



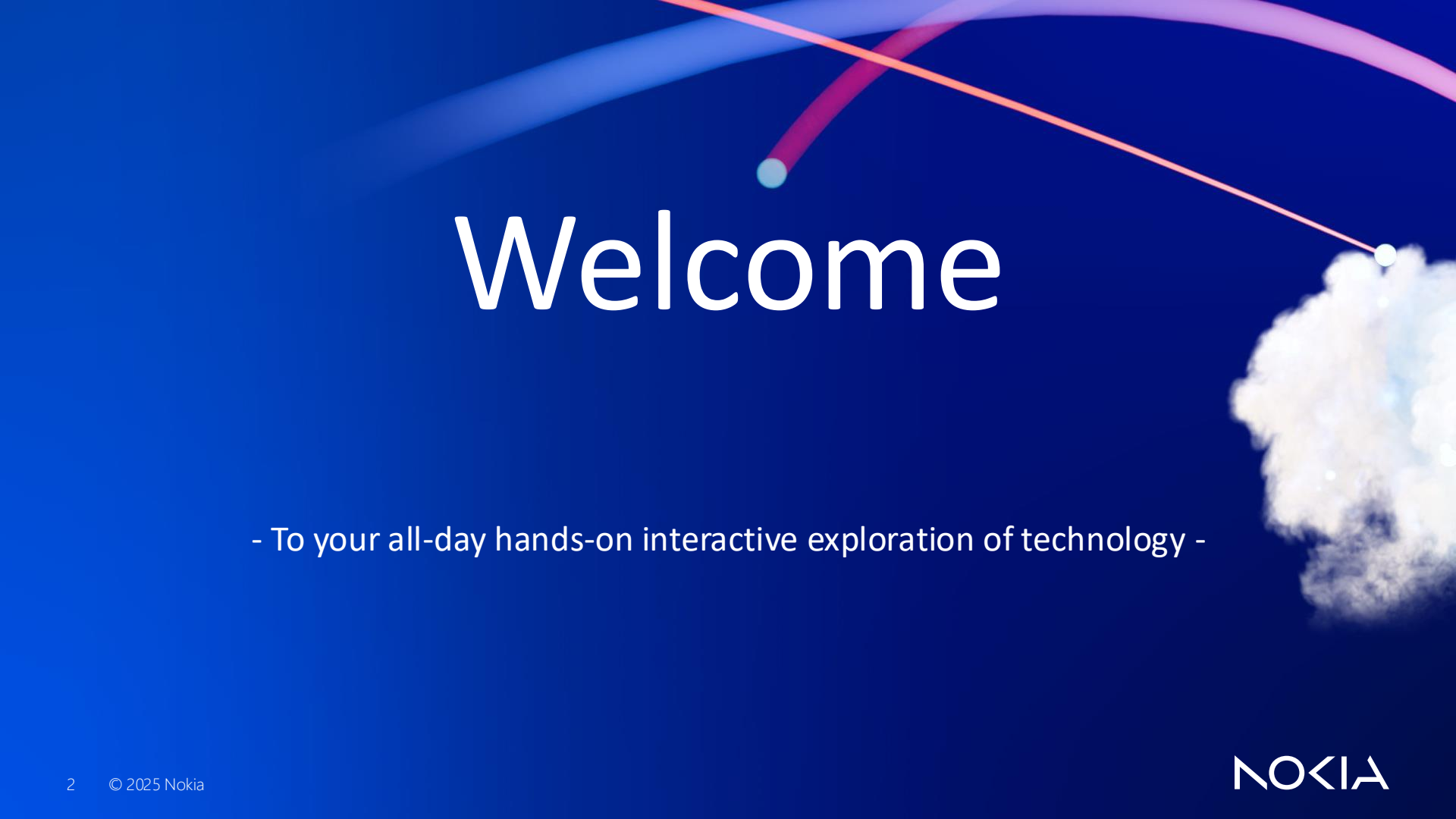
SReXperts²⁰²⁵
Network the cloud

SReXplore 2025 Primer

Event Driven Automation (EDA)

Roman Dodin

Zeno Dhaene



Welcome

- To your all-day hands-on interactive exploration of technology -

SReXperts²⁰²⁵

Network the cloud





SReXplore

What's in store

- Event schedule
 - **Morning: SReXplore Primer Sessions**
Equip you with the knowledge to participate in the main hackathon event
 - **Afternoon: SReXplore**
This is why you're here. Hands on, do anything, try anything, break anything session to test your skills or learn new ones. Fun quiz to close the day.
- Event notices
 - Fire
 - Coffee
 - Restrooms



NOKIA

SReXplore

What do you need

- Yourself
- An open mind and a willingness to try new things
- Your trusty laptop (and charger)
 - SSH client
 - Web Browser
- A small amount of pre-requisite knowledge



Don't panic, that is why you are here!



NOKIA



SSID: SReXperts

Password: Noki@2025



Presentation live sharing on Teams

<https://tinyurl.com/SRX-APAC-HackathonPrimer-EDA>

Event Driven Automation Stream



Hackathon Primer Agenda

1. An introduction to EDA
2. Declarative Intents
3. Configuring Fabrics
4. Virtual Networks
5. Automation Capabilities
6. Tips & Tricks
7. Ready for the afternoon



What is EDA?

The Nokia Data Center Fabric solution

Build and operate at scale with confidence

Event Driven Automation
Automation & operations
toolkit



Intent-driven
design and
operations



Digital sandbox
for true
emulation



Cloud-native
integration
model



FOCUS

Cloud-native, Kubernetes-based extensible foundation

Service Router Linux
Network operating
system

SR
Linux

Open, scalable
telemetry
framework



Open-source
CLI plugins
(Python)



NetOps
Development
Kit (NDK)



Standard
Linux
kernel



Ground-up,
model-driven
foundation



Resilient,
field-proven
protocol stacks

SR
OS

7250 IXR, 7220 IXR
Data center platforms



Based on
merchant
silicon



Common
hardware and
software design



Full range of DC
applications



Nokia Event Driven Automation

“Nokia EDA raises the bar on data center network operations with a modern approach that builds on Kubernetes to bring highly reliable, simplified, and adaptable lifecycle management to data center networks. ”

- Marketing

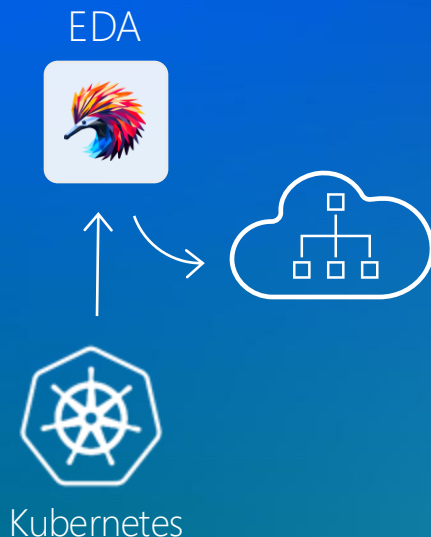
Nokia Event Driven Automation

“Nokia EDA raises the bar on data center network operations with a modern approach that **builds on Kubernetes** to bring highly reliable, simplified, and adaptable lifecycle management to data center networks. ”

- Marketing

Nokia Event Driven Automation (EDA)

Bringing the power of Kubernetes to network automation



1. Builds on Kubernetes as a **platform**
 - Cloud-based, microservices architecture
2. Brings Kubernetes **concepts** to networking
 - Intent-based, declarative, event-driven, revision control
3. Leverages Kubernetes **Resource Model**
 - Tooling and APIs for network consumption

You can see
and you must try

Try EDA

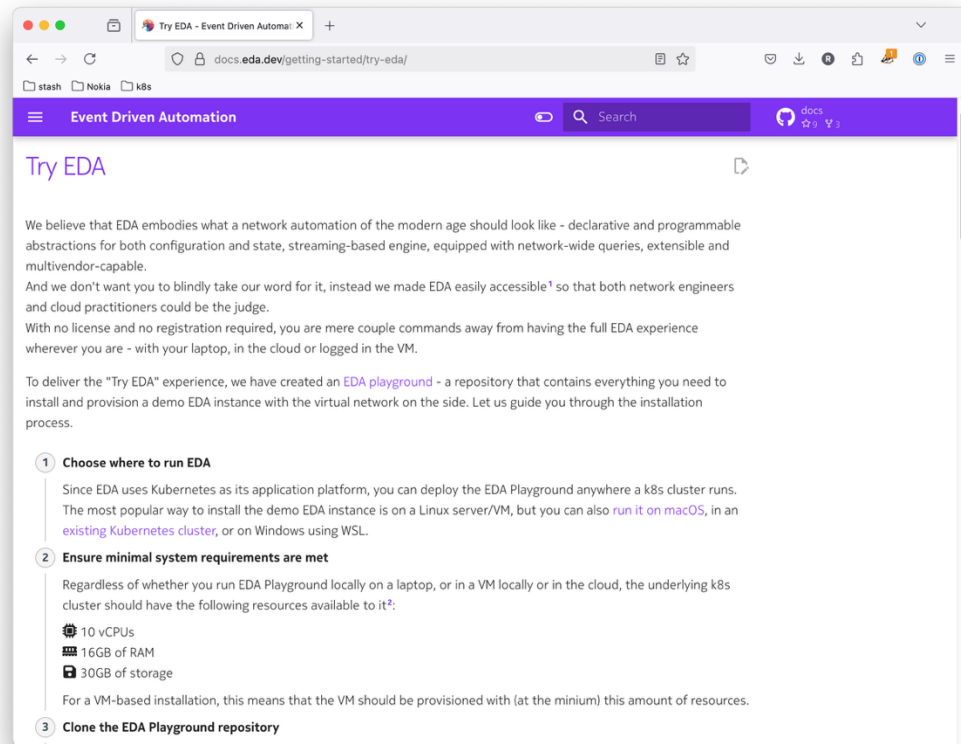
Powerful framework made easy to try

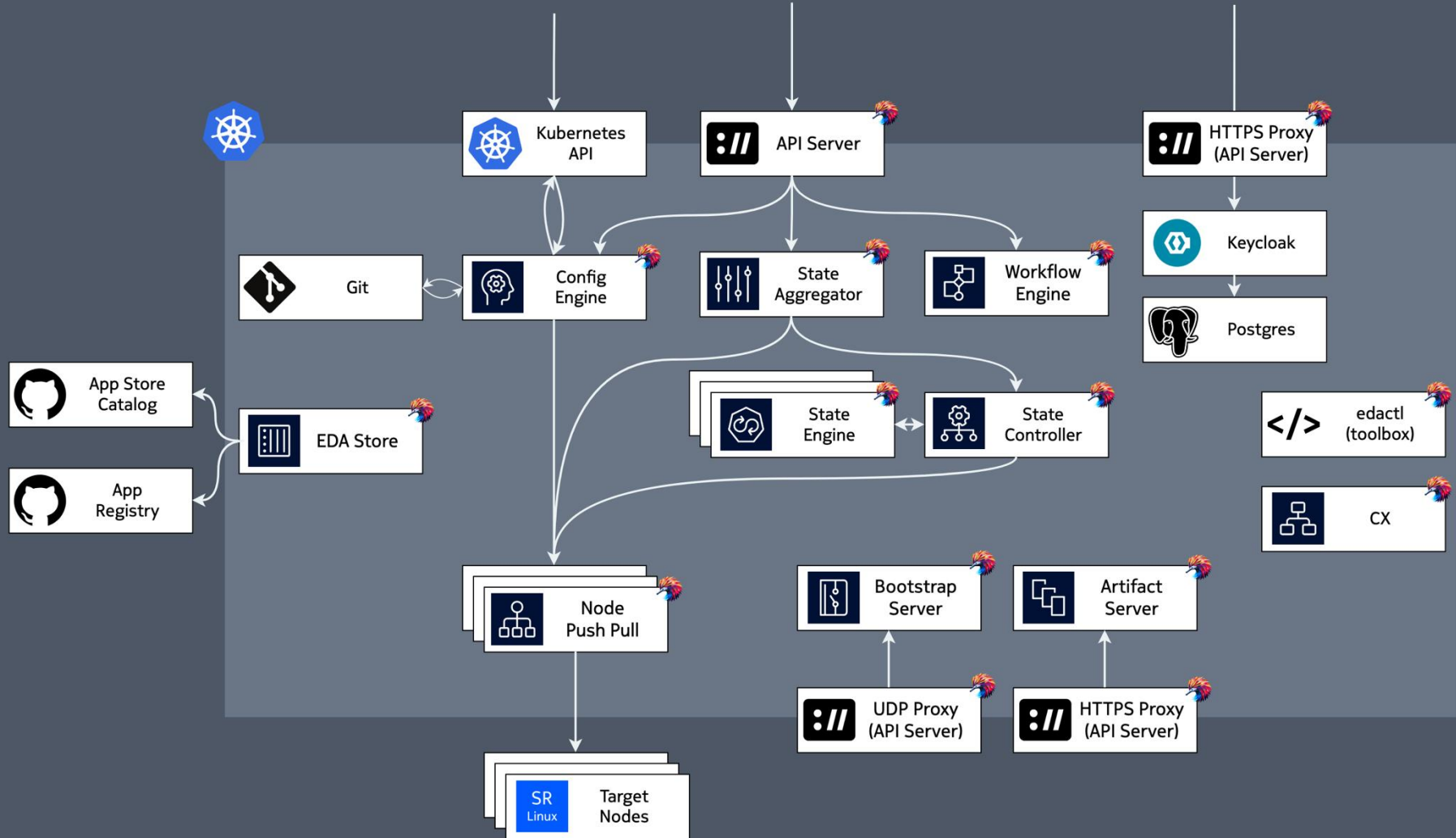
- The **new reality**: If you can't try it – it doesn't exist
- Try EDA sets up the full EDA system in an unattended way
 - No registration, license required
 - **make try-eda** is all it takes
- It is the **only** system of that scale that offers the unlicensed and unlimited try experience
- We **want** you to play with EDA whenever and wherever you like

EDA Playground



- Documentation for engineers by engineers
- Based on the proven success of [learn.srlinux.dev](#)
- Workflow/task-oriented documentation
- Generated from markdown
 - Powered by the popular open-source doc engine
 - Open to changes from internal/external audience
- Responsive, clean and lightweight design
- Continuous, non-release-bound updates





EDA Playground



KinD cluster



Nokia EDA

EDA Apps

app: interfaces

interface

lag

app: fabric

app: services

CX / Digital Twin

SR
Linux

SR
Linux

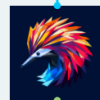
SR
Linux

Access

API client / UI



OpenAPI REST



gNMI

YANG



Reuse, not reinvent

Leveraging k8s

- Unlimited deployment options
- Well-known and adopted
- Horizontal scale out of the box
- Easy to onboard OSS components



The screenshot shows the Kubernetes dashboard interface for the 'eda-system' namespace. The left sidebar contains a navigation menu with categories like Cluster, Workloads, and Networking. The 'Deployments' section is selected. The main table lists 20 deployments, each with columns for Name, Ready status, Desired count, Up-to-date status, Available status, CPU usage, Memory usage, and Age. All deployments are in a 'Ready' state with a '1/1' ratio and are approximately 4 days old.

