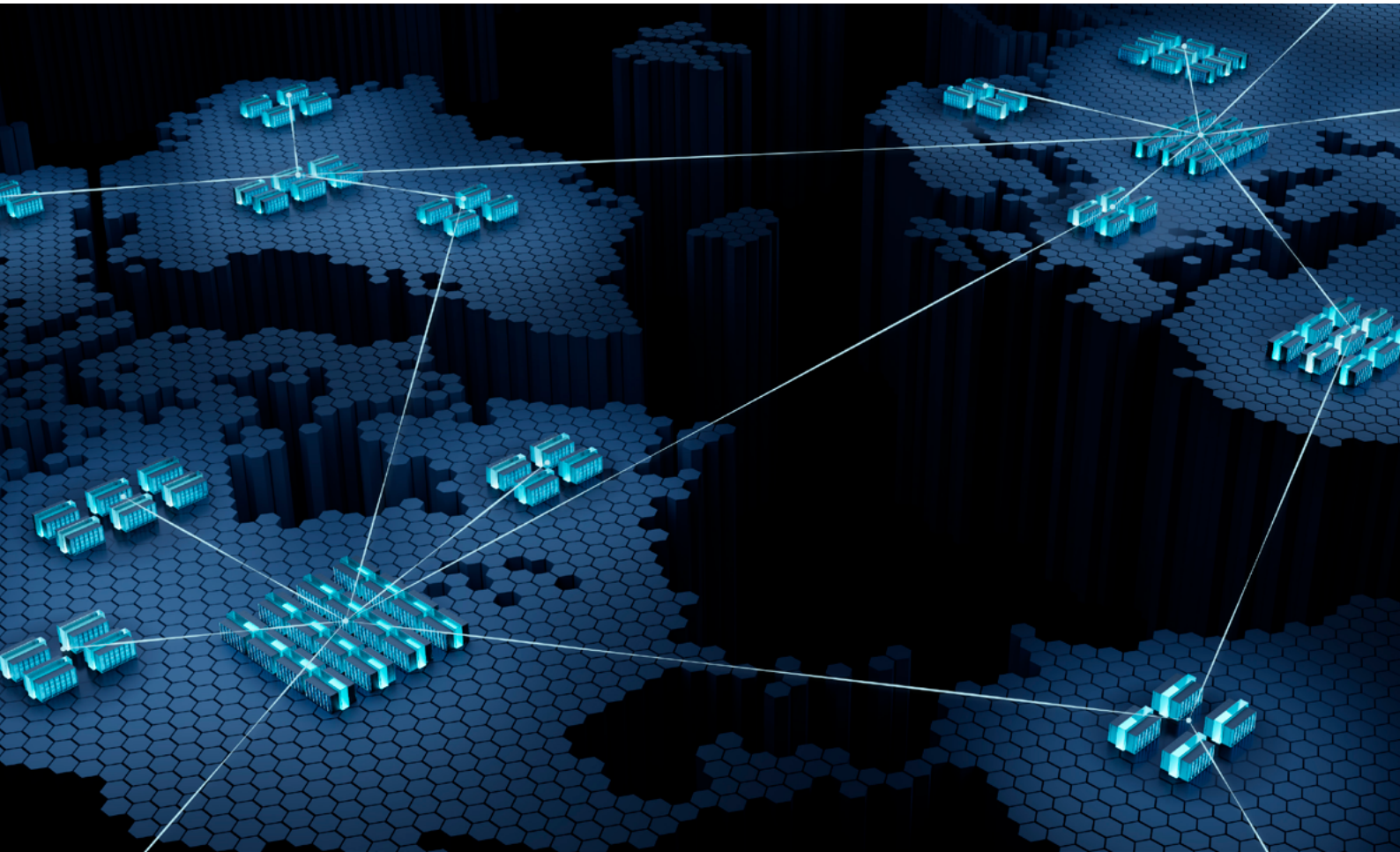


NOKIA



Nokia Service Router Linux

An open, extensible and resilient network operating system

Product description

Abstract

The challenge facing data center network operators is to balance the need to constantly scale their networks and operations against the increasing costs of designing, building and operating these critical networks. Evolving application needs and the increasing adoption of distributed data centers is driving the need to rethink how data center networks are designed and operated.

