NOKIA

# Network automation and programmability

## Using the Nokia NSP for intent-based networking

Application note

# Abstract

Are you looking to develop your IP and optical network to be ready for 5G? You will need network automation and programmability to make it happen. With our Nokia Network Services Platform (NSP) you can start evolving your network today and break down the IT/network barriers that may delay your automation journey.

You may want to:

- Transform your operational practices with automated workflows, or
- Get up to speed quickly on programmability with intent-based networking (IBN), or
- Deploy dynamic networks that can respond to changing demand with minimal human intervention.

Our insight-driven networking approach lets you automate your IP and optical networks to make them more agile, reliable and secure. This paper focuses on using the Nokia NSP for IBN in the entire operational life cycle, along with programmable workflow management to enable more fine-grained automation and control. Learn how NSP can help you to develop improved network capabilities that enable IT/network practices to have more flexibility, agility and efficiency for innovation, delivery and operations.

## Contents

## Automation challenges in evolving IT/network practices–and a solution

The networking industry has focused much of the past decade on virtualization and the move toward automation across multiple domains, including technology solutions for resource orchestration, interconnection of clouds and WAN, and big data analytics with software-defined networking (SDN) control. In addition, the adoption of DevOps cultures has improved operational agility, which has been attributed to a shift to model-driven, programmable, IBN. All of this is part of a journey toward the end-to-end deployment of secure, automated, high-performance 5G networks, which will be our next focus for the coming decade. All of these technologies bring the potential to not only improve productivity and accelerate service delivery, but also to reduce operational costs and eliminate manual errors, making networks more profitable and reliable.

Business objectives often seek to continuously adapt organizational cultures, processes and technologies to assist and make the best use of time, effort and resources for operating the network and services. But these goals are often difficult to realize due to resistance to change and slow adoption of new best practices and technology across various groups that depend on each other.

For example, if there were a way to maximize synergies between various IT and network domain groups, then that would alleviate a key set of barriers. There are cultural, technological and operational barriers historically dividing the network and IT domains and which cause difficulties in cross-organizational collaboration when defining APIs and service policies between groups. In many cases this has led to islands of automation that require more expensive and continuing integration development, slowing time to market and service innovation.

## The Nokia Network Services Platform

Our Nokia Network Services Platform (NSP) addresses these modern organizational, process and networking technology challenges with an insight-driven automated networking approach that brings people together by unifying processes and tools. It provides an open, model-driven, multivendor-capable platform that allows programmable APIs to be more easily defined. It allows IT and network domain groups to be able to more effectively collaborate by alleviating issues, such as those caused by technology complexities and knowledge gaps in domain-specific expertise; traditionally these barriers led to time-consuming, effort-intensive interworking.

NSP gives you the flexibility to leverage existing investments in automation, whether from operational procedures, programming and/or CLI scripts while continuing to help you develop workflows and templates within an open, programmable IBN framework.

Nokia is also invested in your DevOps success with NSP by providing support to you and a community of thousands of NSP users through our Network Developer Portal. This portal provides many reference use cases, implementation examples, documentation, SDK tools, and remote cloud labs to sand-box development—all in the name of helping to make NSP incredibly easy to use and program.

## NSP and intent-based networking

IBN is not a new technology, but it has definitely become more relevant within WAN operations and management in recent years. To quote industry analyst Gartner, who started to promote the benefits of IBN in recent years with their Innovation Insight reporting:

> Intent-based networking systems (IBNS) provide a new way to build and operate networks that improve network availability and agility, compared to traditional approaches. IBNS provide life cycle management for network infrastructure, including design, implementation, operation and assurance. IBNS provide mathematical validation that business intent and network configurations are in sync, and can dynamically take real-time action if they are out of sync, which leads to closer alignment between networking infrastructure and business initiatives.
>
> — Source: Gartner[1]

Unfortunately, many network equipment vendor solutions fall short in performing all the required functions of a true IBNS, or must integrate with other external products, microservices platforms and architectures to fill gaps. While incumbent vendors might find it practical to stitch together existing pieces within their software portfolio (or from acquisitions) to deliver an IBNS solution quickly to market, it is important to recognize the TCO, time-to-market agility and reliability advantage of pre-integrated solutions. Bell Labs Consulting has recently released a publicly available report on “Single Integrated Networking Solutions”[2], which studies the value of a single integrated solution (SIS) toward digital transformation, concluding that a SIS provides a superior alternative to today’s multivendor solutions.