Name ↑	Ready	Desired	Up-to-date	Available	CPU Usa...	Memory Usa...	Age
eda-api	1/1	1	1	1	-	-	4d 20h
eda-appstore	1/1	1	1	1	-	-	4d 20h
eda-asvr	1/1	1	1	1	-	-	4d 20h
eda-bsvr	1/1	1	1	1	-	-	4d 20h
eda-ce	1/1	1	1	1	-	-	4d 20h
eda-cert-checker	1/1	1	1	1	-	-	4d 20h
eda-fe	1/1	1	1	1	-	-	4d 20h
eda-fluentd	1/1	1	1	1	-	-	4d 20h
eda-git	1/1	1	1	1	-	-	4d 20h
eda-git-replica	1/1	1	1	1	-	-	4d 20h
eda-keycloak	1/1	1	1	1	-	-	4d 20h
eda-kx	1/1	1	1	1	-	-	4d 19h
eda-metrics-server	1/1	1	1	1	-	-	4d 20h
eda-notifier	1/1	1	1	1	-	-	4d 20h
eda-postgres	1/1	1	1	1	-	-	4d 20h
eda-px	1/1	1	1	1	-	-	4d 19h
eda-sa	1/1	1	1	1	-	-	4d 20h
eda-sc	1/1	1	1	1	-	-	4d 20h
eda-toolbox	1/1	1	1	1	-	-	4d 20h
trust-manager	1/1	1	1	1	-	-	4d 20h

Nokia Event Driven Automation

“Nokia EDA raises the bar on data center network operations with a modern approach that builds on Kubernetes to brings highly reliable, **simplified**, and adaptable lifecycle management to data center networks. ”

- Marketing

Networks are all the same

Well... the building blocks

Interface

NTP Client

BGP Peer

MAC-VRF

OSPF Area

ISIS Neighbor

NETCONF Server

IP-VRF

ISIS Net

BGP Group

Route Reflector

Static Route

Simple is hard



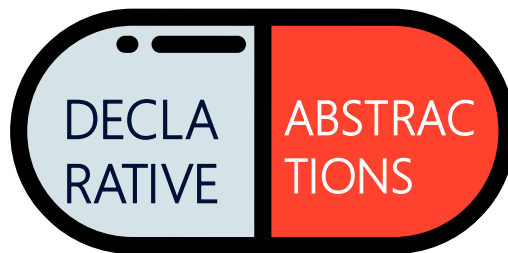
□ Configure NTP □



- Parse input & Validate input
- Identify target nodes
- Translate input to node-specific commands
- Push config
- Ensure it is accepted
- Rollback everything if rejected
- Display state of the configured object
- Monitor config deviations

Simple is hard

But not impossible



What, instead of how



```
kind: NTPClient
spec:
  router: mgmt
  servers:
    - iBurst: true
      server: 10.128.1.1
```

- Omit implementation details
- Assume “meaningful” defaults
- Reuse lower—level abstractions
- Abstraction level vs fine-tuning

The Custom Resource

```
apiVersion: timing.eda.nokia.com/v1alpha1  
kind: NTPClient
```

```
metadata:  
  name: my-ntp-client  
  namespace: eda
```

```
spec:  
  router: mgmt-router  
  routerKind: MANAGEMENTROUTER  
  servers:  
    - iBurst: true  
      preferred: true  
      server: 10.171.10.1
```

```
status:  
  lastChange: '2025-05-17T12:52:15.000Z'  
  operationalState: up
```