Nokia Service Router Linux (SR Linux) is an open, extensible and resilient network operating system (NOS) that delivers scalability, flexibility and efficiency in data center and cloud environments.

This document describes Nokia SR Linux and explains how its unique modular, extensible architecture helps to simplify the design and operation of data center fabrics.

Contents

Abstract	2
Introduction	4
Nokia SR Linux overview	4
Nokia SR Linux open and modular infrastructure	6
Linux-based NOS	6
Field-proven protocol stacks	6
Modular, state-sharing and extensible architecture	6
Third-party application support	7
Hardware extensibility	8
Nokia SR Linux management overview	8
Ground-up, model-driven management infrastructure	8
Scalable streaming telemetry	9
Software extensibility	9
Superior CLI programmability	10
Enhanced NDK	10
Nokia Fabric Services Platform automation and operations toolkit	11
Nokia Data Center routers	11
Summary	11
Learn more	12
Abbreviations	12

Introduction

The challenge facing data center network operators is to balance the need to constantly scale their networks and operations against the increasing costs of designing, building and operating these critical networks. Evolving application needs and the increasing adoption of distributed data centers is driving the need to rethink how data center networks are implemented.

As a key component of the Nokia Data Center Fabric solution, the Nokia Service Router Linux (SR Linux) network operating system (NOS) delivers scalability, flexibility and resiliency within the data center and across hybrid and multi-cloud environments.

The Nokia Data Center Fabric solution is a next-generation solution that includes the following products in addition to Nokia SR Linux:

- Nokia Fabric Services Platform: A declarative, intent-based automation and operations toolkit that delivers agile and scalable network operations for data center and cloud environments.
- Nokia Data Center routers: A portfolio of routers that deliver massive scalability, openness, aggregation and interconnection for data center and cloud environments.

The Nokia Data Center Fabric solution leverages Nokia's expertise in IP routing and network operations, the Nokia Service Router Operating System (SR OS), our IP network automation solutions, and our proven track record of building business-critical Ethernet and IP/MPLS networks for service providers, webscale companies and enterprises globally.

This document describes SR Linux. For details about the Data Center Fabric solution, the Fabric Services Platform and our data center routers, see the Learn more section at the end of this document.

