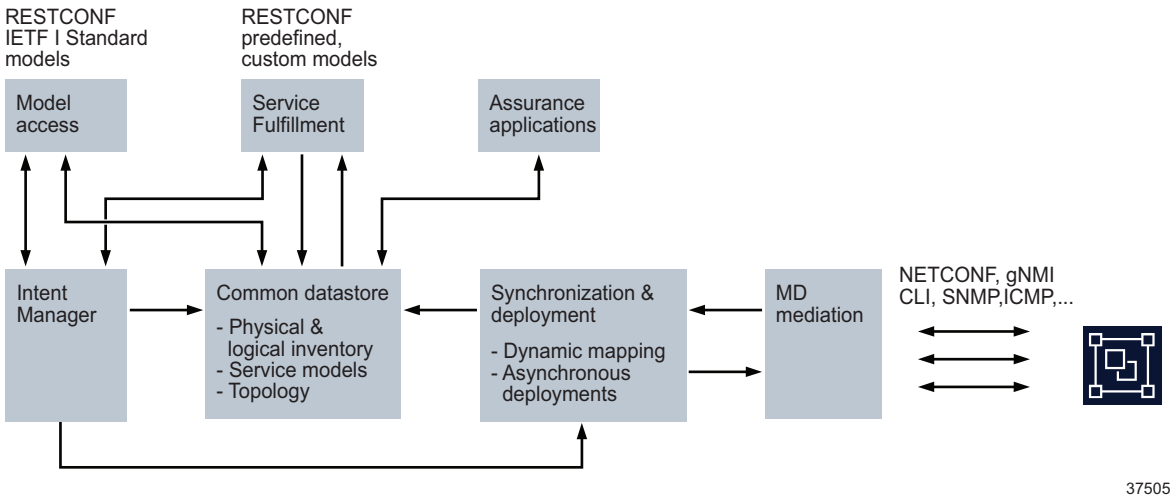
The NSP features an IETF framework that functions as a programmable northbound mediation layer, allowing for the import of IETF-based models into the NSP. CRUDL operations are

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.



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. These can be obtained from the Nokia Support Portal at the following locations:

1. NSP/*release*/Artifacts/*vendor*, where *vendor* is the supported device type. — This directory hosts artifacts available for production NSP deployments. It contains the service management artifact bundles required for IETF services.
2. NSP/*release*/Beta_Artifacts — This directory hosts artifacts that come with certain limitations.

   Most IETF artifacts can be found here, along with their readme documents when available. IETF artifact bundles and readmes include IETF in the filename.

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. Visit the Network Developer Portal for more information about the NSP's RESTCONF APIs.

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

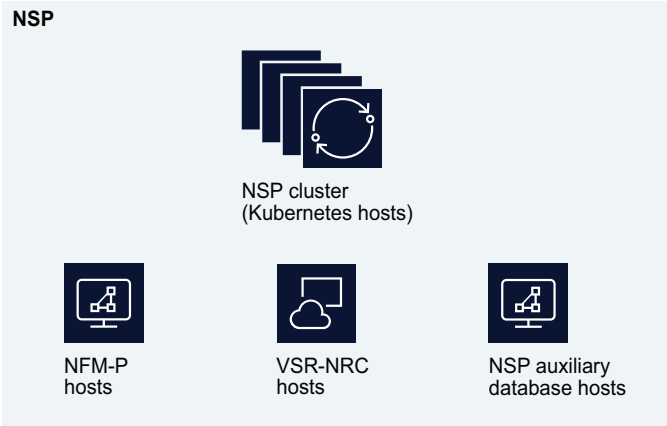
The core elements of the NSP platform are deployed in a Kubernetes container environment that Nokia provides. Other components of an NSP system are deployed outside the Kubernetes environment.

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
- optionally, 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 auxiliary database

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



5.2 NSP deployment environment

5.2.1 NSP containerized infrastructure

The NSP platform is composed of virtual components in a containerized environment:

- **NSP deployer host**: a small-profile VM which deploys the container runtime environment for the NSP cluster

  The NSP deployer host creates a Kubernetes cluster according to your NSP deployment specifications, and subsequently deploys the NSP software to the cluster.

- **NSP cluster**: one or more VMs which together host the NSP software and functions

  An NSP cluster may consist of one member, or three or more members that each host a portion of the installed NSP software.

  - **NSP cluster member**: a Kubernetes worker node VM in an NSP cluster. An NSP cluster member is also called a cluster node, depending on the context.
  - **NSP cluster host**: a specific NSP cluster member from which NSP deployment operations in the cluster are performed. Typically, node 1 of a cluster is chosen as the NSP cluster host and becomes a Kubernetes control node.
  - **Pods**: Kubernetes pods devoted to distinct NSP services are distributed among VMs in the cluster.