1

2

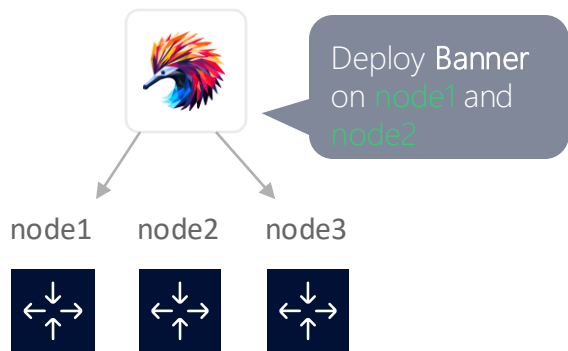
3

4

**Group / Version / Kind**

Strict vs Loose

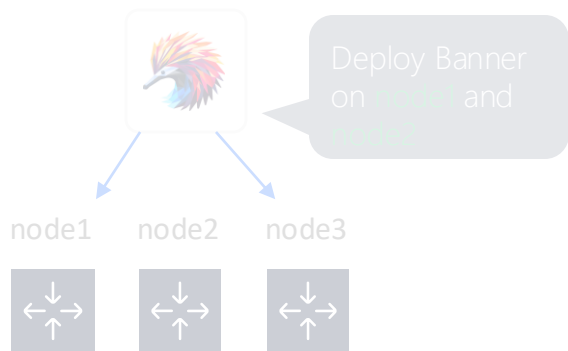
Strict but static



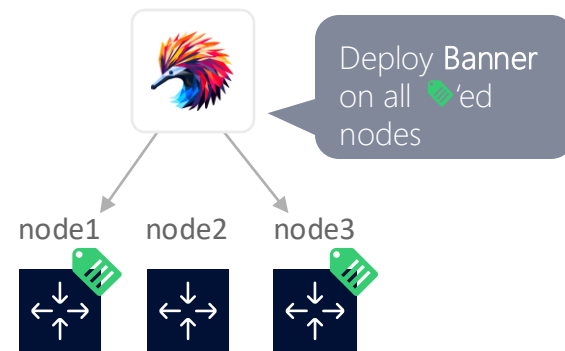
- Granular selection
- Static

Strict vs Loose

Loose but scalable

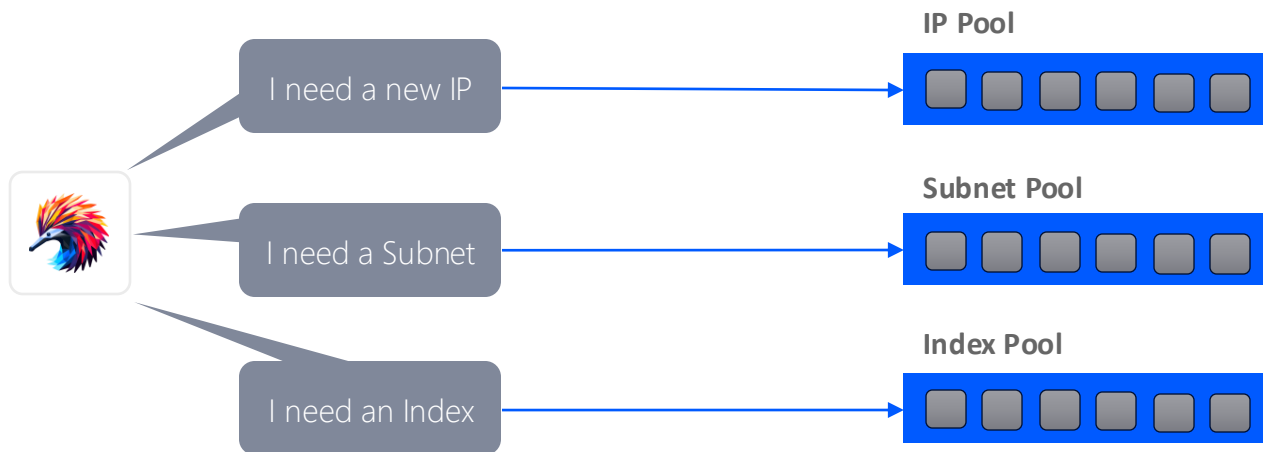


- Granular selection
- Static

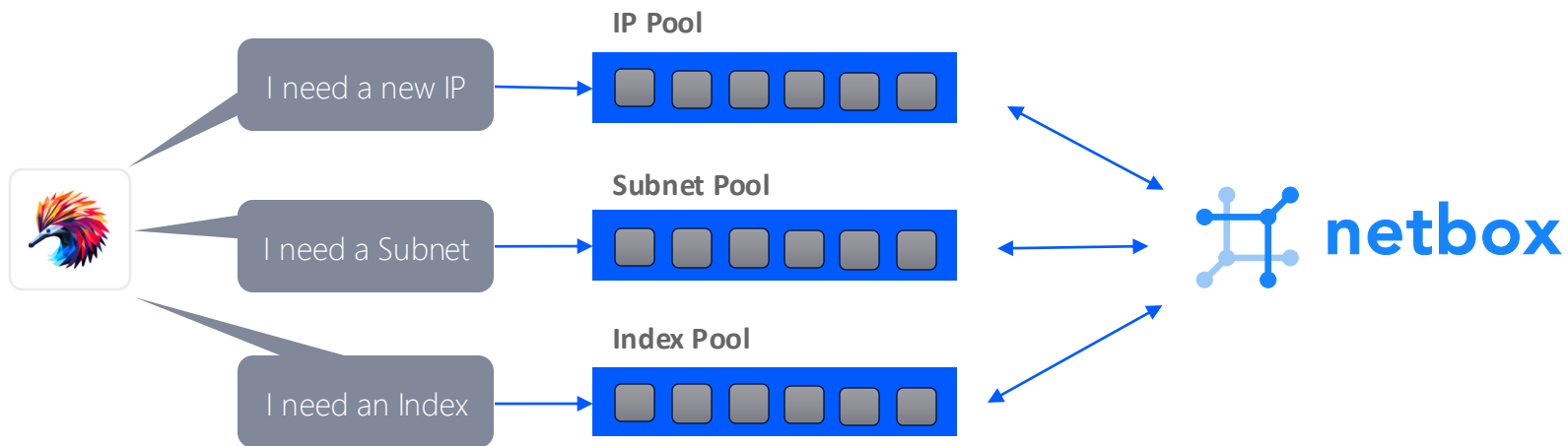


- Scaled selection
- Dynamic

Allocation pools



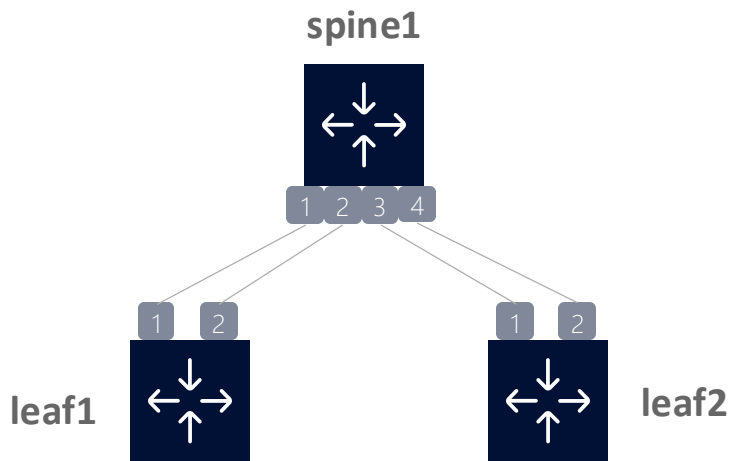
Allocation pools



Building a Fabric

Current logical topology

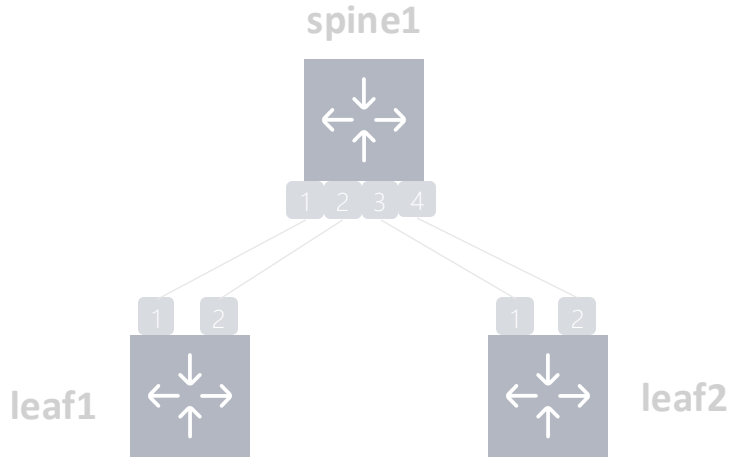
Current



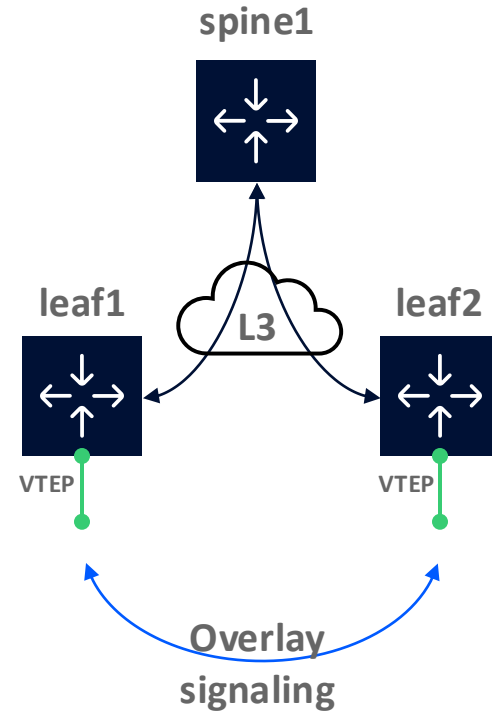
Building a fabric

Target logical topology

Current



Target



Composing a fabric

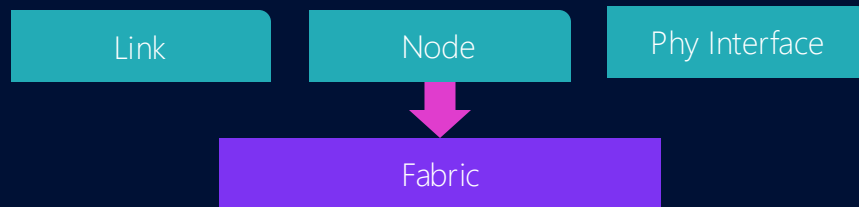
(Re)usable abstractions



Fabric

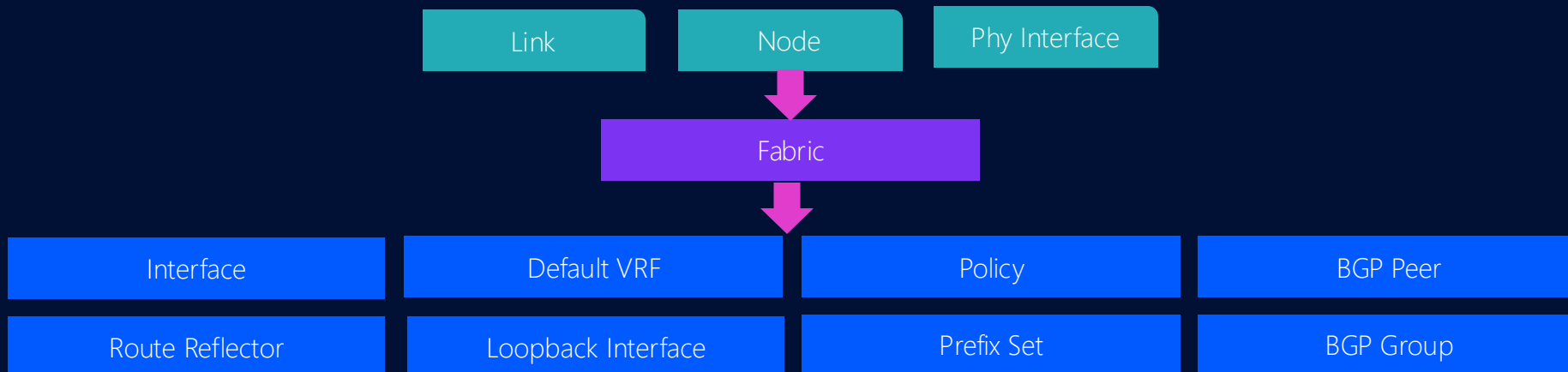
Composing a fabric

(Re)usable abstractions



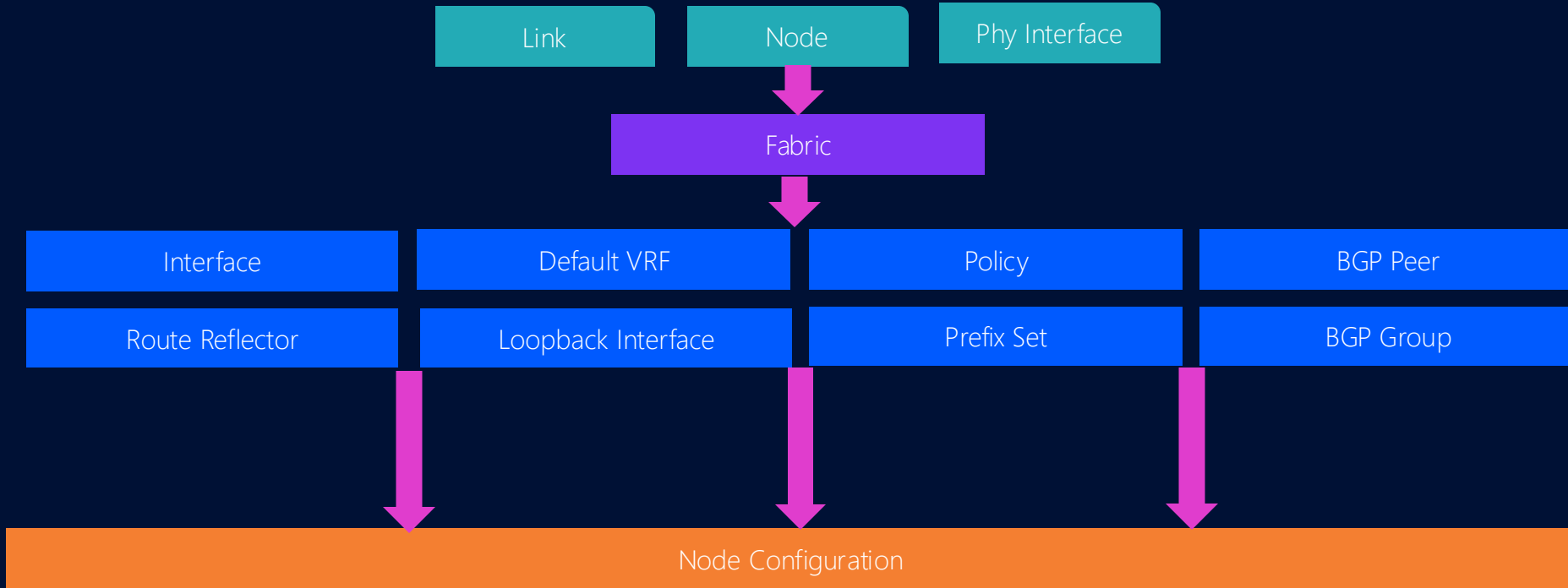
Composing a fabric

(Re)usable abstractions

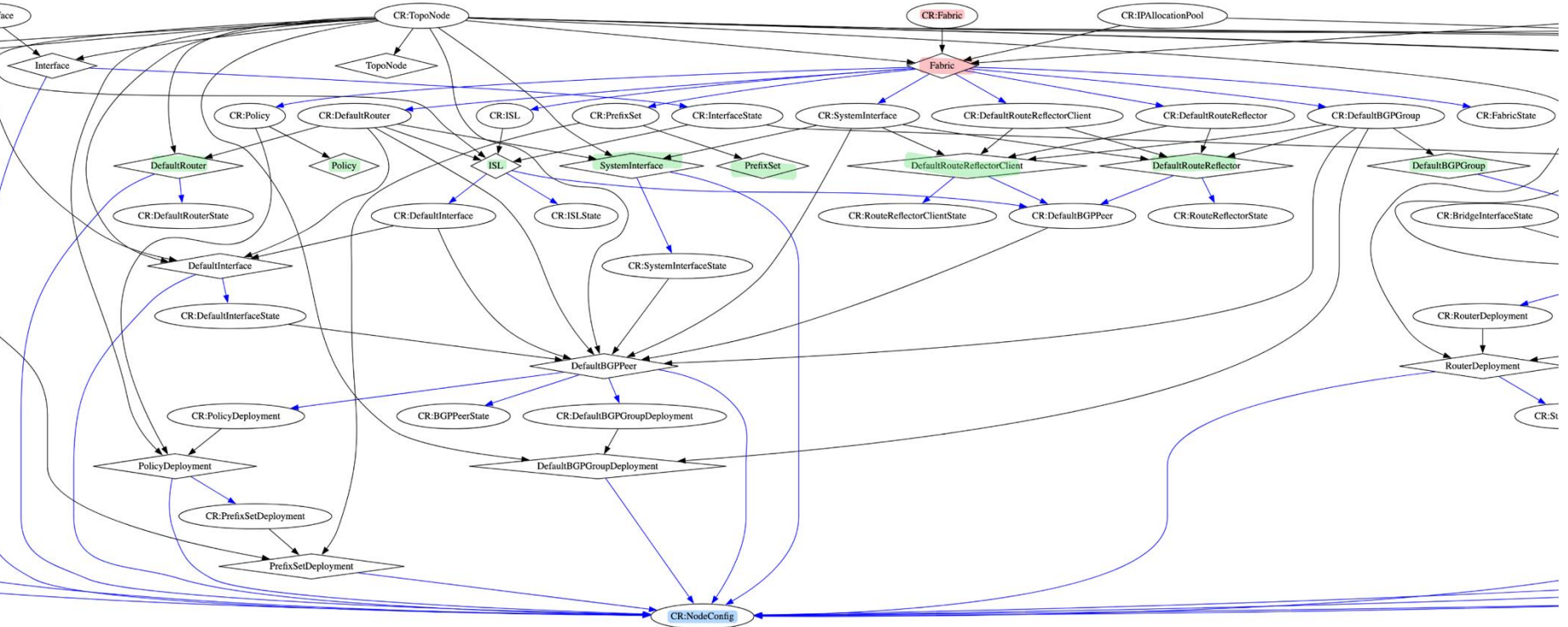


Composing a fabric

(Re)usable abstractions

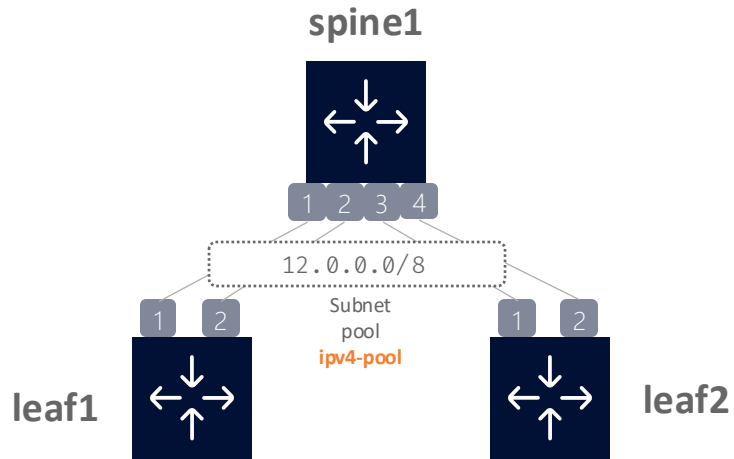


But "simple" is hard



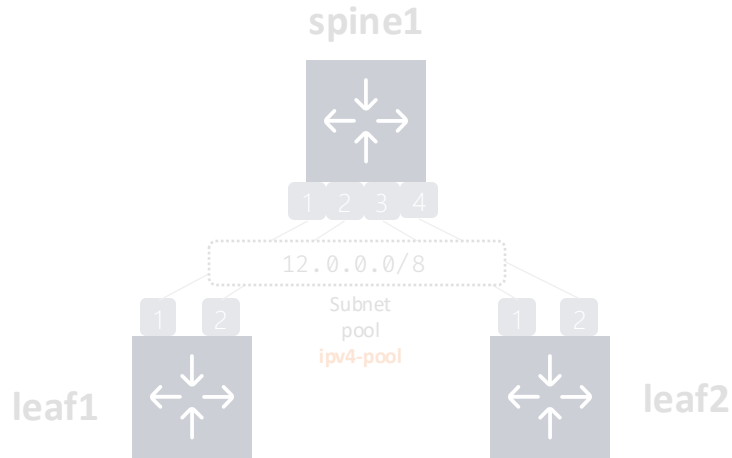
From 4 to 6 in 5 seconds

Current

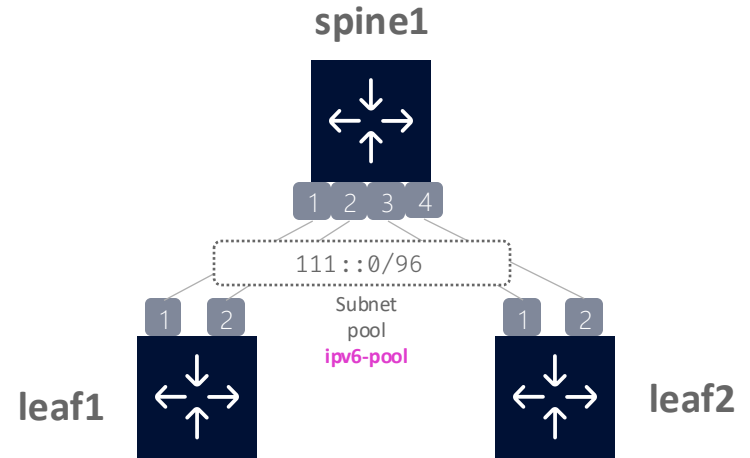


From 4 to 6 in 5 seconds

Current



Target

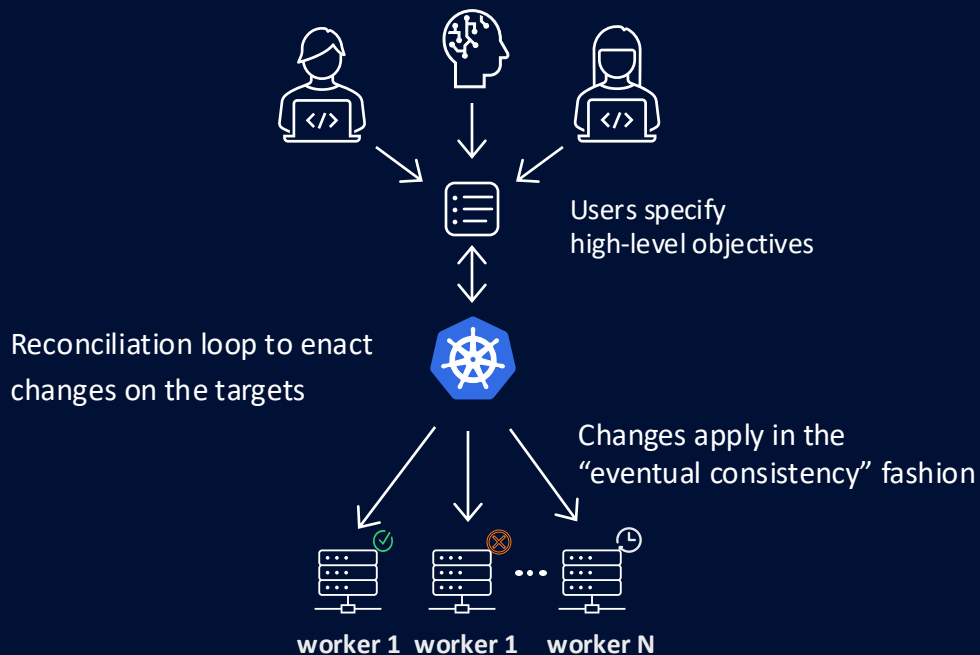


Nokia Event Driven Automation

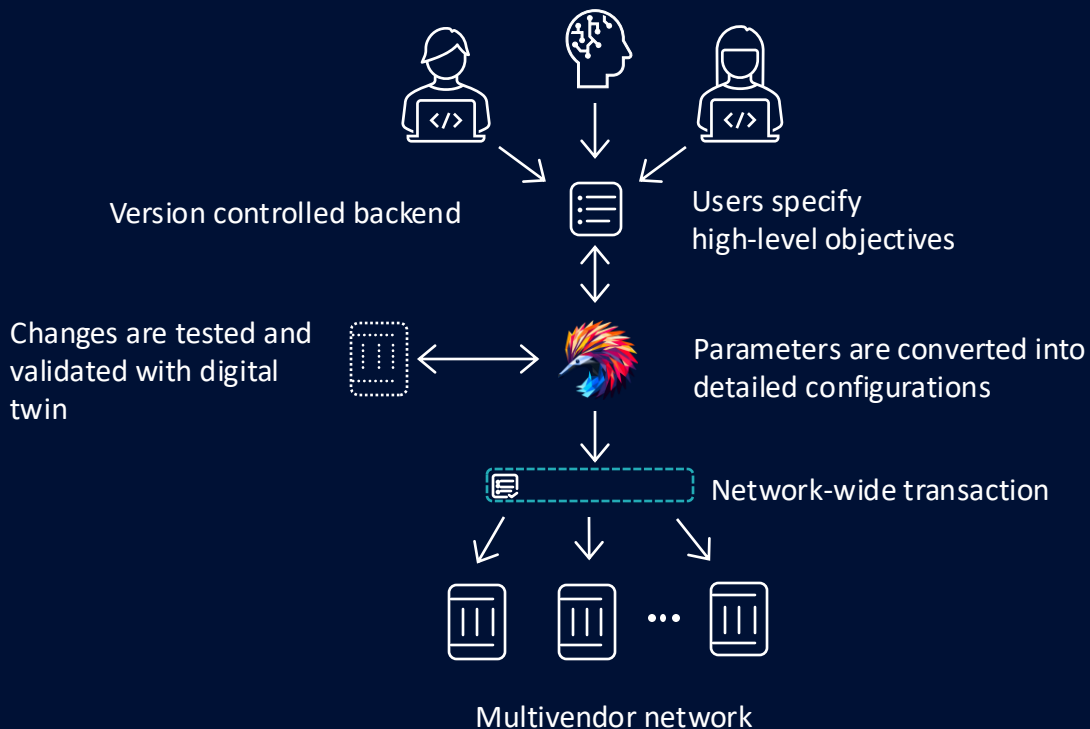
“Nokia EDA raises the bar on data center network operations with a modern approach that builds on Kubernetes to brings highly reliable, simplified, and adaptable lifecycle management to data center networks. ”

- Marketing

Reconciliation is nice!