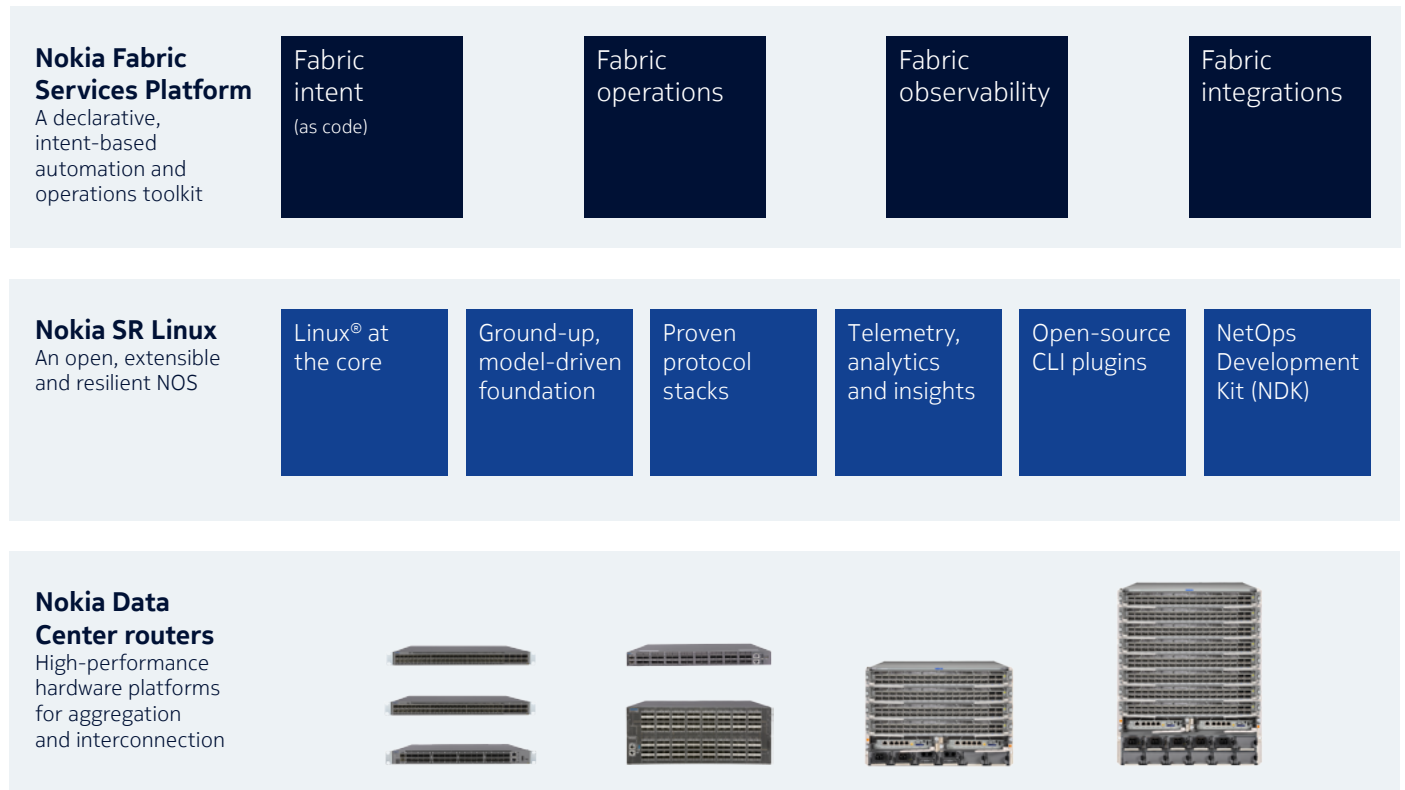
Nokia SR Linux overview

Nokia SR Linux was developed in close collaboration with some of the world's largest data center operators. It was designed to solve real-world challenges in on-premises and cloud environments where the primary challenges are scalability, ease of operations, or both.

A common approach to this problem is to simplify the data center network design, components and protocols. While this allows additional scalability, it reduces flexibility in a world of constantly evolving network requirements.

The overall network must still provide high levels of programmability, openness, reliability and extensibility to meet growing DevOps and agility requirements. These diverging requirements cannot be met by current industry solutions, and they were the genesis for the development of SR Linux and the Nokia Data Center Fabric solution (see Figure 1). The Nokia Data Center Fabric solution provides openness, flexibility, robustness and automation to make data center and cloud environments easier to scale, adapt and operate.

Figure 1. Nokia SR Linux and the Nokia Data Center Fabric solution



SR Linux delivers:

- An open, extensible and resilient NOS that is fully programmable and massively scalable
- A unique, model-driven architecture designed from the ground up for simplified operations and integrations, and ultimate visibility
- Field-hardened protocol stacks for enhanced IP routing features including, MP-BGP, EVPN and VXLAN
- Extensive streaming telemetry, designed from the core out to support unprecedented granularity and volume
- Unrivalled support for integrating community and customer-driven applications into the core of the system
- Plug-and-play hardware integration
- A customizable, open-source command line interface (CLI) and on-demand, customized CLIs leveraging the flexibility of Python.

Nokia SR Linux open and modular infrastructure

Nokia SR Linux provides an open Linux[®]-based NOS with an innovative architecture foundation, field-hardened protocol stacks and model-driven management. It enables an open, modular and extensible data center networking infrastructure with support for third-party network applications.

Linux-based NOS

In the past, NOSs were closed, proprietary systems. Openness is best achieved with an underlying open operating system such as Linux. Nokia SR Linux is a Linux-based NOS that builds on the Linux kernel to provide a set of loosely coupled services that come together to provide the functional blocks of a NOS, along with simple interfaces for their consumption.

Nokia SR Linux provides a set of modular applications that are isolated into their own failure domains. A central application manager is responsible for the life cycle of each application and provides full control of the protocols running on the system. As a result, each operator needs to run only the applications required for a given environment; this simplifies administration and improves security.