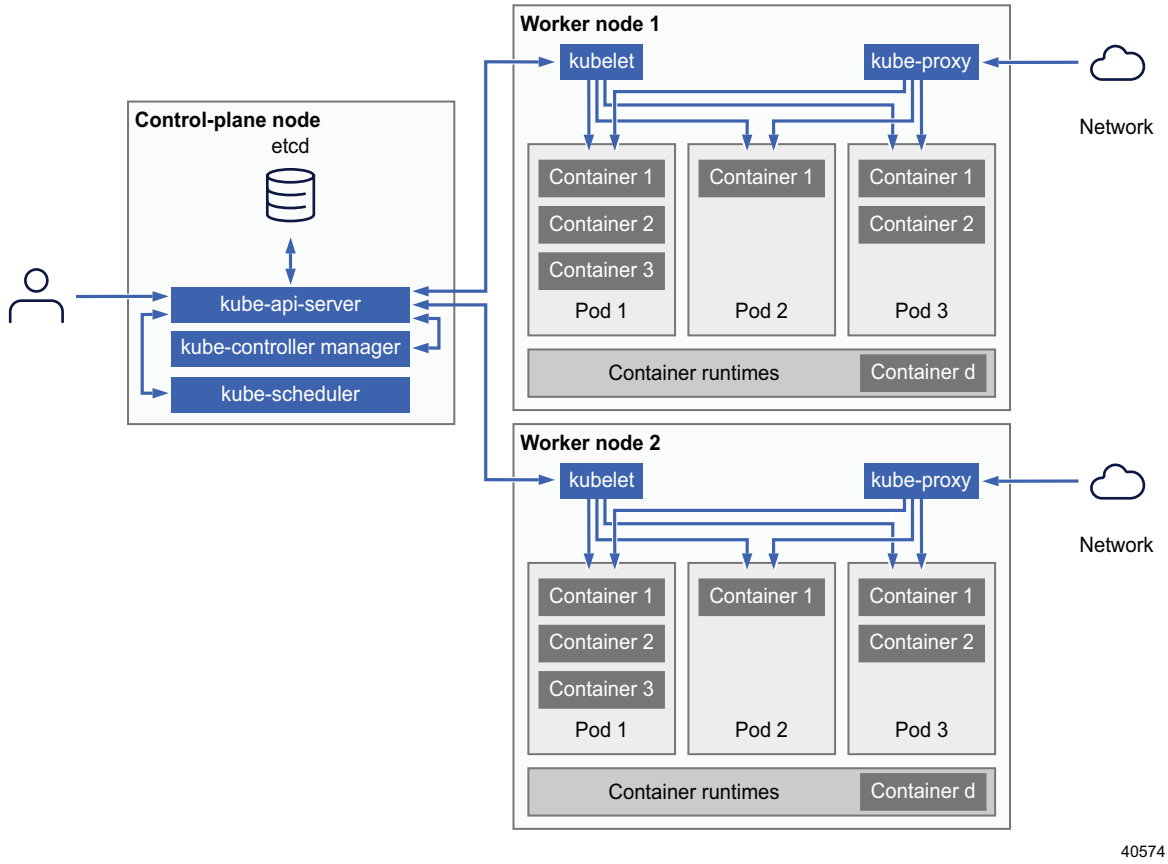
    The NSP software runs in pods on the NSP cluster nodes.

For information about NSP deployment and working with NSP deployers, see “NSP deployment terms and concepts” and “NSP deployment basics” in the *NSP Installation and Upgrade Guide.*

For information about how to manage NSP clusters and pods, see “NSP cluster administration” in the *NSP System Administrator Guide.*

The following figure shows a high-level conceptual diagram of a Kubernetes cluster.

Figure 5-2 Kubernetes cluster



5.2.2 Storage

When NSP is deployed in a single node cluster, it uses the storage local to the node.

A multi-node NSP cluster uses shared storage for RWX and Kubernetes local-provisioner storage for RWO, since NSP pods are deployed across the cluster members.

If an NSP cluster has more than three nodes the last three nodes are used for NSP-provided storage; in a two- or three-node cluster all nodes are used for NSP-provided storage.

5.3 Deployment options

5.3.1 Deployment types

The following table describes the NSP deployment types available in the nsp-config.yml file. See your NSP Platform Sizing Response for information about the correct deployment type for your requirements. See “NSP cluster requirements” in the *NSP Planning Guide* for platform requirements.

NSP provides redundancy options through disaster recovery (DR) and high availability (HA). All deployments have the option of DR, where NSP is deployed in identical redundant clusters in geographically separate data centers. See Chapter 8, “System redundancy and fault tolerance” for more details about redundancy.

Deployment type	Description	Redundancy options
Customer-lab	Lab deployment	Option for DR
Small	Production deployment for small enterprise networks, single node cluster	
Medium	Production deployment for medium sized networks	
Basic	Production deployment with a single node cluster	
Standard	Production deployment	
Enhanced	Production deployment always deployed in multi-node cluster for HA	HA and option for DR Enhanced is the only deployment type that includes HA.
Special deployments	Deployments that only include the NRC-P Simulation tool or Centralized License Manager are standalone only	None

5.4 VSR-NRC

5.4.1 Description

The VSR-NRC is the mediation component of NSP for discovering IGP information from the network and mediating PCEP and BGP for managing Path Control operations. The VSR-NRC is a virtual SR OS instance based on the VSRi architecture. For platform requirements and deployment information, see the *VSR VNF Installation and Setup Guide*.

5.5 NSP auxiliary database

5.5.1 Description

An NSP auxiliary database is an optional, horizontally scalable database that expands the 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 the geo-redundant deployment of identical auxiliary database clusters in separate data centers.

5.6 Classic management components

5.6.1 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

5.6.2 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.