As a true IBNS, the Nokia NSP is a SIS that performs all of the key functions that define an IBNS through a unified platform, specifically:

1) Translation and validation

2) Automated implementation

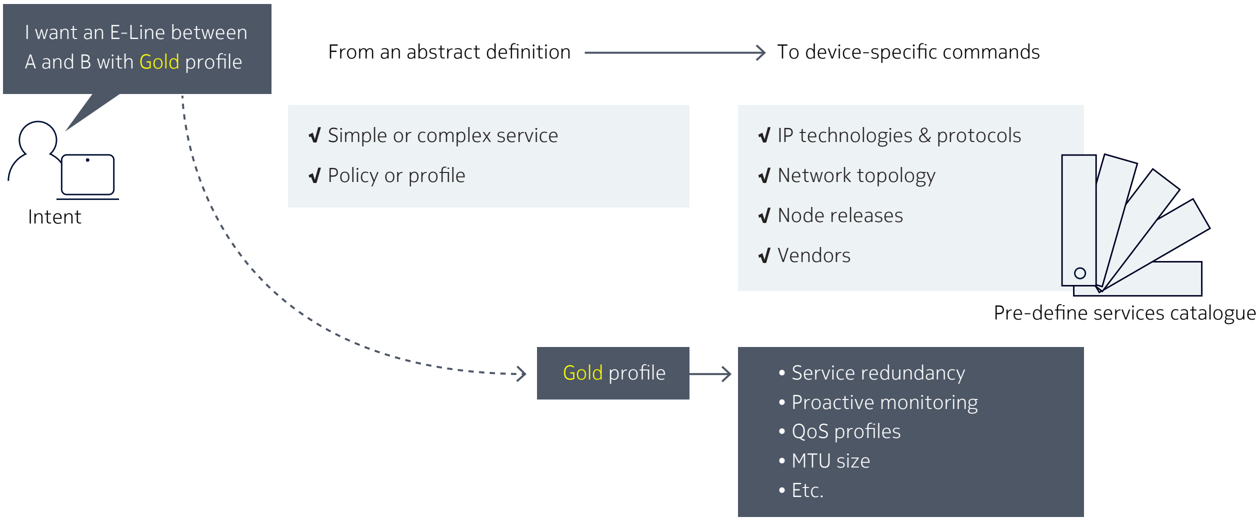
3) Awareness of network state

4) Assurance and dynamic optimization/remediation.