Unlike some other Linux-based NOS in the industry, SR Linux uses an unmodified Linux kernel as the foundation on which to build a suite of network applications. This provides many benefits, including reliability, portability and ease of application development. Using an unmodified kernel also speeds the availability of non-Nokia applications (for example, OpenSSH) and security patches for operating system components.

Field-proven protocol stacks

Nokia designed SR Linux for the DevOps era using the field-proven protocol stacks from the Nokia SR OS. The Nokia SR OS has a strong pedigree in IP routing developed over the past 20 years. Nokia customers have deployed more than a million IP/MPLS routers in over 1,300 networks, including the internet backbone and some of the largest service provider networks in the world.

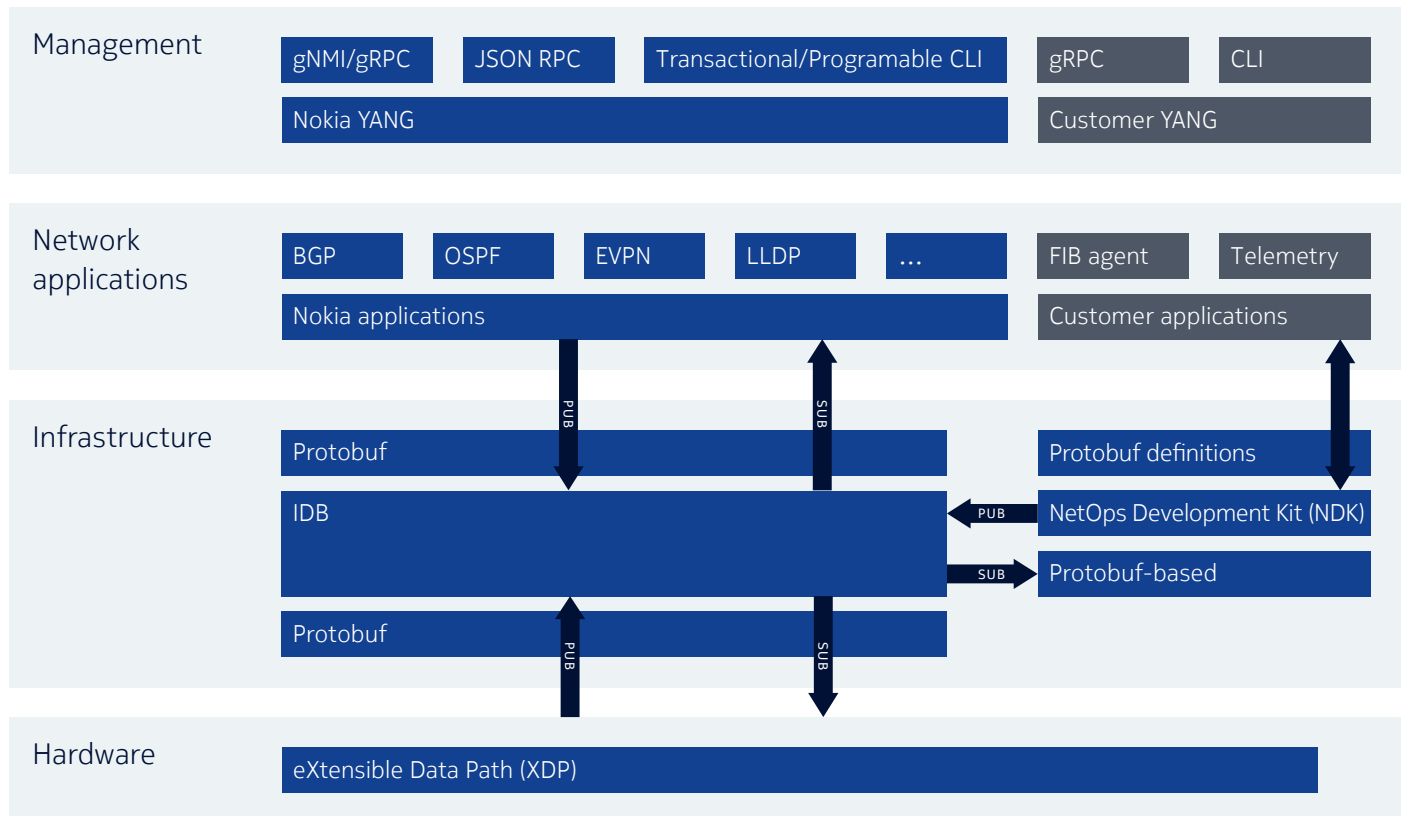
Webscale, service provider and enterprise data centers are increasingly adopting leaf-spine fabrics using enhanced IP routing with MP-BGP, EVPN and VXLAN. By using these field-proven protocol stacks, customers immediately benefit from the resiliency, stability, scalability and proven interoperability of the SR OS.