5.6.3 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.

5.6.4 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.

5.6.5 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.

5.6.6 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.

5.6.7 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

5.7 Optical management

5.7.1 Description

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

The WS-NOC application in Wavesuite 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 Platform Feature Guide* for more information.

6 Security architecture

6.1 NSP system security

6.1.1 TLS

An NSP system is secured using Transport Layer Security (TLS). TLS ensures secure external communication between NSP UI and other clients and the NSP cluster, and secure internal communication among NSP components. CA-signed and self-signed certificates are supported.

Note: The NSP supports TLS v1.2 and v1.3.

The NSP provides an internal Public Key Infrastructure (PKI) service to automate the generation and distribution of TLS artifacts within an NSP deployment. The PKI service can generate, sign, and distribute self-signed TLS certificates, or use certificates that you provide.

The NSP TLS artifacts are stored in Kubernetes secrets on each NSP cluster. An NSP tool facilitates secret management. You can use the tool to create or update the required secrets using internally generated or imported certificates, and to back up and restore the secrets.

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 classically managed 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 system elements, with the exception of an NSP 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 an upgrade. Switching to enforcing mode is done only after a deployment operation.

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.

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

6.2 User security and session management

6.2.1 Single sign-on

NSP single sign-on (SSO) provides a common security framework for all supported NSP functions and services. NSP SSO is based on OAUTH2, which is based on the Keycloak open-source identity and access management solution, and uses the standard OAuth 2.0 protocol.

OAUTH2 supports local user management and external authentication agents such as LDAPS, RADIUS, and TACACS+ servers. NFM-P users must be imported to the NSP local user database in order to gain NSP UI access.

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.

When WaveSuite is integrated with NSP in shared mode, the systems share common GUIs, logins, and authentication, with the NSP SSO framework serving as the common authentication system. In non-shared mode integrations, NSP and WaveSuite operate with separate user interfaces and authentication systems, and SSO is not shared between the platforms. See the *NSP Installation and Upgrade Guide* for details on integrating WaveSuite in shared mode and enabling common authentication.

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.

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.

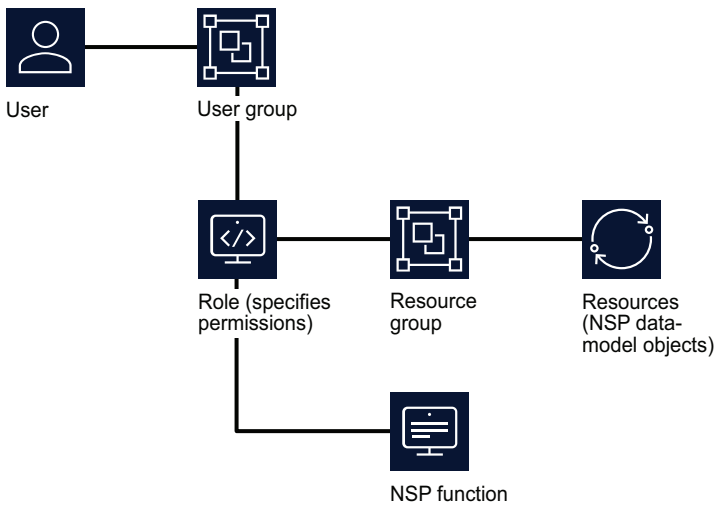
Resource groups

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



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 Classic management 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 components are secured using TLS.

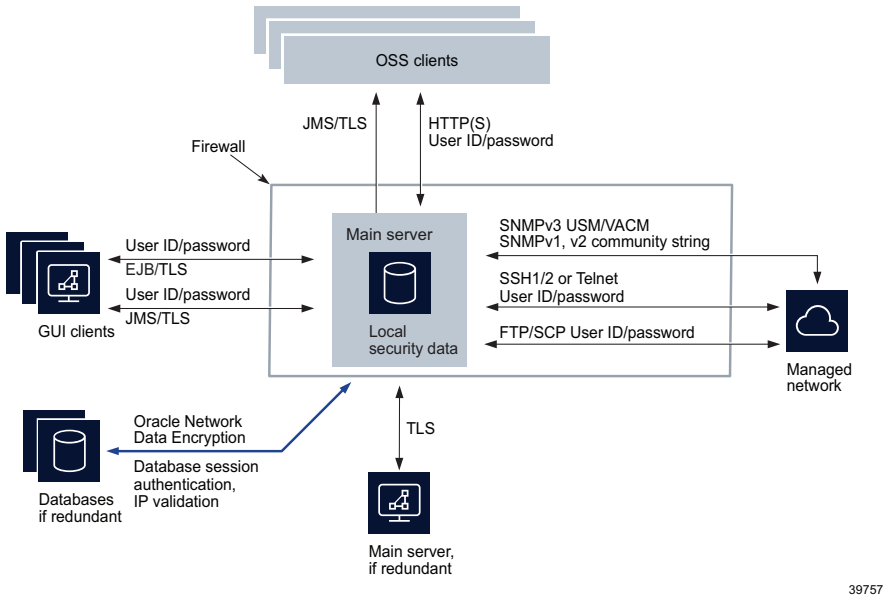
Communication with a main database is secured using Oracle Network Data Encryption.

A GUI, browser, 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.
- 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 and 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.