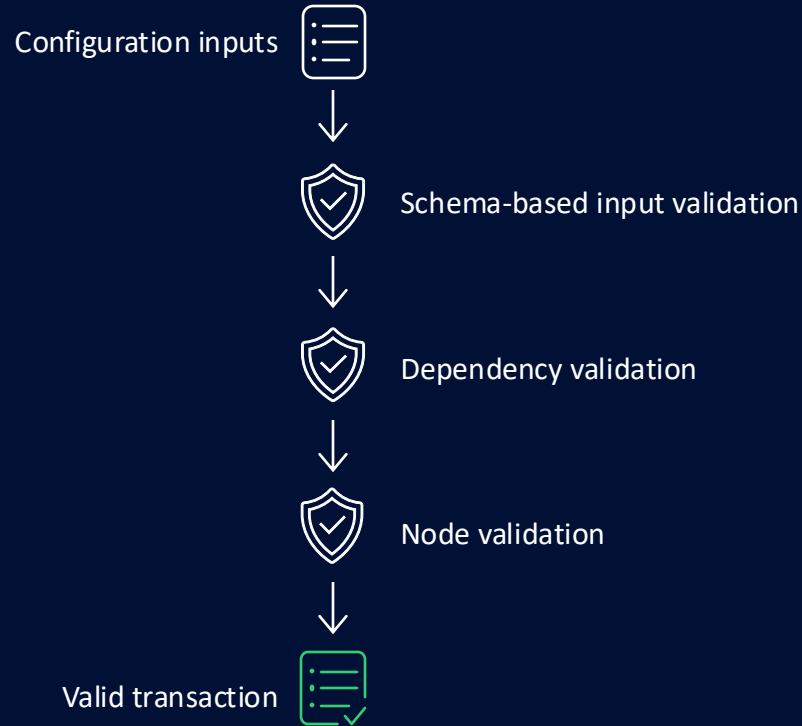
But networks need transactions



Reliability in operations is achieved by making the configurations **predictable** and the process orderly and **controlled**

- ✓ Intent-based networking
- ✓ Network-wide transactions
- ✓ Digital twin, CI/CD pipeline
- ✓ Revision control, rollbacks
- ✓ State-aware, event-driven

Shared Fate Transactions



- ✓ Allow for grouping of configuration changes
- ✓ Auto roll-back on any failure
- ✓ All or nothing approach
- ✓ Persistent log in Git

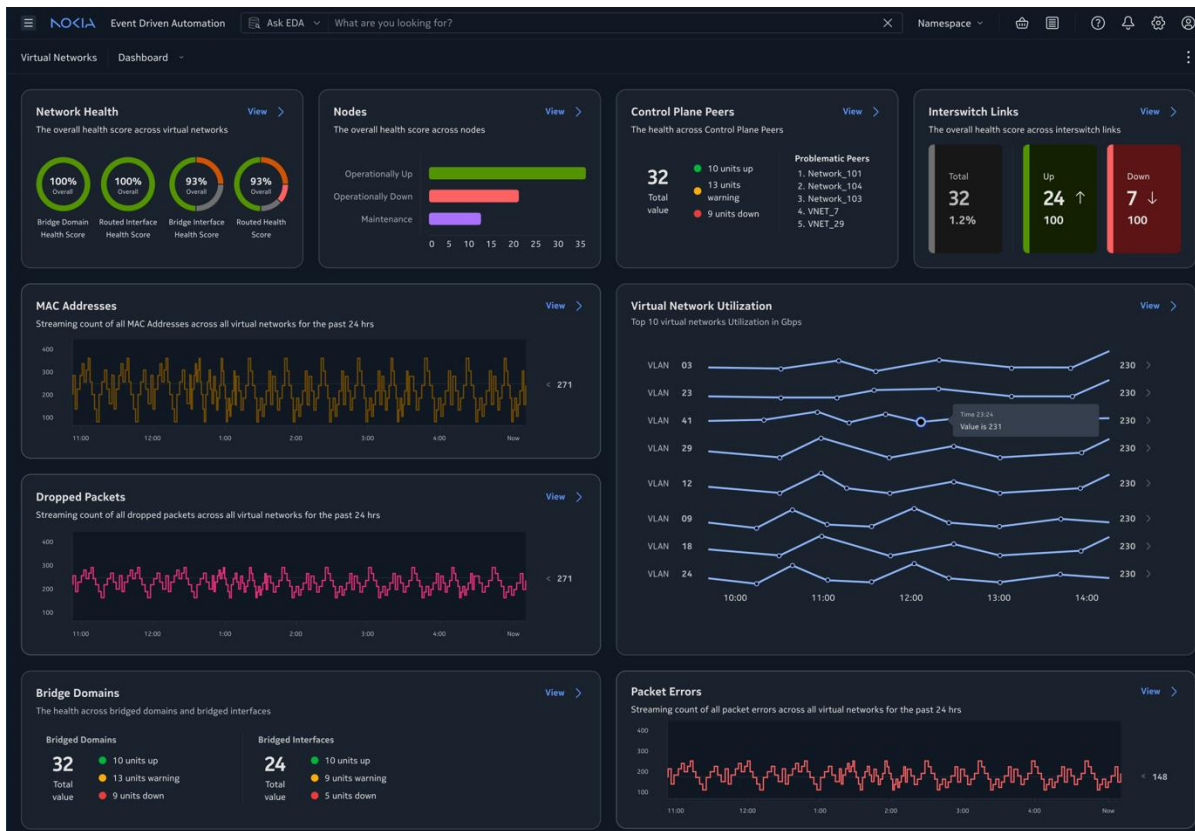
Run it Dry and check the Diffs



The great divide



Single dashboard to view health



- Customizable view
- Real-time monitoring
- Improved visibility
- Increased efficiency
- Faster incident response
- Enhanced collaboration
- Better decision-making

What if you could run queries?

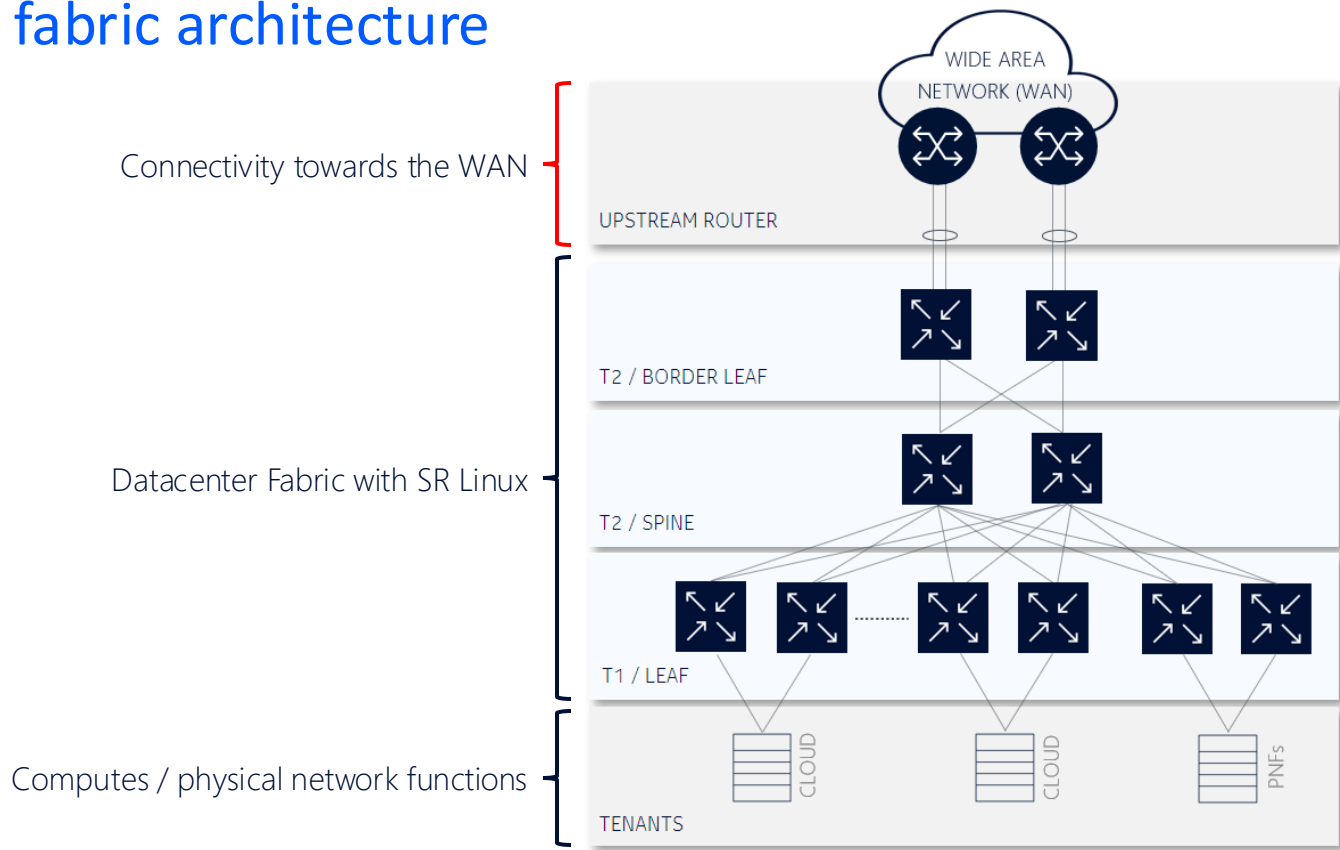
Over your whole network

Realtime + network-wide + operands
=
EDA Query Language

Virtual Networks

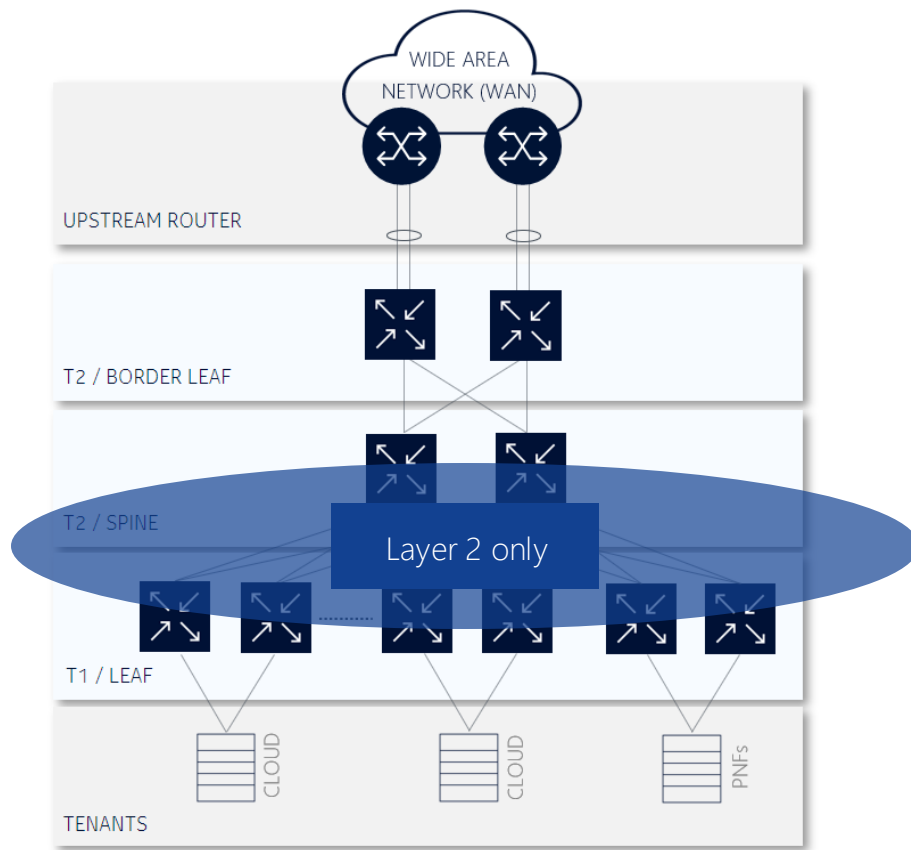
An abstract graphic on a dark blue background. It features several curved lines in shades of blue and purple. A thin red line with a small white dot at its end extends from the upper right towards a white, fluffy cloud on the right side of the image. Another thin red line with a small blue dot at its end extends from the upper left towards the center.