Modular, state-sharing and extensible architecture

Nokia SR Linux applications share state with each other through a publish/subscribe (pub/sub) architecture (see Figure 2). The Nokia pub/sub architecture uses two recent opensource innovations: protocol buffers (protobufs) and generalized Remote Procedure Call (gRPC).

Previous-generation NOSs were forced to implement proprietary and non-performant inter-process-communication (IPC), defeating any hope of extensibility and leading to potential resource starvation and inconsistencies under convergence scenarios. Protobufs and gRPC were chosen because they are both extremely fast, efficient and scalable, which are key attributes for any NOS. In addition, external applications can be supported with no extra effort required.

Figure 2. Nokia SR Linux architecture provides an open infrastructure



SR Linux has a state-sharing architecture that relies on the Nokia Impart Database (IDB), a lightweight database developed by Nokia specifically to meet the performance demands of network applications. The Nokia IDB runs in memory and is optimized for handling high volumes of messages while protecting against any one application slowing down the whole system.

The IDB acts as a highly scalable key-value store, where both the keys and values are serialized protobufs, and IDB is completely agnostic to the data being published. This promotes simplicity and scalability in the architecture and provides extensibility at the heart of the system.

In addition, the IDB uses backpressure mechanisms and intelligent message coalescing to ensure that neither a chatty publisher nor a slow subscriber can negatively impact any other applications. Message coalescing allows subscribers that are under load to process updates when they can, ensuring that the intent of the operator will be met, even when network convergence events drive contention for resources.

Unlike other pub/sub architectures that act as simple message queues, the IDB provides a reliable and scalable delivery mechanism, guaranteeing delivery of updates to subscribers. gRPC provides an efficient, secure communication channel for applications and allows native extensions through third-party applications.

Third-party application support

Nokia SR Linux allows third-party applications to be fully integrated into the system with the same functionality as Nokia applications. This includes consistent configuration using YANG, telemetry support, life cycle management (LCM) and visibility of system resources.

In addition, the SR Linux NetOps Development Kit (NDK) provides full access to the IDB, CLI and APIs. The NDK uses gRPC and protobufs to provide maximum flexibility for languages supported and backwards compatibility. This approach differs from others, which are restricted to certain languages, versions and/or libraries.

The other approaches also require third-party applications to be managed independently, which creates additional operational overhead.

Hardware extensibility

Hardware flexibility is an essential requirement for large-scale data center operators. Nokia SR Linux supports a variety of data center networking chipsets through the Nokia eXtensible Data Path (XDP).

The Nokia XDP serves as a hardware abstraction layer that speeds time-to-market for new or different data center networking chipsets. The XDP provides a common set of software instructions that northbound applications use so they are not directly dependent on ASIC-vendor SDKs. The XDP borrows from the Nokia development experience for high-performance virtual network functions and makes use of user space acceleration for traffic destined to the control plane and any non-ASIC interfaces.

Nokia SR Linux management overview

The Nokia SR Linux native model-driven design and infrastructure delivers superior software extensibility. Data center operators have the flexibility to implement their own management and orchestration solutions or implement a toolkit for data center fabric management using the Nokia Fabric Services Platform.