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.

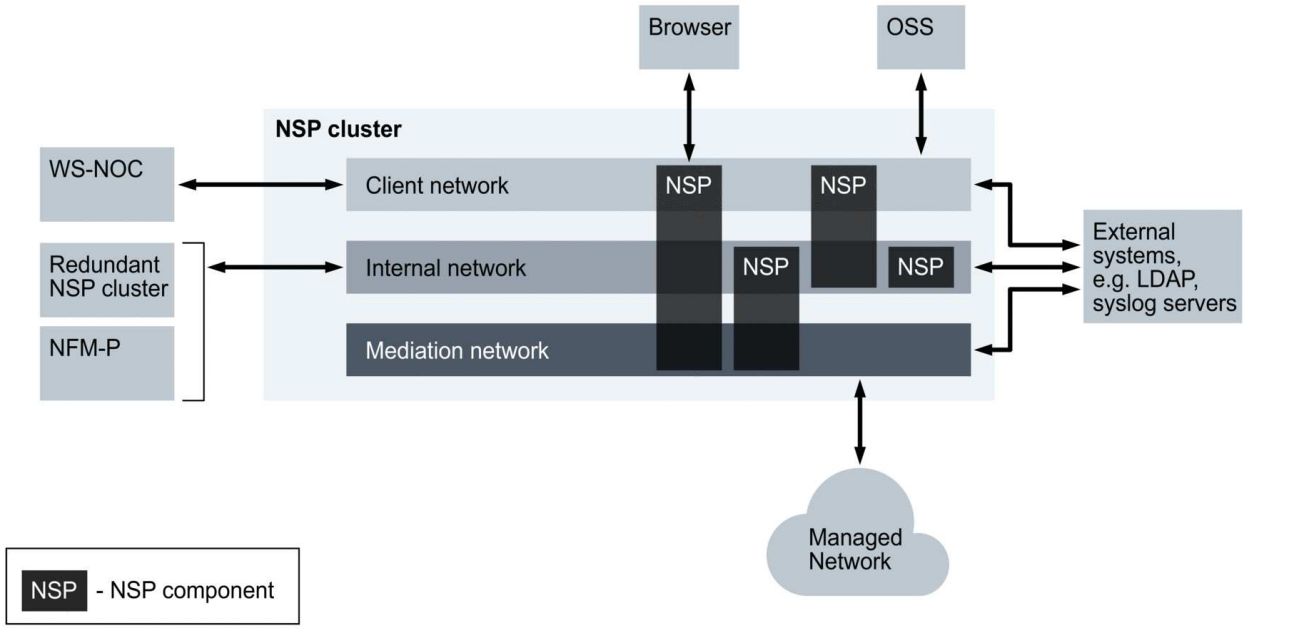
During system deployment, you can enable the use of separate networks by specifying IP addresses for the client, mediation and internal traffic, as required. If you opt to use only one interface per component, only one network is used.