### Mandatory functions of a true IBNS

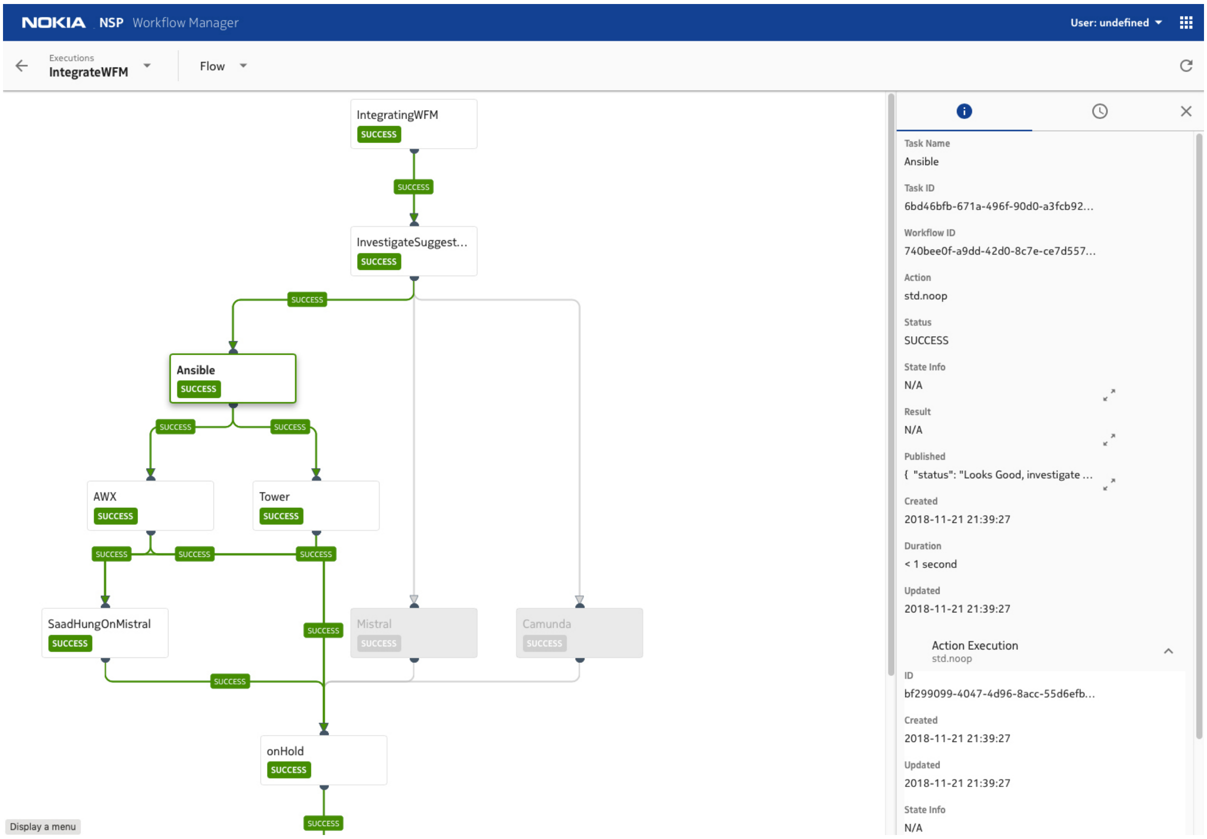
**1) Translation and validation** — The Nokia NSP takes a higher-level business policy that is abstracted and expressed in simple terms (the “what”) as input from end users, and through its northbound API, converts it to the necessary network configuration artifacts (the “how”) for deployment on multivendor networks. Figure 1 shows an illustration of how NSP intent-based service delivery is used to provide an abstract definition and translates this to device-specific commands.

## Figure 1. Example of intent-driven service delivery using NSP


I want an E-Line between A and B with Gold profile
Intent
From an abstract definition
To device-specific commands
✓ Simple or complex service
✓ Policy or profile
✓ IP technologies & protocols
✓ Network topology
✓ Node releases
✓ Vendors
Pre-define services catalogue
Gold profile
• Service redundancy
• Proactive monitoring
• QoS profiles
• MTU size
• Etc.

The NSP's IBN framework with workflow management is central to bringing enhanced programmatic control to extend abstracted vendor models. It is also used as a tool to generate and validate the resulting design and configuration for correctness (including capabilities for using OpenStack Mistral workflow visualization – see Figure 2) before using NSP's model-driven configurator to execute the configuration for deployment in the network.

## Figure 2. OpenStack Mistral workflow visualization within Nokia NSP



Task Name
Ansible

Task ID
6bd46bfb-671a-496f-90d0-a3fcb92...

Workflow ID
740bee0f-a9dd-42d0-8c7e-ce7d557...

Action
std.noop

Status
SUCCESS

State Info
N/A

Result
N/A

Published
{ "status": "Looks Good, investigate ...