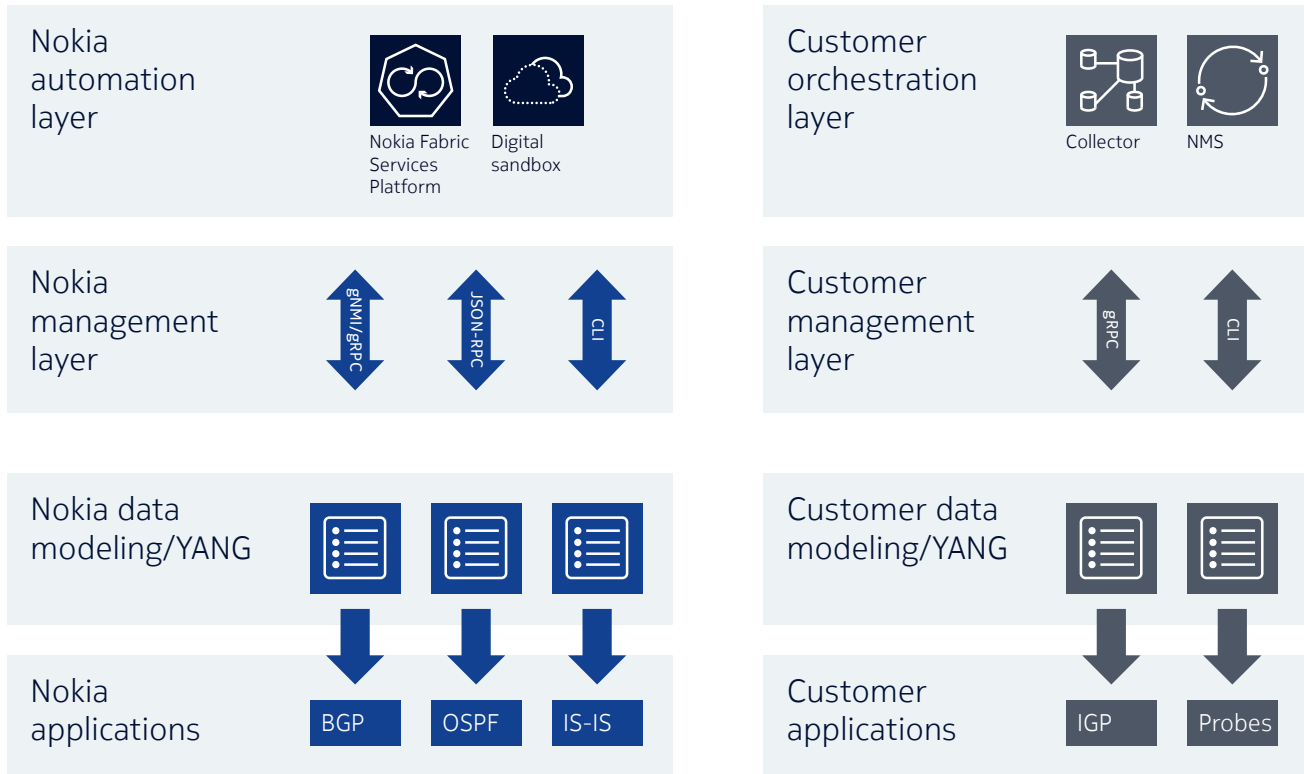
Ground-up, model-driven management infrastructure

Nokia believes a strong data model is the foundation for enabling rich and flexible automation (see Figure 3). A data model defines the language between the operating system and the outside world of users and northbound systems. The data model defines the syntax and semantics on which to base applications, scripts and documentation.

In the past, operators relied on Simple Network Management Protocol (SNMP) data models, which are often proprietary, are not human readable and are inflexible to changes. As an alternative, operators often reverted to CLI scripting and “screen scraping” for automation. Although functional, neither of these options is scalable.

Nokia SR Linux features a complete and ground-up model-driven architecture that delivers flexible and simplified management and operations.

Figure 3. Ground-up, model-driven management infrastructure



Scalable streaming telemetry

SR Linux is designed to meet the demands of a model-driven world where visibility—and the scalability and granularity of that visibility—are paramount. SR Linux delivers an open, extensible and performant infrastructure that allows the retrieval of fine-grained system state, setting of configuration, and a scalable interface to support more granular data with push-based streaming.

SR Linux was built with an open, scalable telemetry framework at its core, where the system and applications internally speak the same transport and message-declaration mechanisms as used in gRPC, gRPC Network Management Interface (gNMI) and protobufs. Because SR Linux is natively model-driven, it is immediately ready for streaming telemetry without requiring any translation layers.