Datacenter fabric architecture



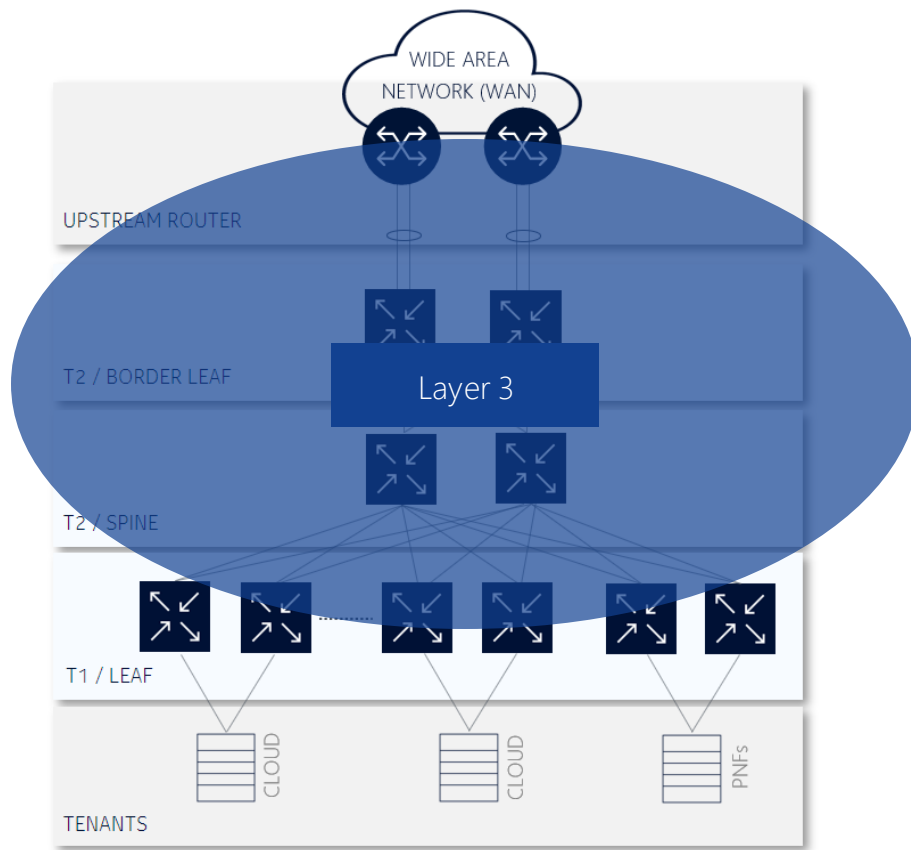
Workload types

- Layer 2 broadcast domains
- Infrastructure VPNs
- Not generally extended to the WAN



Workload types

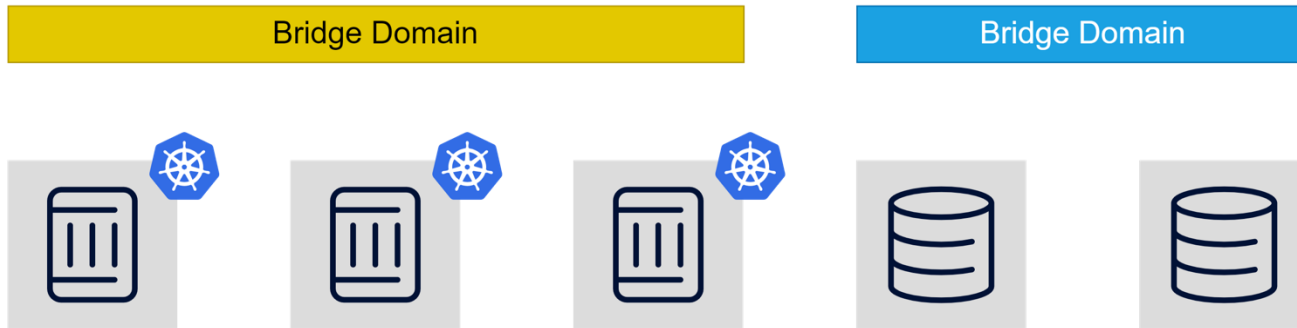
- Layer 3 routed networks
- Terminate layer 2 networks
- Borderleafs terminate VxLAN traffic
- Advertised as BGP-IPv4 to the WAN