Created
2018-11-21 21:39:27

Duration
< 1 second

Updated
2018-11-21 21:39:27

Action Execution
std.noop

ID
bf299099-4047-4d96-8acc-55d6efb...

Created
2018-11-21 21:39:27

Updated
2018-11-21 21:39:27

State Info
N/A

NSP’s intent-based template catalog simplifies and brings consistency to translation of configuration by allowing differences to be compared; this also ensures that business logic matches the operator’s intent. The intent-based template catalog has custom profiles that are typically developed and maintained by network domain experts in conjunction with YANG-defined intent models. The catalog enables IT and operational support systems to have abstracted selection options and a simplified development interface so that focus can be put on the business and operational logic that really matter for the specific use case, not the network implementation details. Intent-based profiles hide the network complexity details that matter mainly only to network operators and engineers, such as service redundancy requirements, proactive monitoring, QoS profiles and other service attributes like MTU size (as shown previously in the gold profile in Figure 1).

Programmable workflows continue to be important for achieving the most flexible and fine-grained control when developing network automation. In addition, the abstraction benefits of NSP’s intent-based templates further simplify development work and bring consistency of implementation by enabling the use of “golden configurations” rather than repeating similar implementation patterns in many different places. This helps to alleviate the chance of human error or misalignments. Designing workflows with templates and “golden configs” brings uniformity to implementation for proper deployment and enables auditing for validation as well. In addition, programmable workflows form a critical linkage to enable assurance for meeting full operational life-cycle requirements—a key requirement of a true IBNS. For example, once the network is configured and services provisioned (along with validation functions that detect misalignments from intent), intent-based workflows may also execute performance validation to ensure SLAs are successfully met. Closed-loop assurance workflows would be automatically invoked to address any issues or would flag the issue to operations staff to take suggested actions. (This closed-loop assurance will be discussed later in the “**Assurance and dynamic optimization/remediation**” section.)

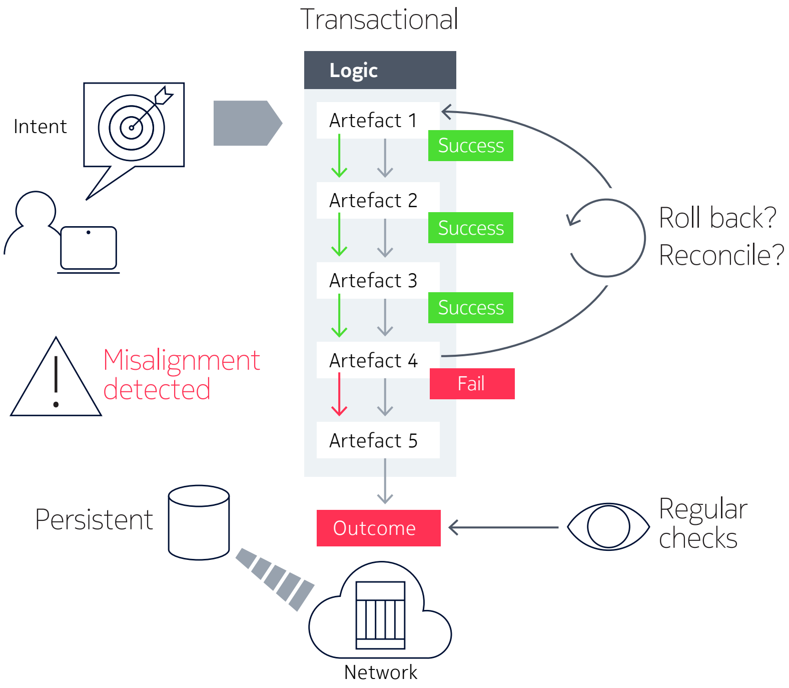
**2) Automated implementation** — NSP uses network automation to configure the appropriate network changes (the “how”) across existing network infrastructure. Templated parameters are derived from a YANG-defined intent model to provide automated implementation.