Software extensibility

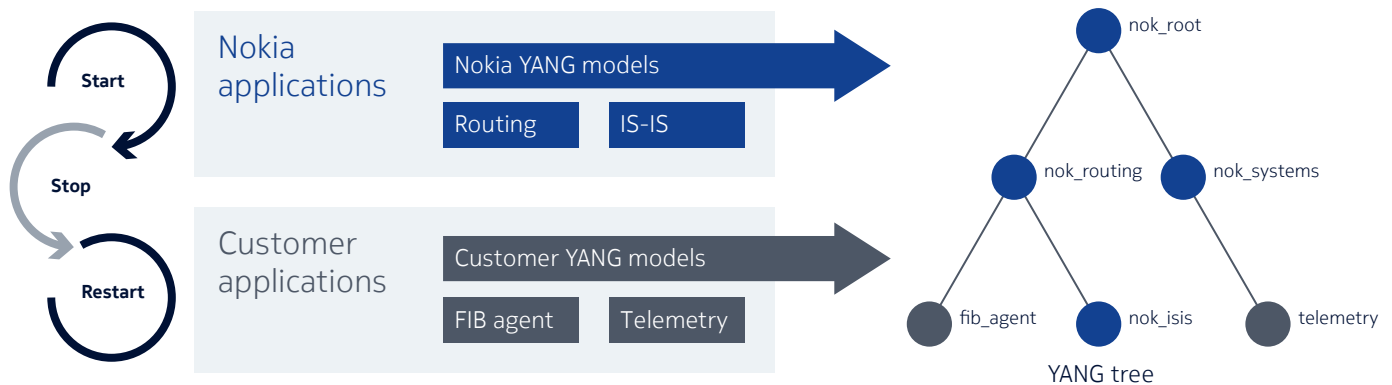
Nokia has built, from the ground up, a management architecture to meet the demands of a model-driven world where visibility—and the scalability and granularity of that visibility—are paramount. YANG has become the networking industry standard for data modeling because it is human readable, extensible and easy to learn.

Every SR Linux application (including third-party applications) supports its own YANG model, which can be loaded into the system. Operators can see and define the syntax and semantics of their application in a simple and standardized form. With this design, the YANG data model is defined first, and from it operators can derive the CLI, the APIs, the show-output formats and even the documentation of every feature of the system.

SR Linux handles management and operations through gNMI. Because SR Linux is natively model-driven, it is immediately ready for streaming telemetry without requiring any translation layers.

Third-party applications also have access to the full streaming telemetry framework. This enables applications developed for the SR Linux architecture to be operationalized, monitored, configured and debugged in the same way as any other application on the system, with standard open interfaces (see Figure 4).

Figure 4. Nokia SR Linux software extensibility



Superior CLI programmability

Every SR Linux application (including third-party applications) supports its own YANG model, which can be loaded into the system. With this design, the YANG data model is defined first, and from it, operators can derive the CLI, the APIs, the show-output formats and documentation related to software capabilities.

In addition to the gNMI interface, SR Linux includes an advanced, Python-based CLI and a JSON-RPC API for management. The CLI provides a flexible framework for accessing the system's underlying data models and is based on quality-of-life (features that improve the operator experience) embraced by DevOps communities.

Operators can also leverage plugins to completely customize the way the CLI operates, plugging in Linux commands, or pulling the state/configuration from various locations, combining them with system state/configuration to allow advanced logic. This capability streamlines the adoption of SR Linux because the interface can be customized.

The JSON-RPC API provides a modern interface that supports queries to the data models as well as allowing programmable interface access to the extensible plugin framework in the CLI. This access allows simplified use of operator customizations, tying them directly into the automation layer.

Enhanced NDK

Nokia SR Linux allows third-party applications to be fully integrated into the system with the same functionality as Nokia applications. This includes consistent configuration using YANG, telemetry support, life cycle management and visibility of system resources.

The Nokia NetOps Development Kit (NDK) enables application developers to leverage SR Linux's underlying model-driven architecture, with a simple, clean, decoupled integration. This allows all applications in the system to support data modeling, transactional configuration and—most important—massively scalable streaming telemetry.



The NDK uses gRPC and protobufs to provide maximum flexibility for languages supported and backwards compatibility. This approach differs from others, which are restricted to certain languages, versions and/or libraries.

The NDK allows data center operators to teach the network their language and respond directly to business demands—all without worrying about the scalability or functionality of the routing stack or the underlying infrastructure.