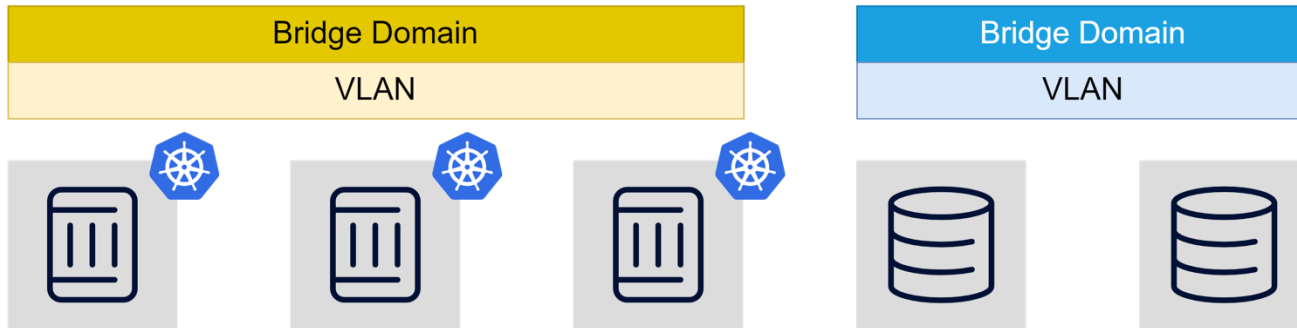
Abstraction



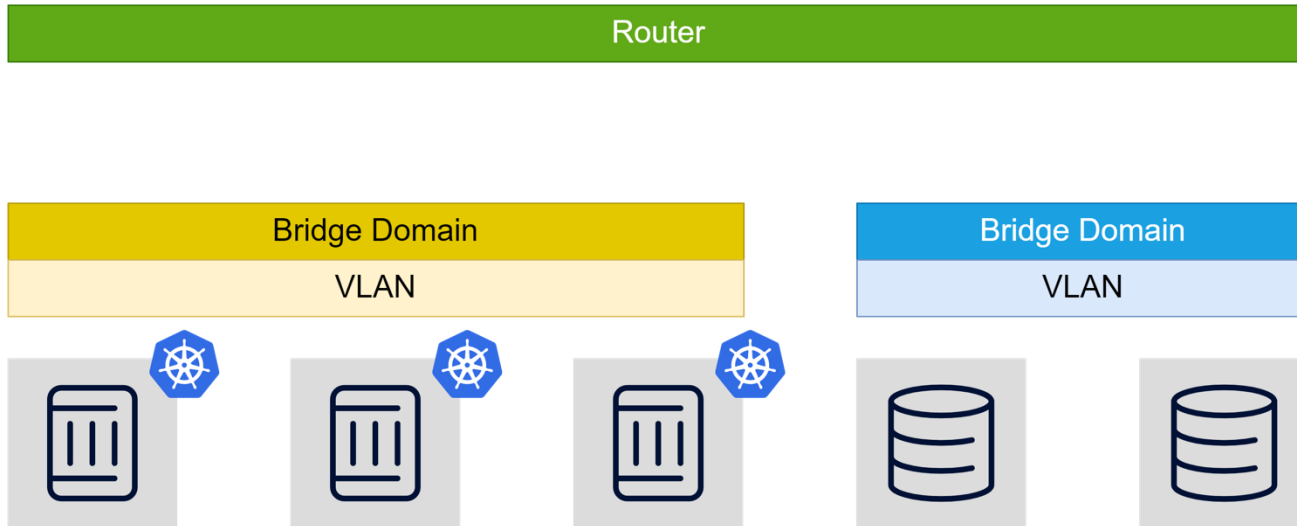
Abstraction



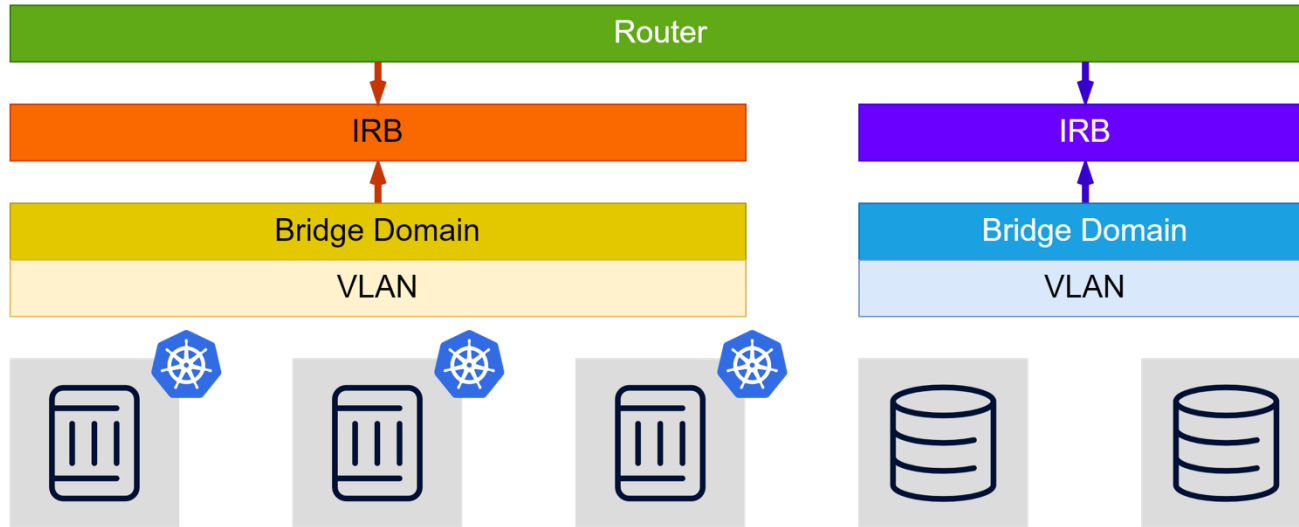
Abstraction



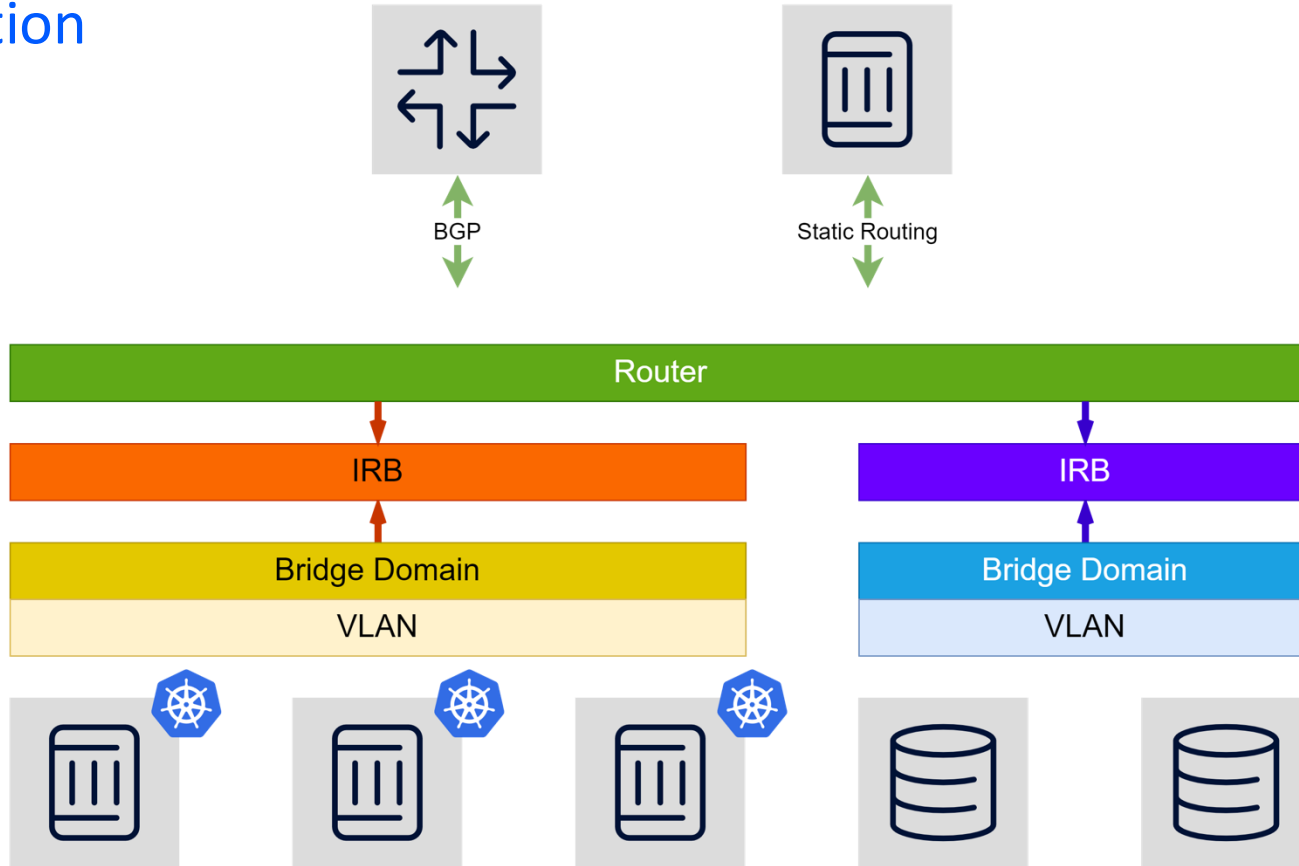
Abstraction



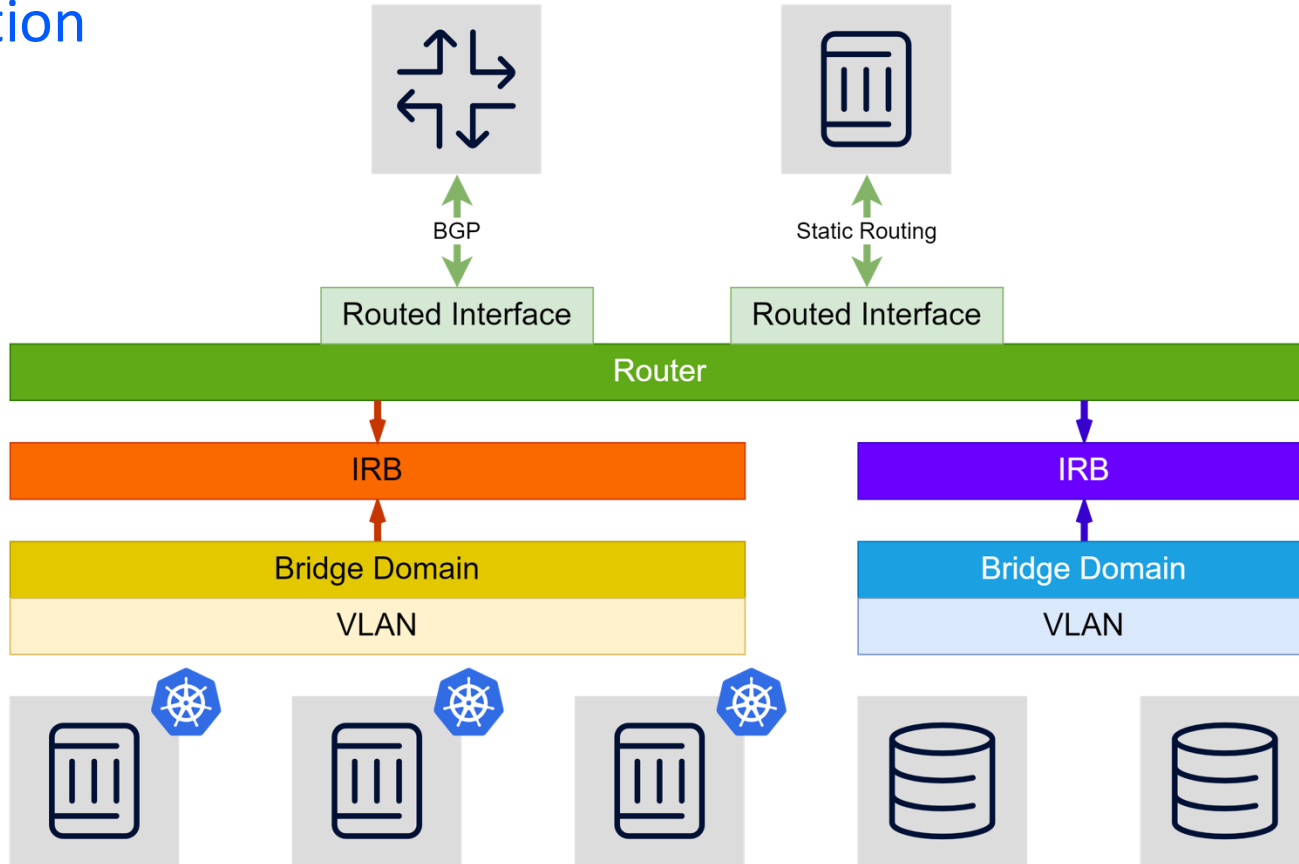
Abstraction



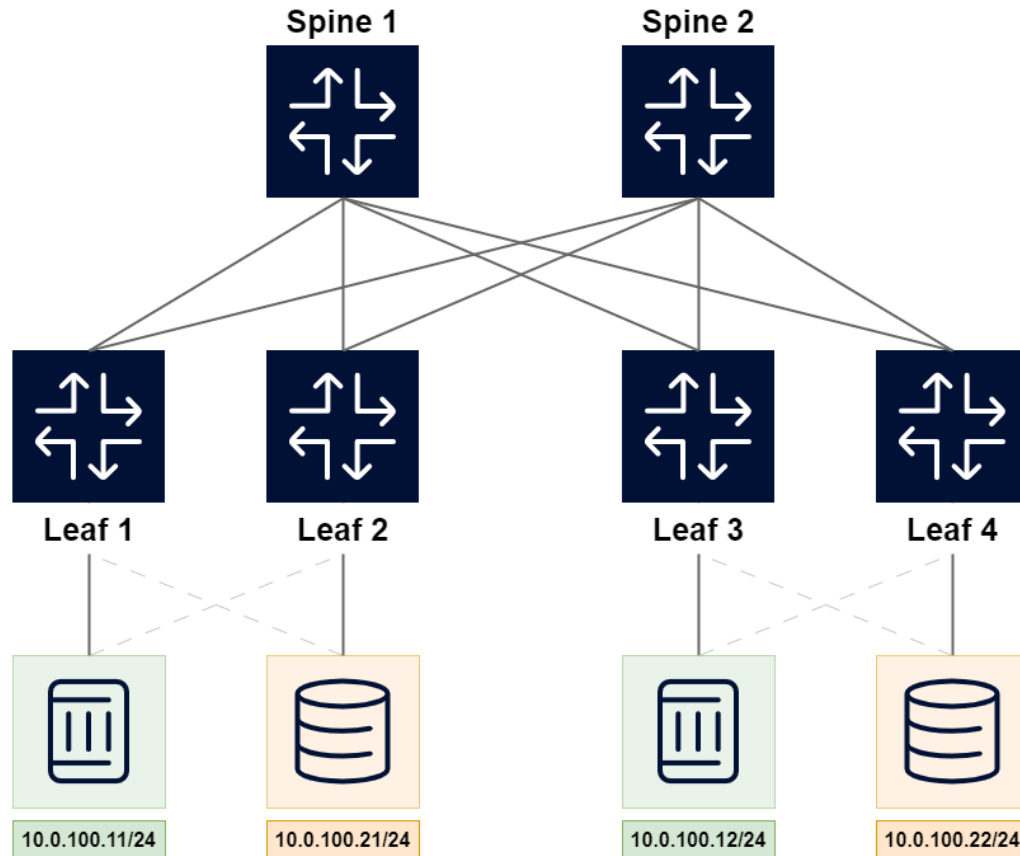
Abstraction



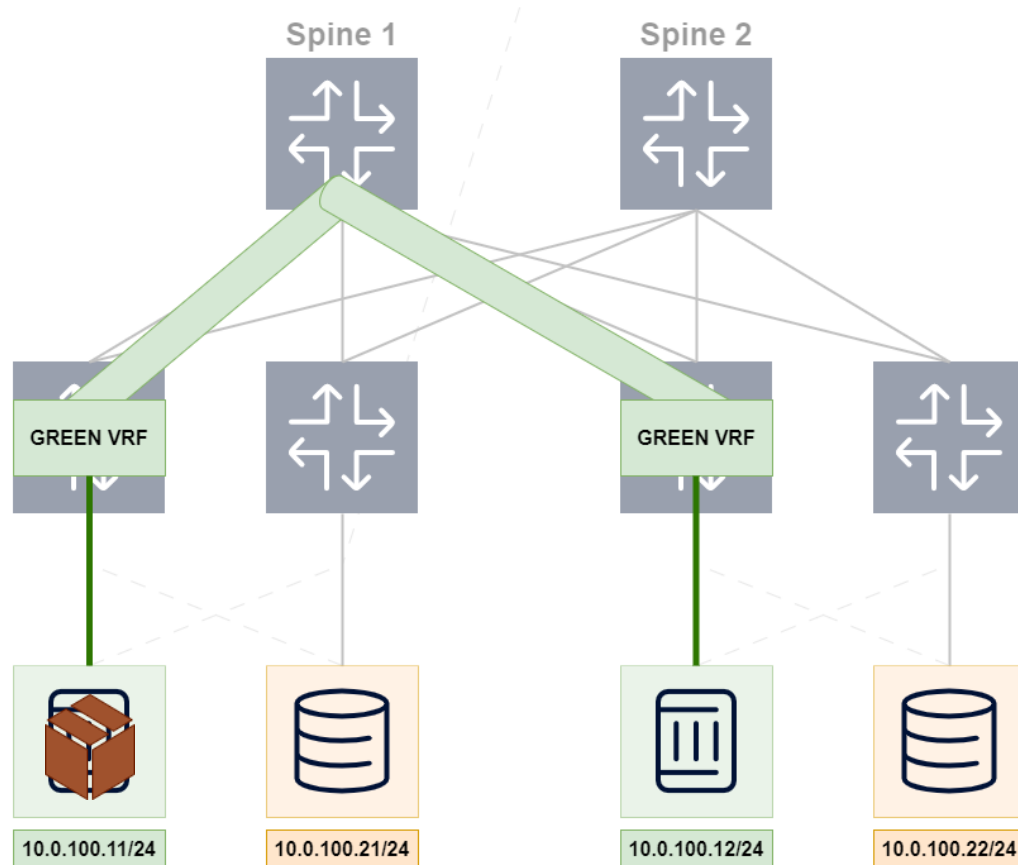
Abstraction



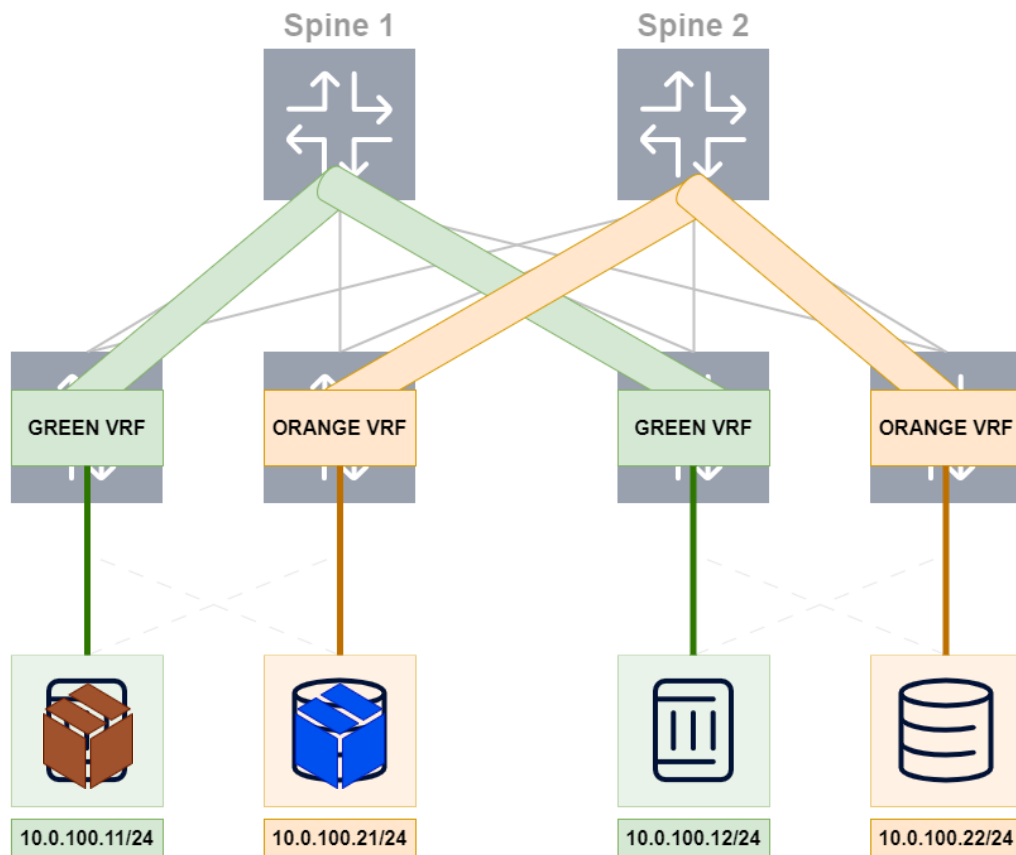
Demo setup



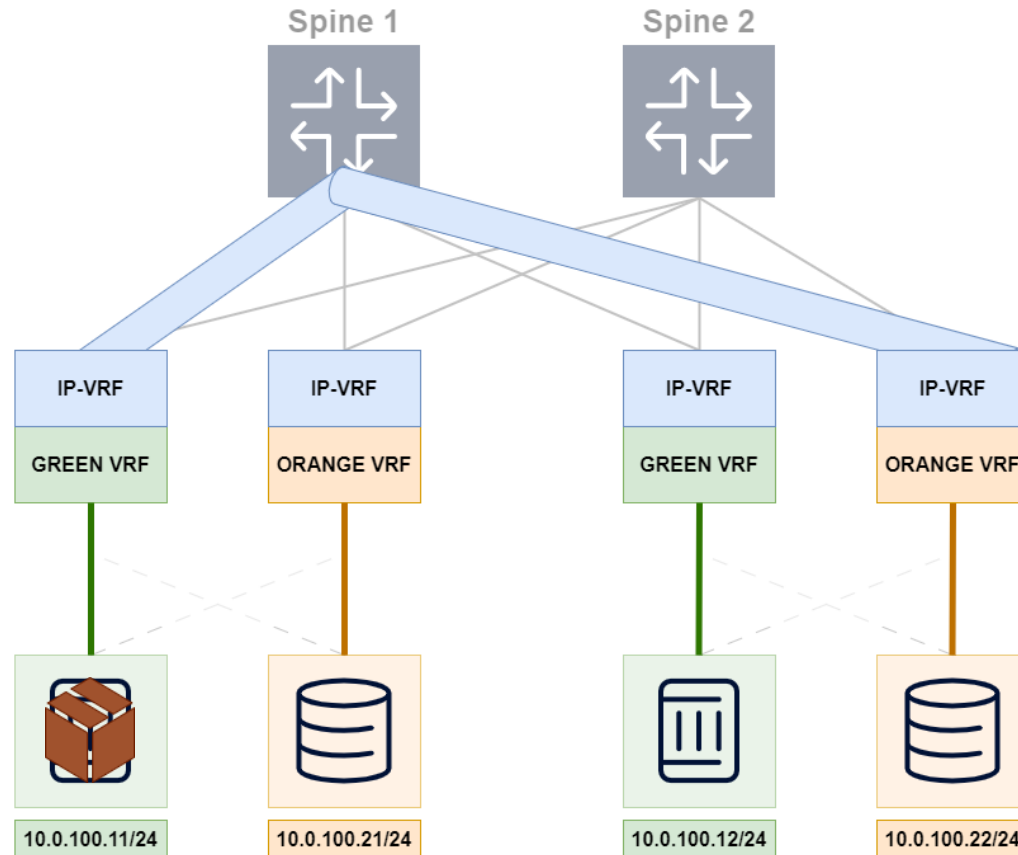
Connectivity between hosts in the same broadcast domain