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

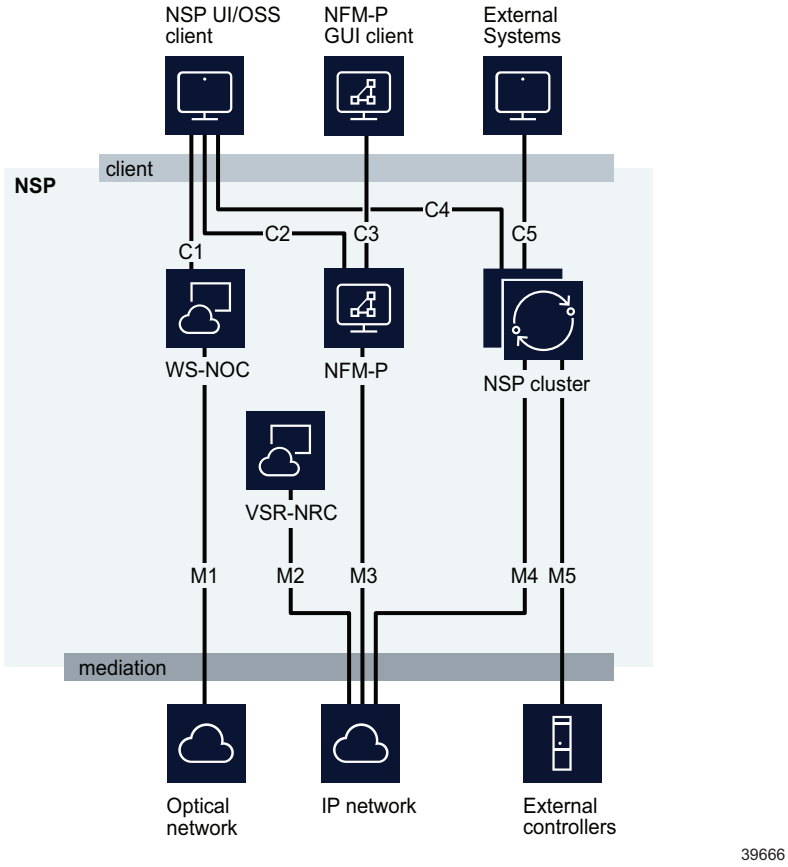


7.2 External NSP communication

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

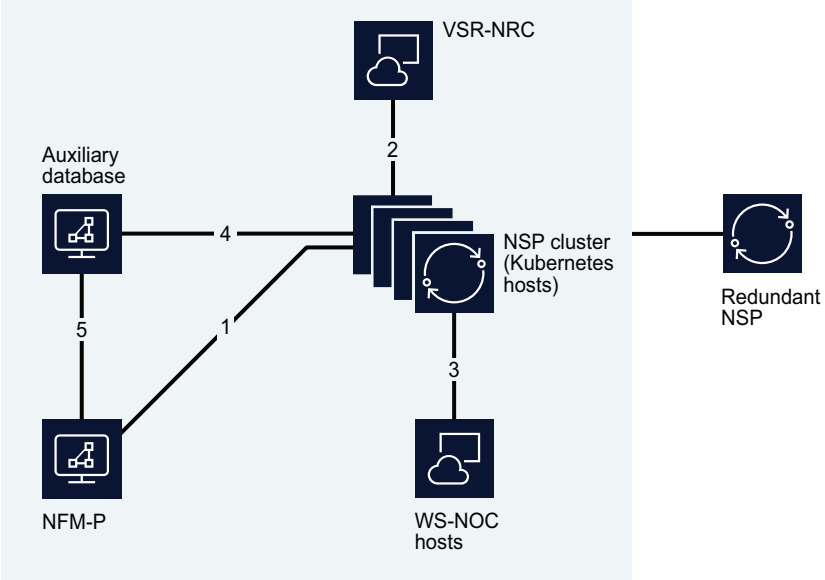
7.3 Internal NSP communication

7.3.1 Introduction

Communication among NSP cluster members and other components in an NSP deployment is performed over the internal network. Each internal interface is secured using TLS. TLS is also applied to the communication among internal NSP system services and processes.

Each NSP cluster member has a unique IP address; however, the cluster members share one virtual IP address for access by other components.

Figure 7-3 Internal NSP interfaces



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 – CPROTO (non-secure)
3	WS-NOC NSP cluster	SSO authentication, Zookeeper registration (non-secure), REST over HTTPS, proprietary WS-NOC HTTP communication
4	NSP cluster NSP auxiliary database	Database transactions, RESTCONF API calls
5	NFM-P NSP auxiliary database	Database transactions

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”.

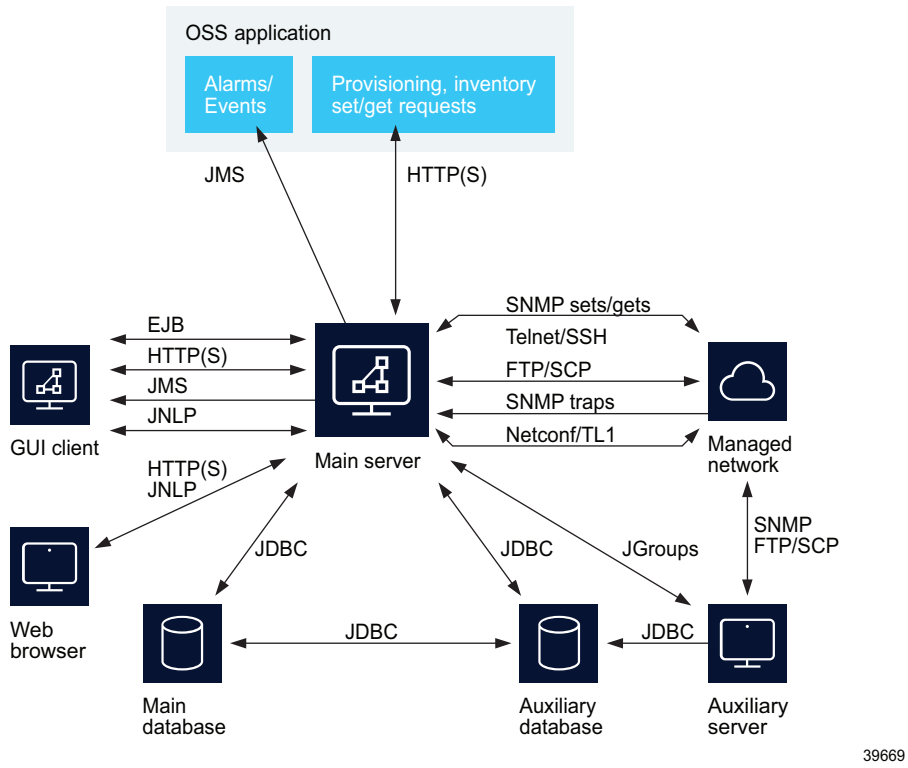
7.4 Classic management communication

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



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 Network Developer Portal 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 Disaster recovery

8.1.1 Disaster recovery

A DR NSP deployment consists of identical primary and standby NSP clusters and ancillary components in separate, geographically distributed data centers, or “sites”. 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: Nokia strongly recommends that all primary components in a DR deployment be 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, determining which cluster will run in 'active' and which in 'standby' mode.

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 base 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

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.