For example, commands are automatically populated into templates using the NSP intent-model and profiles in combination with minimal inputs selected and catered to the specific operational environment (for example, to specify identifying parameters such as: Service Name/ID, RD/RT values, Service Tunnels, VC-IDs). There is also the ability to design business intents that translate to network requirements (for example, objectives to fulfill specific optimization criteria such as minimum bandwidth, maximum latency requirement, or to provide other network features for greater resiliency, maximum security or improved quality of experience).

> Nokia is increasing the configurability of NSP with its upcoming “NSP templating engine”.... This will allow users to define intent-based service models (YANG), logic to execute them, and facilities to trigger them from, and influence their behavior with external data. In the end this means that NSP users will not be limited to built-in capabilities, but can create services and automation, on their own, throughout the lifetime of the solution. Critically, this is intent-based automation which, Appledore argues, greatly simplifies closed-loop actions.
>
> — Source: Appledore Research[3]

Execution of individual command sets is transactional. This is fundamental because it enables rollback to the initial state/config if a step fails during the process. Rollback, however, is not mandatory. Flexibility is required where it is better to "continue on error" rather than roll back on a large sequence of transactions, in which case it will bring the intent into a "misaligned" state (as shown in Figure 3).

Figure 3. Example of transactional automated implementation using NSP



At this point the decision can be made to delete the intent, adjust the intent definition or to try it again (reconcile). The intent reflects the operational or engineering desire, so in this case the desire will not change just because a command could not be deployed.

For example, it could be just a temporary issue (nodal connectivity) that caused the failure at the moment of execution. This automated implementation behavior is especially a requirement for large service topologies, like when deploying an infrastructure service to thousands of sites where there may always be some sites that are temporarily unavailable. Strict enforcement using "rollback-on-error" for such services would be highly inefficient.

In addition to network configuration, NSP provides programmable automation across various use cases, including:

- Rapid network commissioning and maintenance
- Zero-touch provisioning and delivery of elastic bandwidth services
- Network management and assurance for maximum performance and reliability
- SDN control to optimize network utilization, latency and traffic engineering.

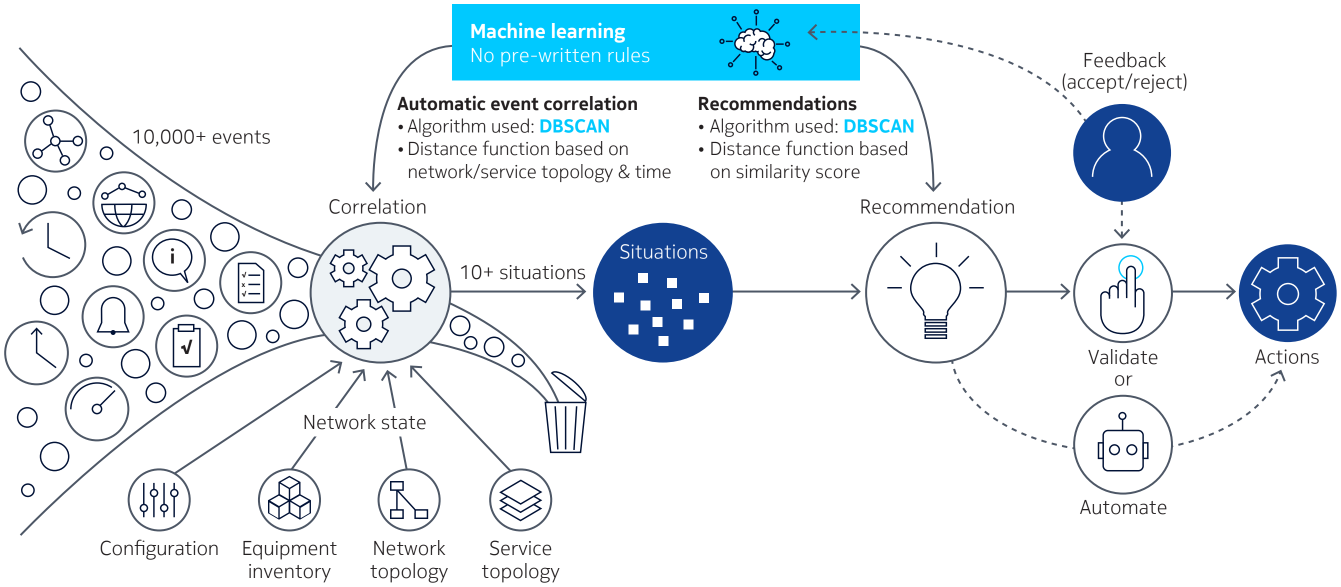
**3) Awareness of network state** — NSP ingests real-time telemetry and network status for multivendor network domains and resources under its control and management. It is protocol- and transport-agnostic, working across multiple network technologies and layers, including physical/virtual infrastructure.