With the mindset to build for openness, common infrastructure utilized by Nokia applications is exposed, allowing a uniform operating model and visibility deep into the heart of the system.

Nokia Fabric Services Platform automation and operations toolkit

The Nokia Fabric Services Platform complements and extends the open Nokia SR Linux foundation to enable true LCM of data center fabrics. It delivers intent-based automation at scale for all phases of data center fabric operations, including Day 0 design, Day 1 deployment and Day 2+ configuration, operation, measurement and analysis of a data center fabric.

The platform implements a modern architecture based on a Kubernetes foundation combined with intent-based, distributed microservices and cloud-native approaches. It includes the Fabric Services Platform digital sandbox, an operational tool capable of emulating a data center fabric, application workloads and external BGP speakers.

The Fabric Services Platform represents intent and configuration state in a declarative way in YAML format, fitting into the bigger movement toward infrastructure as code. The platform provides the ability to validate designs, deploy fabric and application workload intent, and to make configuration changes and upgrades without risk.

For more information, read the [Nokia Fabric Services Platform Product Description](#).

Nokia Data Center routers

The Nokia portfolio of Data Center routers addresses the needs of modern data centers. The portfolio includes the Nokia 7250 IXR-10/IXR 6 Interconnect Routers for data center fabrics and the Nokia 7220 IXR routers for data center fabrics. The portfolio offers a broad range of high-performance platforms for data center leaf-spine deployments. Both modular and fixed-form-factor platforms are available, enabling data center network teams to choose the appropriate hardware while maintaining the same SR Linux NOS and its benefits.

For more information, see the [Nokia Data Center routers portfolio web page](#).

Summary

As the demands on data center networks continue to drive openness and efficiency, the Nokia Data Center Fabric solution, which includes the Nokia SR Linux NOS, the Nokia Fabric Services Platform and the Nokia Data Center routers, is ready to meet the challenge.

Nokia SR Linux is a truly open, extensible and resilient NOS. It implements a ground-up, model-driven architecture combined with proven and mature routing protocol stacks to create a unique foundation that delivers superior scalability, flexibility and efficiency in data center and cloud environments. This superior design foundation includes extensive telemetry, unrivalled third-party application support and plug-and-play integration—all critical features for modern data center networking.

Learn more

To learn more about the Nokia Data Center Fabric solution:

- Visit the [Nokia Data Center Fabric solution web page](#)
- See the [Nokia Data Center Fabric solution eBook](#)
- Read the [Nokia Fabric Services Platform Product Description](#)

Read the data sheets:

- [Nokia Service Router Linux](#)
- [Nokia 7250 IXR-10/IXR-6 Interconnect routers for SR Linux](#)
- [Nokia 7220 IXR-D series Interconnect routers for SR Linux](#)

Abbreviations

API	application programming interface	MP-BGP	Multiprotocol Border Gateway Protocol
ASIC	application-specific integrated circuit	MPLS	multiprotocol label switching
BGP	Border Gateway Protocol	NDK	SR Linux NetOps Development Kit
CLI	command line interface	NMS	network management system
DevOps	development and operations	NOS	network operating system
EVPN	Ethernet virtual private network	OSF	Open Software Foundation
FIB	forwarding information base	Pub/sub	publish/subscribe
gNMI	gRPC Network Management Interface	protobuf	protocol buffer
gRPC	generalized Remote Procedure Call	RPC	remote procedure call
IDB	Nokia Impart Database	SDK	software development kit
IGP	Interior Gateway Protocol	SR Linux	Nokia Service Router Linux
IP	Internet Protocol	SR OS	Nokia Service Router Operating System
IS-IS	Intermediate System to Intermediate System	SSH	Secure Shell
JSON	JavaScript Object Notation	VXLAN	Virtual eXtensible Local Area Network
LCM	life cycle management	XDP	eXtensible Data Path
LLDP	Link Layer Discovery Protocol	YANG	Yet Another Next Generation (data modeling language)



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 is a registered trademark of Nokia Corporation. Other product and company names mentioned herein may be trademarks or trade names of their respective owners.

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

© 2020 Nokia

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

Document code: SR2007045884EN (August) CID207604