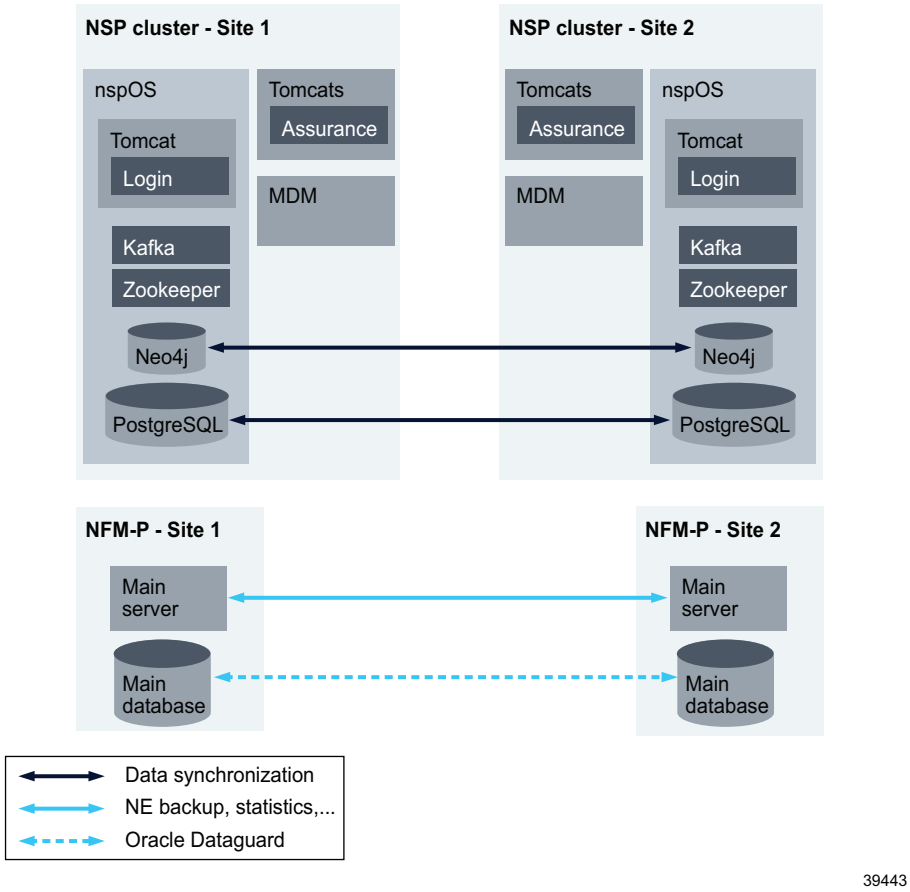
1. An nspOS service at the primary site fails.
2. An activity switch occurs; the standby site consequently assumes the primary role.
3. A major ActivitySwitch alarm is raised against the former primary site, which is now the standby site.

DR for integrated components

A DR NSP deployment can include NFM-P and WS-NOC. NFM-P can be standalone or redundant; however, WS-NOC must be redundant. For example, if a DR deployment includes classic management, the NFM-P can be standalone and WS-NOC is redundant.

The following figure shows a simple NSP DR deployment.

Figure 8-1 NSP DR deployment with integrated NFM-P



8.2 High availability

8.2.1 High availability

The high availability solution is used to provide redundancy within a single site. NSP cluster deployment supports high availability of critical services through replica pods in a container environment. Specific pods are deployed with multiple replicas.

The high availability solution uses a few different approaches to managing a failure.

- Some services are deployed using multiple load-balanced instances where the load handled by any single instance can be picked up by the remaining instances should that single instance fail.
- Some services are deployed in a 3:1 model where there are three instances of the service running and one of them is active. If that active instance fails, then one of the remaining two standby instances will take over immediately.

- For services that are not replicated, additional resources are provided so that a new instance can always be instantiated whenever a failure occurs.

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.

8.3 NSP system failure and recovery scenarios

8.3.1 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.

8.3.2 Primary NSP cluster failure

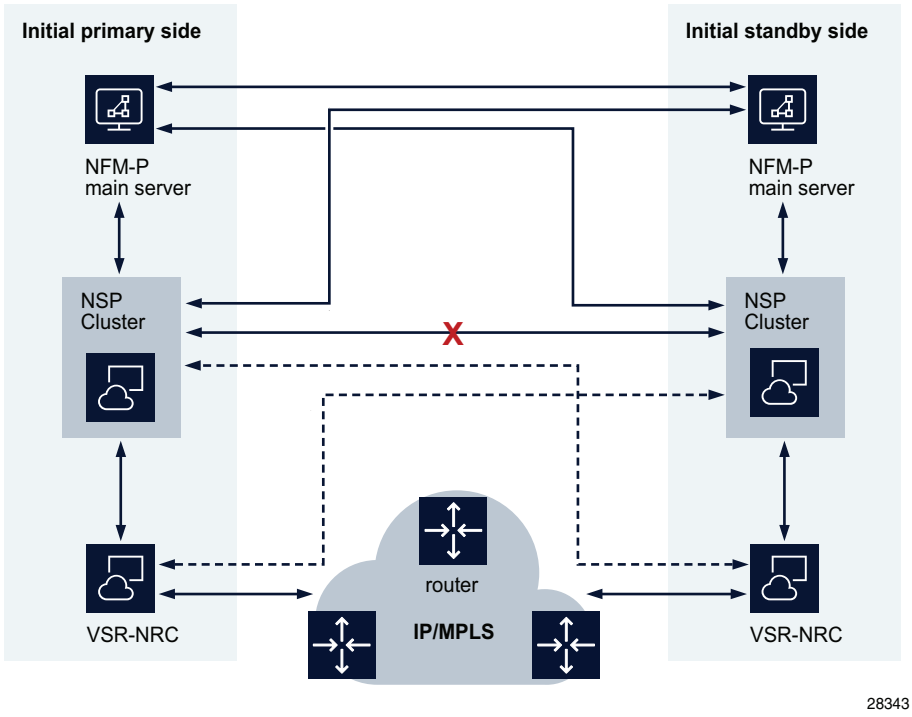
The redundant nsp-role-manager agents exchange a heartbeat every five seconds. If the agent on the standby cluster does not receive a heartbeat within 60 seconds, the standby cluster is promoted to primary. The new primary cluster subsequently communicates with the NFM-P and the newly active VSR-NRC. Primary cluster base services that stop running can also trigger a switchover.