Visibility is key for improving automation and control. Nokia believes that automation should not be a black box, so in order to build trust in automated processes, NSP makes them transparent. Within NSP, feedback and dashboards are provided during automation so that network operators and engineers can see what is happening at all times. NSP carefully monitors automated processes in the network, highlights key correlations and provides effective ways to visualize them. NSP also allows workflows to be built to reserve key decisions for human operators to provide inputs (i.e., check and validate actions before proceeding).

Post-execution, NSP also tracks network state persistently so that it can regularly check the outcome, detect a misalignment, and reconcile/resync (or self-adjust) to maintain the intended network state (as shown previously in Figure 3).

The Nokia NSP continuously monitors network state across many dimensions of configuration, equipment inventory, network and service topologies, routing updates, performance statistics, and SLA test diagnostics. It collects and correlates tens of thousands of events with network state using our machine learning algorithm to significantly reduce the overwhelming data pool down to a manageable number of incidents (as shown in Figure 4, which illustrates awareness of network state being correlated with events to apply the NSP machine learning algorithm toward automation).

Figure 4. Network state and event correlation used by machine learning in NSP



The machine learning algorithm can then make recommendations on how to fix the situation based on similar situations that happened in the past. At any point in time, the operator can check the recommendation, as well as validate the recommendation. Then the actions needed to fix the situation can be taken using NSP. Once the operator has proven out the action, the decision can be made to let NSP perform the changes automatically. Nokia believes that this approach of enabling a self-paced evolution toward automation is key to being able to increase adoption. Confidence in automation tools must be built through complete visibility with control for operators and engineers to step in when desired.

> Automation depends on intelligence and context. NSP has adopted machine learning to support improved correlation analysis and issue resolution. NSP manages the collection of parameters, incidents, and trends and their analysis. It also correlates this with service-impacting events and conditions in order to identify meaningful fault scenarios.
>
> — Source: Appledore Research[3]

**4) Assurance and dynamic optimization/remediation** — NSP continuously validates (in real time) that the original business intent is being met, and can take corrective actions (such as blocking or steering traffic, modifying network capacity or optimizing network paths, as well as notifying for human intervention) when desired intent is not met.

To make automation work well in live network deployments, it is critical that an IBNS needs to support the full operations life cycle that includes assurance as well as analytics to trigger automated actions for dynamic optimization and remediation. NSP performs this function as a single, unified platform for IBN and automation. It allows operations to keep pace by driving and automating smarter services placement on network resources that meet SLAs and improve end-user experience. It brings closed-loop optimization and remediation to deliver maximum reliability and performance.

> **Closed loop automation and the new role of assurance**
>
> Automation, the “merger” of fulfillment and assurance, and the emerging micro-services-based “dev-ops” methods all imply a new, and in many ways expanded, role for evolved assurance, and also for analytics. Closed loops, and the automation they effect, rely on timely, accurate, insightful intelligence that identifies problems, correlates them with technical causes, and drives the original process to make automatic modifications. Assurance and analytics combine to become complementary components of that intelligence platform.
>
> — Source: Appledore Research[4]

Assurance and analytics feed NSP as an SDN controller with the KPIs needed to deliver intelligent steering and load-balancing of traffic. For this use case, NSP functions as an SDN standards-based Path Computation Engine (PCE) for IP/MPLS and segment routed networks, and separately provides a hierarchical multi-layer SDN module for cross-domain control and coordination in IP/optical networks.