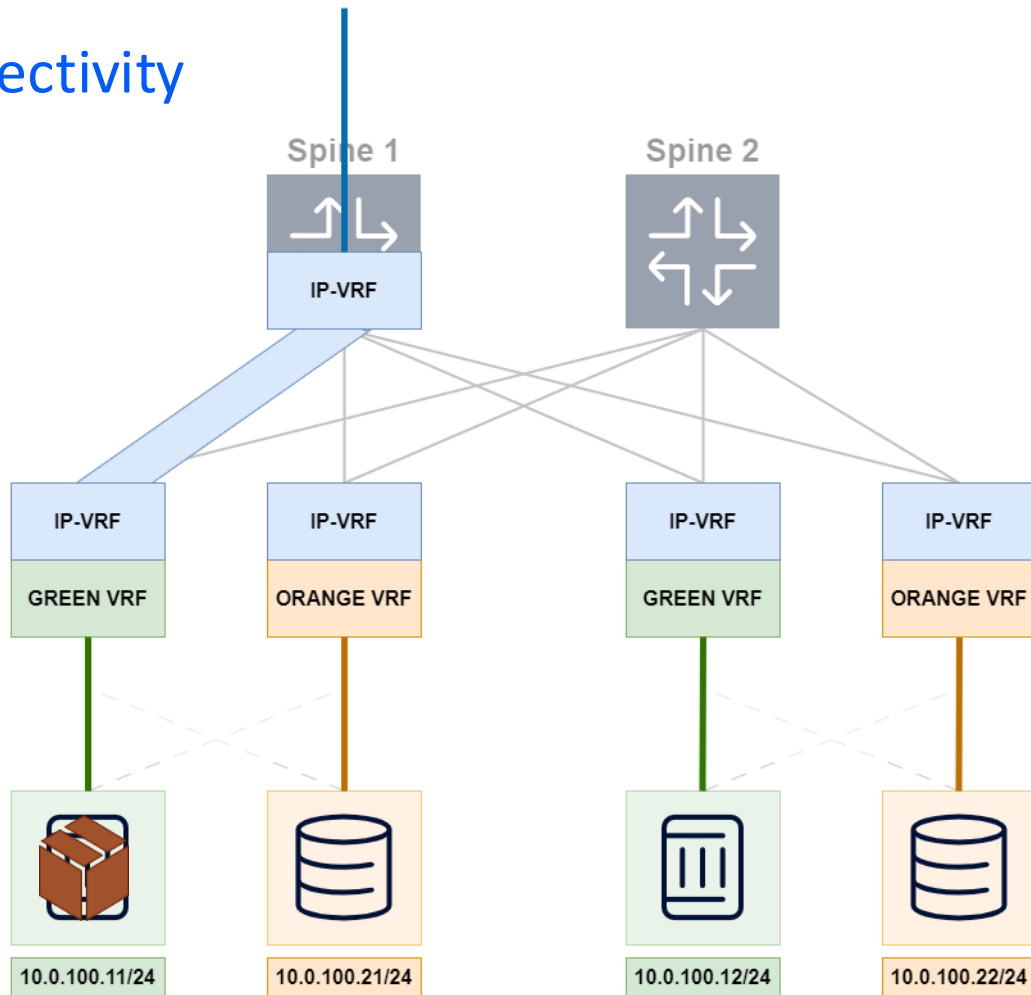
Connectivity between hosts in the same broadcast domain



Connectivity between subnets



External connectivity



Nokia Event Driven Automation

“Nokia EDA raises the bar on data center network operations with a modern approach that builds on Kubernetes to brings highly reliable, simplified, and adaptable lifecycle management to data center networks. ”

- Marketing

Everything is an app

Script-based

Fabrics

Filters

Services

- Typically represents a managed object
 - BGP peer
 - ACL rule
 - Fabric
 - etc
- Written in uPython
- Runs in Config/State engine
- Leverage EDA APIs exclusively

Everything is an app

Script-based

Fabrics

Filters

Services

- Typically represents a managed object
 - BGP peer
 - ACL rule
 - Fabric
 - etc
- Written in uPython
- Runs in Config/State engine
- Leverage EDA APIs exclusively

Controller-based

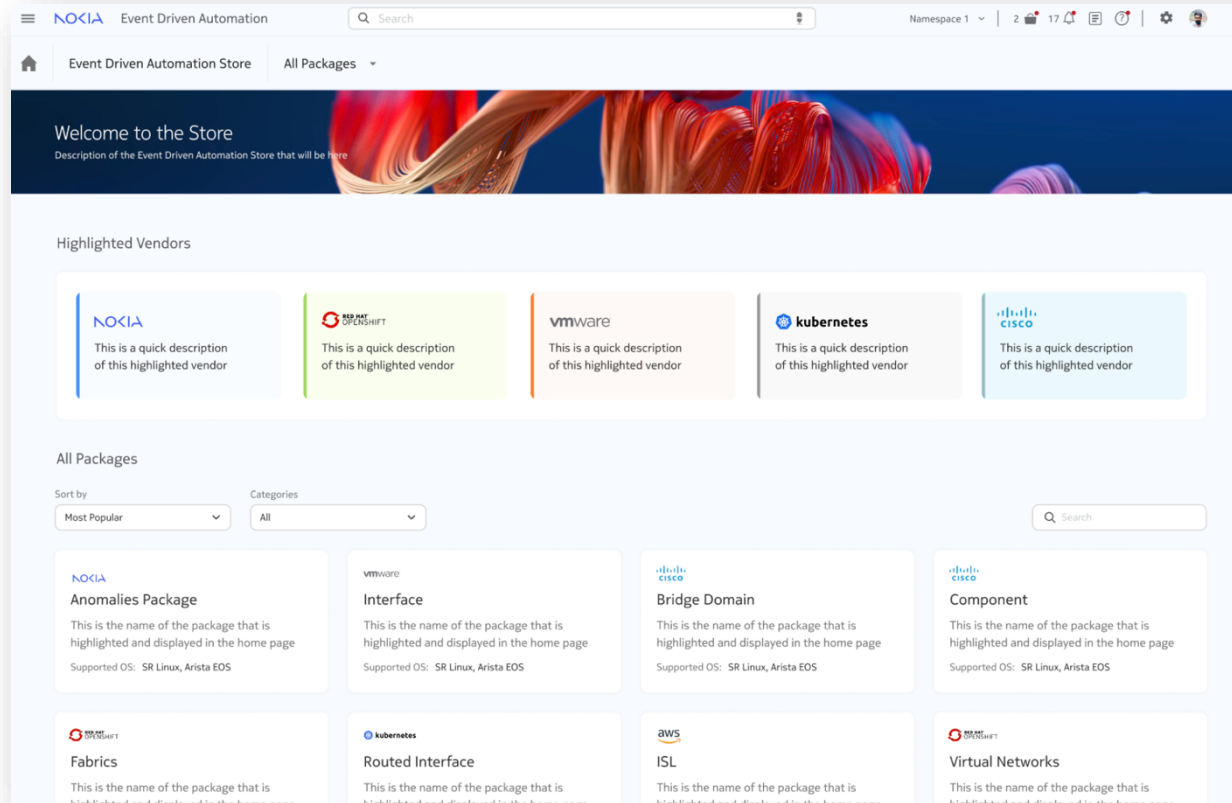
Prometheus Exporter

Service Now

Netbox Sync

- Often represents an integration application
 - Exporting/Importing data
 - Integrating with 3rd party API
 - Complex automation workflow
- Written in any language
- Runs natively on k8s
- Can access EDA APIs and any other API

Everything is an app. And EDA store is where you get them



Declarative abstraction – but at what cost?

Abstract
input



Black
box

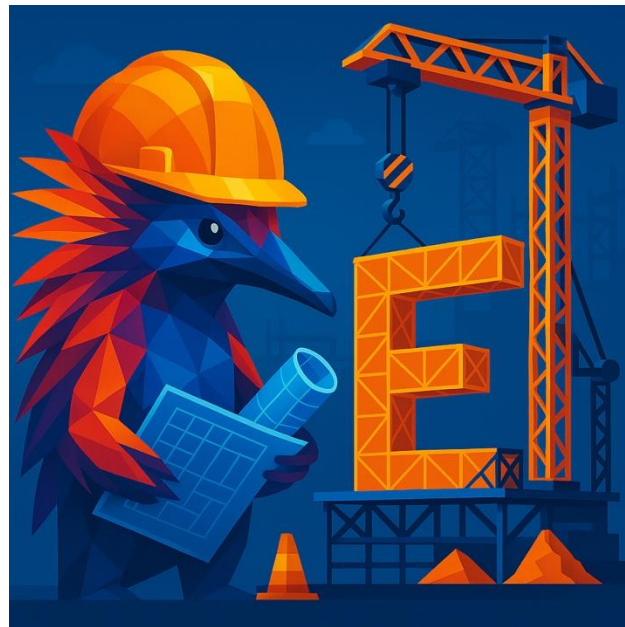


How can we avoid the black box?

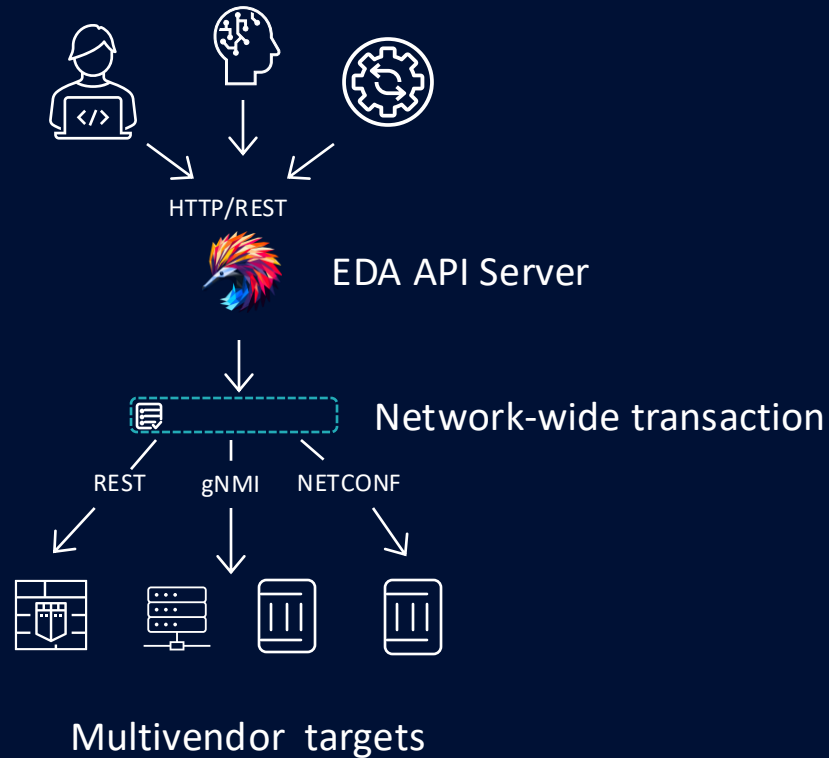
EDABuilder

- Extensibility via Apps is the EDA's holy grail
- But EDA apps are not "just scripts"
 - API definitions
 - Config/State intents
 - Normalization
 - Alarms
 - UI views and dashboards
 - Workflows
 - Package descriptors
 - Docs
- **edabuilder** is a CLI tool to scaffold all the app components
- Same tool is used by Nokia and you!

<https://docs.eda.dev/development/apps/>

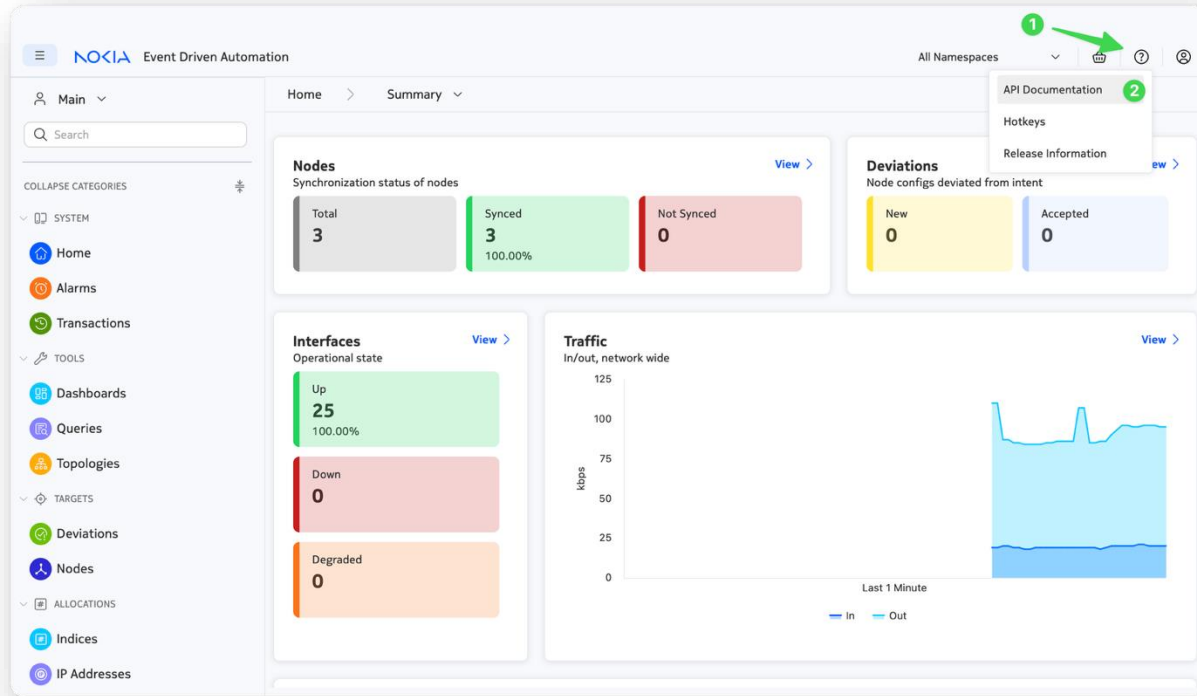


EDA is API ~~first~~ only



API documentation

UI



API documentation

UI

Discovery API

```
curl -k -s https://${EDA_URL}/openapi/v3 \  
--header 'Content-Type: application/json' \  
--header 'Authorization: Bearer ***'
```

API documentation

UI

Discovery API

Community repo

 github.com/eda-labs/openapi

 README  Apache-2.0 license

EDA OpenAPI specs

Unofficial catalog of the OpenAPI specs for the [Nokia EDA](#).