8.3.3 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



8.4 MDM fault tolerance

8.4.1 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.

8.5 Classic management fault tolerance

8.5.1 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,

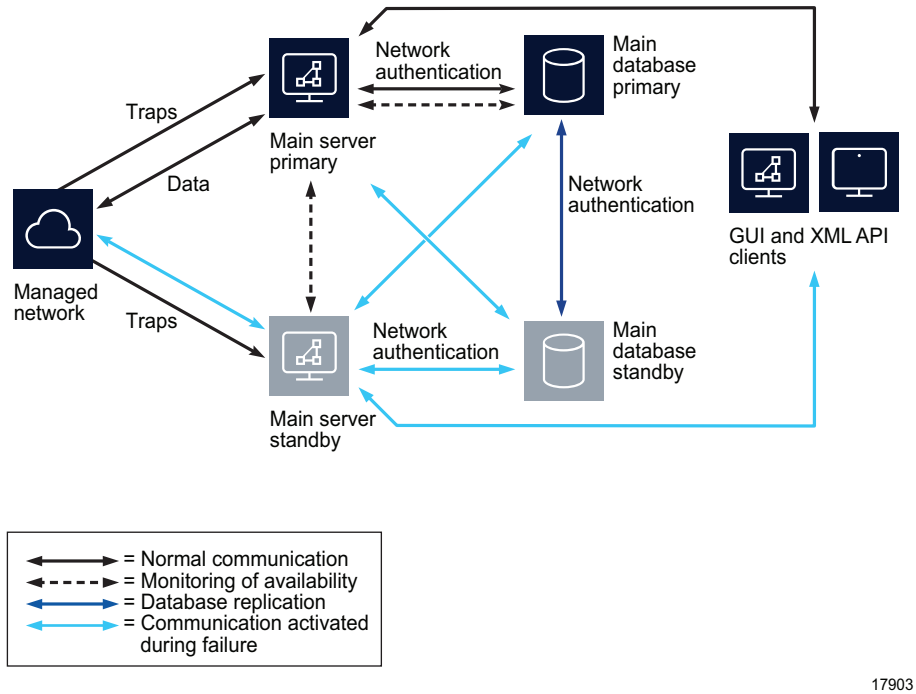
for example, in-band and out-of-band management, can help to prevent the isolation of a main server from the network in the event of a routing failure.

When a communication failure is detected between the primary NSP and the NFM-P main/DB server, JMS server, auxiliary server, or auxiliary database, the *NspBaseServiceDown* alarm is raised against the affected component to provide real-time notification of the fault.

8.5.2 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



Main server redundancy

Main server redundancy is achieved using clustering technology provided by a JBOSS 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.

8.5.3 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.

8.6 VSR-NRC fault tolerance

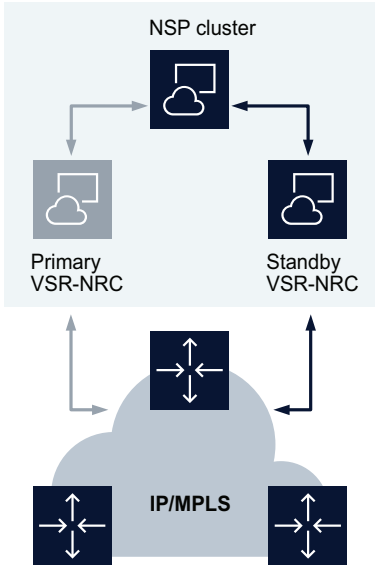
8.6.1 Overview

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.