Telemetry on performance, latency, link utilization and traffic flow statistics is also collected by NSP to monitor for congestion. KPIs enable NSP’s analytics-driven policies to automate actions that ensure critical SLAs are met and optimal use is made of IP/optical assets. For example, to avoid network congestion that causes latency and performance degradation, actions could be dictated by intent-based policy to have traffic flows redirected, new multi-layer paths established or existing paths resized dynamically.

In addition, machine learning algorithms (as shown previously in Figure 4) provide significantly great potential for IBN automation. Early customer deployments of the service and network assurance with the machine learning algorithm in NSP have seen results of 99 percent alarm reductions that have improved customer satisfaction due to shorter repair times and immediate reliable customer information. Augmented assurance visibility also enables the tightening of SLAs both internally and through negotiations with integrating network providers.

> Nokia's NSP solution is able to deliver a complete solution for the complete network service life cycle, including network assurance. It enables closed loop automation of network services on a common platform, without the need for system (and data) integration of multiple systems. Nokia believes that this is now enabling it to make significant wins against its traditional competition.
>
> — Source: Appledore Research[3]

# Abbreviations

| | |
|---|---|
| API | application programming interface |
| CLI | command line interface |
| IBN | intent-based networking |
| IBNS | intent-based networking system |
| KPI | key performance indicator |
| MDM | model-driven mediation |
| MPLS | Multiprotocol Label Switching |
| NSP | Network Services Platform |
| PCE | Path Computation Engine |
| QoS | quality of service |
| SDN | software-defined networking |
| SIS | single integrated solution |
| SLA | service level agreement |
| TCO | total cost of ownership |

# References

1. Gartner: Andrew Lerner, Joe Skorupa, Sanjit Sanguli, "Innovation Insight: Intent-Based Networking Systems"; Published: 07 February 2017; Refreshed:13 April 2018.
2. Bell Labs Consulting: Narayan Raman et al, "Single Integrated network solutions - The TCO, agility and reliability advantage of pre-integrated solutions", February 2019.
3. Appledore Research: Grant Lenahan, "Solution Profile – Nokia Network Services Platform (NSP)"; August 19, 2019.
4. Appledore Research: Grant Lenahan, "Closed Loop Automation and the New Role of Assurance"; September 1, 2016.

# Related resources

- Application note: "Programmability for network automation: Supporting modern workflows while evolving to an intent-based networking system and DevOps"
- Demo: Service and network assurance with machine learning
- Data sheet: Network Services Platform

# Appendix: Intent-based management – NSP product details

## Intent models and data store

Network/service engineering teams define NSP intent models with the purpose of abstraction so that the only attributes exposed are those that operators (or operations support systems) care about. Typically, IT/service specifics and behavior are modeled to expose upwards to northbound systems, while attributes needed purely for the sake of the service implementation are able to be determined automatically by custom business logic bound to the intent model.

The user interface (NSP web UI) and NSP northbound APIs (RESTCONF and NETCONF) are also automatically derived from intent models. NSP implements its own configuration data store for intents. This data store is considered the "source of truth" (master). It is possible to audit intents, to determine if the actual network configuration is aligned or misaligned with the intent. In the case of a misalignment, intents can be synchronized back to the network to reconcile the configuration.

A service script (developed using JavaScript) defines the business logic on how to use the intent data instance to create, update, delete, audit and reconcile the corresponding configuration. The service script makes use of templates (using the Apache FreeMarker Java-based template engine) and other resource files (e.g., JavaScript, JSON) to improve the readability and maintainability of intent definitions.

To keep intent definitions simple, it is preferred that NSP communicates to a model-driven configuration interface, which is fully transactional and declarative. NSP is designed to use its model-driven mediation (MDM) framework to support configuration through its RESTCONF API. It is also open and programmable for third-party integration.