- Core
 - [core](#)
 - [apps/core](#)
- [AAA](#)
- [AlFabrics](#)
- [AppStore](#)
- [Bootstrap](#)
- [Certcheck](#)
- [Components](#)
- [Config](#)

Report repository

Releases 2

 **v25.4.1** Latest
3 weeks ago

[+ 1 release](#)

Packages

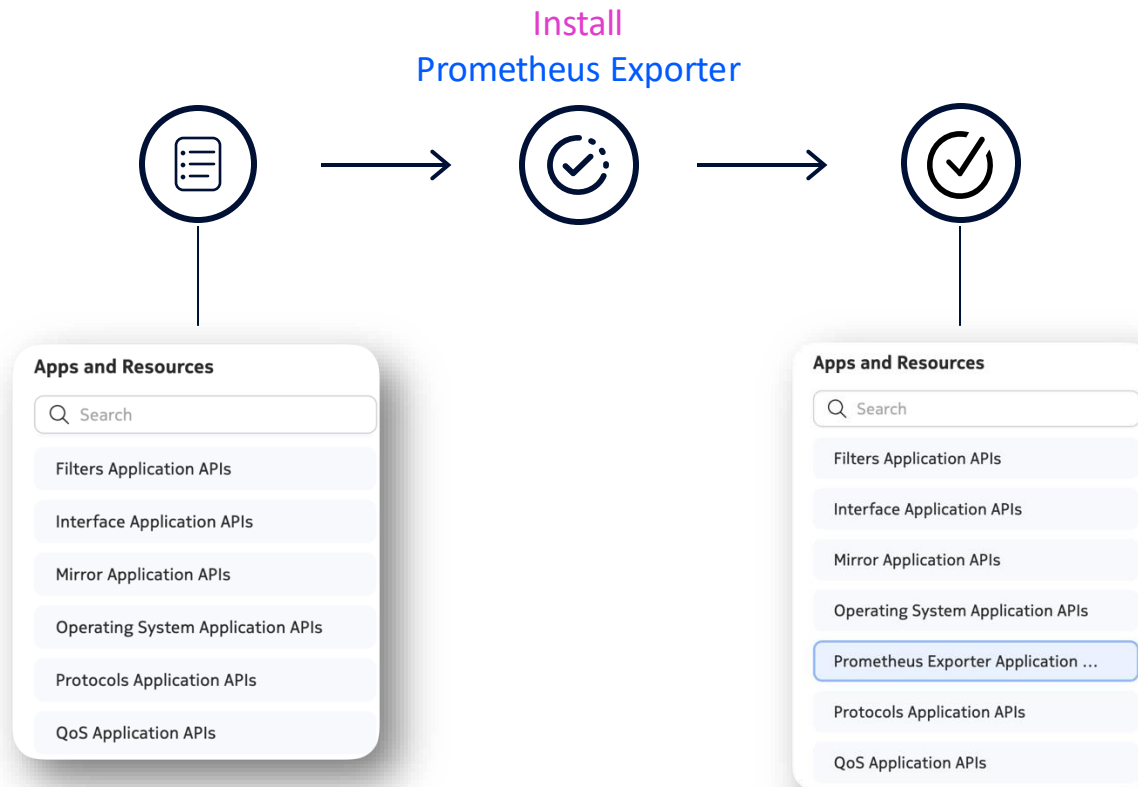
No packages published
[Publish your first package](#)

Languages

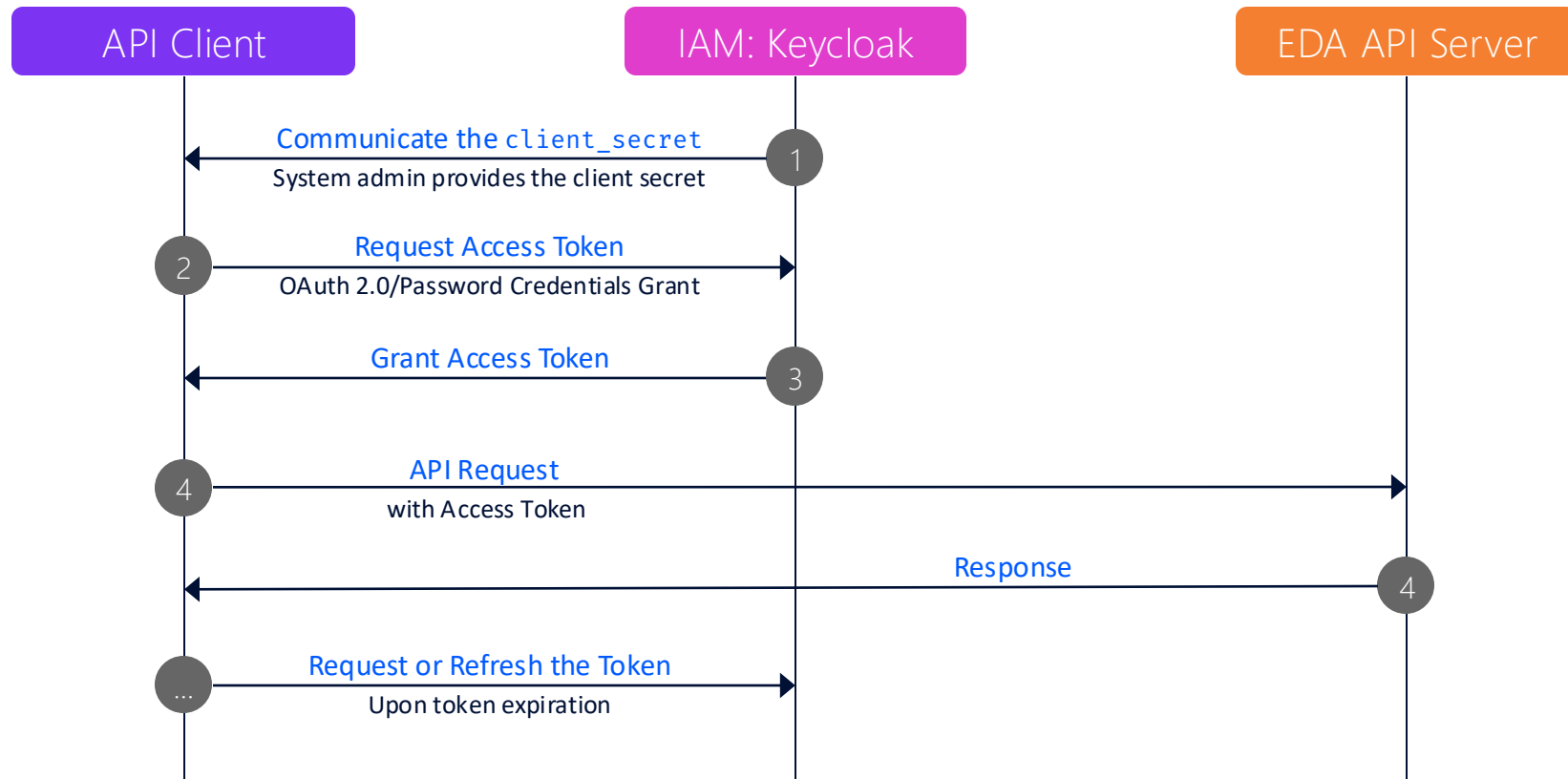
 ● Shell 100.0%

API extends with Apps

Every bit of automation comes with an API. Always.



API Client Flow



Token Request



```
curl -s https://{EDA_URL}/core/httpproxy/v1/keycloak/realms/eda/protocol/openid-connect/token \

-H 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'client_id=eda' \
--data-urlencode 'client_secret=${EDA_CLIENT_SECRET}' \

--data-urlencode 'grant_type=password' \
--data-urlencode 'scope=openid' \

--data-urlencode 'username=${USERNAME}' \
--data-urlencode 'password=${PASSWORD}'
```

Resource Creation



```
curl -s 'https://${EDA_URL}/apps/siteinfo.eda.nokia.com/v1alpha1/namespaces/eda/banners' \
--header 'Content-Type: application/json' \
--header 'Authorization: ***' \
--data '{
  "apiVersion": "siteinfo.eda.nokia.com/v1alpha1",
  "kind": "Banner",
  "metadata": {
    "name": "api-banner",
    "namespace": "eda"
  },
  "spec": {
    "loginBanner": "some banner text",
    "nodeSelector": [
      "foo=bar"
    ]
  }
}'
```

Resource Creation

```
curl -s 'https://{EDA_URL}/apps/siteinfo.eda.nokia.com/v1alpha1/namespaces/eda/banners' \
```

```
--header 'Content-Type: application/json' \
```

```
--header 'Authorization: ***' \
```

```
--data '{
```

```
  "apiVersion": "siteinfo.eda.nokia.com/v1alpha1",
```

```
  "kind": "Banner",
```

```
  "metadata": {
```

```
    "name": "api-banner",
```

```
    "namespace": "eda"
```

```
  },
```

```
  "spec": {
```

```
    "loginBanner": "some banner text",
```

```
    "nodeSelector": {
```

```
      "foo=bar"
```

```
    }
```

```
  }
```

```
}'
```

1

2

3

Resource Creation

```
curl -s 'https://${EDA_URL}/apps/siteinfo.eda.nokia.com/v1alpha1/namespaces/eda/banners' \  
--header 'Content-Type: application/json' \  
--header 'Authorization: ***' \  
--data '{  
  "apiVersion": "siteinfo.eda.nokia.com/v1alpha1",  
  "kind": "Banner",  
  "metadata": {  
    "name": "api-banner",  
    "namespace": "eda"  
  },  
  "spec": {  
    "loginBanner": "some banner text",  
    "nodeSelector": {  
      "foo=bar"  
    }  
  }  
}'
```

Resource Creation

```
curl -s 'https://${EDA_URL}/apps/siteinfo.eda.nokia.com/v1alpha1/namespaces/eda/banners' \
--header 'Content-Type: application/json' \
--header 'Authorization: ***' \
--data '{
  "apiVersion": "siteinfo.eda.nokia.com/v1alpha1",
  "kind": "Banner",
  "metadata": {
    "name": "api-banner",
    "namespace": "eda"
  },
  "spec": {
    "loginBanner": "some banner text",
    "nodeSelector": [
      "foo=bar"
    ]
  }
}'
```

Resource body is the same for API and UI

API

```
{
  "apiVersion": "siteinfo.eda.nokia.com/v1alpha1",
  "kind": "Banner",
  "metadata": {
    "name": "api-banner",
    "namespace": "eda"
  },
  "spec": {
    "loginBanner": "some banner text",
    "nodeSelector": [
      "foo=bar"
    ]
  }
}
```

UI

```
1  apiVersion: siteinfo.eda.nokia.com/v1alpha1
2  kind: Banner
3  metadata:
4    name: api-banner
5    namespace: eda
6    labels: {}
7    annotations: {}
8  spec:
9    loginBanner: some banner text
10   motd: some message of the day
11   nodeSelector:
12     - containerlab=managed
13   nodes: []
```

Pydantic EDA

Taking it one step further

- OpenAPI is a good starting point, but...
- Manually crafting JSON payloads is error-prone
- Jinja-templating path is full of runtime errors
- You want to get all the help from the IDE



github.com/eda-labs/pydantic-eda



OpenAPI spec



Pydantic



Pydantic classes
for EDA REST API

Pydantic EDA

```
import pydantic_eda.apps.siteinfo.v1alpha1.models as siteinfo

def main():
    eda = EDAClient(base_url="https://devbox.netdevops.me:9444")

    my_banner = siteinfo.Banner(
        apiVersion="siteinfo.eda.nokia.com/v1alpha1",
        kind="Banner",
        metadata=siteinfo.BannerMetadata(
            namespace=NS,
            name="my-banner",
        ),
        spec=siteinfo.BannerSpec(
            loginBanner="Pydantic EDA says Hi",
            nodes=["leaf1"],
        ),
    )

    eda.add_to_transaction_create(my_banner)
    _ = eda.commit_transaction()
```

And SDKs are coming



EDA

A modern automation



1

Extensible framework – the cookie cutter factory

2

Modern consumption – SaaS, app stores

3

Multivendor by design, in a modular fashion

4

Built for the modern tooling ecosystem

5

Kubernetes-style reconciliation, with nuances for infrastructure

6

State as a first-class citizen in the automation stack

7

Unbounded scale – ultra low to monumentally large

8

Stream everything – no exceptions

9

Use-case driven visualizations – extensibility in the UI

10

Built to enable DevOps principles – CI/CD

11

Programmable interfaces all around.

12

Integrated with the application world – on prem and off

You are ready for the afternoon!

- The hackathon is what you've been waiting for
- A full afternoon of hands-on challenges and learning
- All experience levels and all interests
- You will need
 - Yourself
 - An open mind and a willingness to throw yourself into things
 - Your laptop (With either an SSH client and/or a web browser)

14:00-18:00hrs



Support each other

Achieve together

See you at 14:00hrs



Copyright and confidentiality

The contents of this document are proprietary and confidential property of Nokia. This document is provided subject to confidentiality obligations of the applicable agreement(s).

This document is intended for use by Nokia's customers and collaborators only for the purpose for which this document is submitted by Nokia. No part of this document may be reproduced or made available to the public or to any third party in any form or means without the prior written permission of Nokia. This document is to be used by properly trained professional personnel. Any use of the contents in this document is limited strictly to the use(s) specifically created in the applicable agreement(s) under which the document is submitted. The user of this document may voluntarily provide suggestions, comments or other feedback to Nokia in respect of the contents of this document ("Feedback"). Such Feedback may be used in Nokia products and

related specifications or other documentation. Accordingly, if the user of this document gives Nokia Feedback on the contents of this document, Nokia may freely use, disclose, reproduce, license, distribute and otherwise commercialize the feedback in any Nokia product, technology, service, specification or other documentation.

Nokia operates a policy of ongoing development. Nokia reserves the right to make changes and improvements to any of the products and/or services described in this document or withdraw this document at any time without prior notice.

The contents of this document are provided "as is". Except as required by applicable law, no warranties of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose, are made in relation to the accuracy, reliability or contents

of this document. NOKIA SHALL NOT BE RESPONSIBLE IN ANY EVENT FOR ERRORS IN THIS DOCUMENT or for any loss of data or income or any special, incidental, consequential, indirect or direct damages howsoever caused, that might arise from the use of this document or any contents of this document.

This document and the product(s) it describes are protected by copyright according to the applicable laws.

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.

Copyright and confidentiality

The contents of this document are proprietary and confidential property of Nokia. This document is provided subject to confidentiality obligations of the applicable agreement(s).

This document is intended for use by Nokia's customers and collaborators only for the purpose for which this document is submitted by Nokia. No part of this document may be reproduced or made available to the public or to any third party in any form or means without the prior written permission of Nokia. This document is to be used by properly trained professional personnel. Any use of the contents in this document is limited strictly to the use(s) specifically created in the applicable agreement(s) under which the document is submitted. The user of this document may voluntarily provide suggestions, comments or other feedback to Nokia in respect of the contents of this document ("Feedback"). Such Feedback may be used in Nokia products and

related specifications or other documentation. Accordingly, if the user of this document gives Nokia Feedback on the contents of this document, Nokia may freely use, disclose, reproduce, license, distribute and otherwise commercialize the feedback in any Nokia product, technology, service, specification or other documentation.

Nokia operates a policy of ongoing development. Nokia reserves the right to make changes and improvements to any of the products and/or services described in this document or withdraw this document at any time without prior notice.

The contents of this document are provided "as is". Except as required by applicable law, no warranties of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose, are made in relation to the accuracy, reliability or contents

of this document. NOKIA SHALL NOT BE RESPONSIBLE IN ANY EVENT FOR ERRORS IN THIS DOCUMENT or for any loss of data or income or any special, incidental, consequential, indirect or direct damages howsoever caused, that might arise from the use of this document or any contents of this document.

This document and the product(s) it describes are protected by copyright according to the applicable laws.

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.