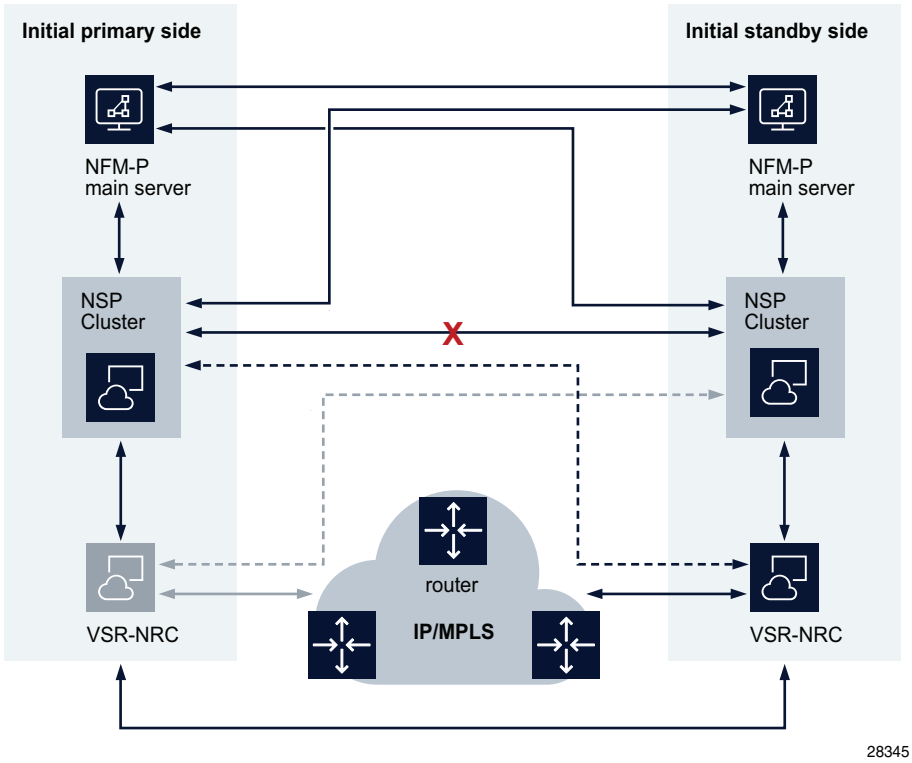
Figure 8-4 Primary VSR-NRC failure in single-site deployment



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



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.

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-idr-bgp-ls-app-specific-attr-16	Application-specific attributes advertisement with BGP link-state
draft-ietf-idr-bgp-ls-flex-algo-06	Flexible algorithm definition advertisement with BGP link-state
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-idr-bgpls-srv6-ext-14	BGP link-state extensions for SRv6
draft-ietf-idr-segment-routing-te-policy-23	Advertising segment routing policies in BGP
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-binding-label-sid-15	Carrying binding label/segment iIdentifier (SID) in PCE-based networks
draft-ietf-pce-pceps-tls13-04	Updates for PCEPS: TLS connection establishment restrictions
draft-ietf-pce-segment-routing-08	PCEP extensions for segment routing

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

Standard or interface	Description
draft-ietf-pce-stateful-pce-14	PCEP extensions for stateful PCE
draft-ietf-teas-yang-te-36	A YANG data model for traffic engineering tunnels and interfaces
Keycloak	Version 26.2.5
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
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 5246	The Transport Layer Security (TLS) Protocol, v1.2
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 7231	HTTP/1.1 Semantics and Content
RFC 7420	PCEP Management Information Base (MIB) model
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

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

Standard or interface	Description
RFC 7952	Defining and Using Meta Data with YANG
RFC 8040	RESTCONF
RFC 8072	YANG patch media type
RFC 8253	PCEPS: Usage of TLS to provide a secure transport for the Path Computation Element Communication Protocol (PCEP)
RFC 8281	PCEP extensions for PCE-initiated LSP setup in a stateful PCE model
RFC 8231	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 8408	Conveying path setup type in PCE Communication Protocol (PCEP) messages
RFC 8446	The Transport Layer Security (TLS) Protocol, v1.3
RFC 8476	Signaling MSD using OSPF (node MSD)
RFC 8491	Signaling MSD using IS-IS (node MSD)
RFC 8525	YANG Library
RFC 8528	YANG schema mount
RFC 8570	IS-IS Traffic Engineering (TE) metric extensions – minimum and maximum unidirectional link delay metric for flex-algo, RSVP, and SR-TE
RFC 8571	BGP - Link State (BGP-LS) advertisement of IGP Traffic Engineering performance metric extensions
RFC 8664	Path Computation Element Communication Protocol (PCEP) extensions for segment routing
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 8919	IS-IS application-specific link attributes
RFC 8920	OSPF application-specific link attributes
RFC 8944	A YANG data model for Layer 2 network topologies

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

Standard or interface	Description
RFC 9086	Border Gateway Protocol - Link State (BGP-LS) extensions for Segment Routing BGP egress peer engineering
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
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
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

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

Standard	Description
RFC 1034	DNS
RFC 1213	SNMPv1
RFC 1738	Uniform Resource Locators (URL)
RFC 2138	RADIUS client 2618
RFC 2328	OSPFv2
RFC 2385	MD5 encryption
RFC 2474	DS Field in IPv4 and IPv6 Headers
RFC 2547bis	BGP/MPLS VPN
RFC 2684	Multiprotocol Encapsulation over ATM
RFC 3107	Labeling in BGP-4
RFC 3411-3415	SNMPv3
RFC 3416	SNMPv2c
RFC 4115	TrTCM
RFC 4379	Detecting MPLS Data Plane Failures
RFC 4761	VPLS
RFC 5246	The Transport Layer Security (TLS) Protocol, v1.2
RFC 5340	OSPFv3
RFC 5925–5926	TCP Authentication Option
RFC 5938	Individual Session Control for TWAMP
RFC 6052	IPv6 Addressing of IPv4/IPv6 Translators
RFC 6241	Network Configuration Protocol (NETCONF)
RFC 6374	Packet Loss and Delay Measurement for MPLS Networks
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.