## Model-driven mediation

The NSP MDM provides an agile, DevOps-ready multivendor framework for supporting new equipment releases and service models at just-in-time speed. With an MDM framework, device upgrades can be decoupled from traditionally lengthy NMS/controller upgrades. Forward-compatibility for supporting new devices and service models can be inherently provided by NSP with MDM as well—without requiring a platform upgrade.

This new paradigm shift to model-driven management is fundamentally different from the present mode of operations and delivers a dramatic improvement over the current process. For example, in the past, operators may have had to wait many months for some new equipment releases to be supported because equipment feature support and the necessary device and service object models needed to be changed in the management system code base (by vendor software designers in the case of a vendor-supplied system, or by in-house developers in the case of home-grown systems). In many cases there were also further delays that resulted from waiting for the new release to be made available in the next upcoming vendor release cycle.

In addition, the deployment timeline for vendor software needs to be planned, implemented and tested for NMS/controller platform-wide upgrades, which adds many more months to go live—especially when OSS integrations also need to be re-validated. With MDM, NSP significantly reduces these deployment delays, sometimes lasting months, to a minimum—as little as hours or days in many cases, depending on the project scope.

With MDM, new device features can efficiently be exposed to northbound systems by adopting new southbound and northbound models and by creating new adaptation scripts to translate between the two. There is no longer a need to change internal models.

The maximum level of automation is enabled by leveraging YANG modeling, which has become predominant in modern IP networks. With the YANG model being hot-deployed, NSP support can be ready to use as soon as the YANG model is made available and deployed using the MDM. This is possible because the object models and NMS/controller support are automatically derived from the YANG model. Support should also be provided out-of-the-box for many standardized YANG models, such as IETF L2 and L3 service models. In addition, GUIs (such as for model-driven configuration) and RESTCONF northbound APIs can be auto-generated from YANG models.

As part of its model-driven network management, the NSP provides a Workflow Manager for programmable network and service automation.

### Intent Manager

The NSP Intent Manager is designed to deal with updates of the intent definition. The operator can choose to either automatically reconcile all network services based on the newer definition or to clone the definition before changing; this gives operators full control to migrate services as required. To avoid large network impact, in case of automatic reconciliation the service migration follows operator-defined policies (e.g., to limit them to maintenance windows, or limit the number of concurrent migrations).

Example service configuration life-cycle use cases include: Flexible service management, enabling:

- Upcoming, innovative services
- Composition of multi-technology services
- Hierarchical services
- Multi-segment/multi-domain services
- Programmatic support for standard service models
- Policy/profile management
- Golden configuration (e.g., network equipment templates).

Dynamic assurance and optimization use cases are made possible by an abstracted intent model that is enhanced when attributes are added for:

- Network state and statistics (read-only)
- Network performance and SLA diagnostics, ( e.g., populated by triggering on-demand OAM measurement)
- Custom service-related alarms and notifications
- Embedded resource management, to help the system to automatically select appropriate keys, addresses and/or identifiers following custom-defined building rules, ranges or from pools
- Enhanced operational tools like preview, dry-run validation and service history
- Network service discovery using reverse mapping rules
- Catalogue of lookup/helper functions to add intelligence
- Service re-optimization
- Service assurance: health/alarms with correlation, topology views, statistics, measures
- Service activation tests and birth certificates.

## About Nokia

We create the technology to connect the world. Powered by the research and innovation of Nokia Bell Labs, we serve communications service providers, governments, large enterprises and consumers, with the industry's most complete, end-to-end portfolio of products, services and licensing.

From the enabling infrastructure for 5G and the Internet of Things, to emerging applications in digital health, we are shaping the future of technology to transform the human experience. networks.nokia.com

Nokia Oyj
Karakaari 7
FI-02610 Espoo, Finland
Tel. +358 (0) 10 44 88 000

Document code: SR1909038523EN (October) CID206794