



The Three Stages of Automation and Orchestration

Stage 2—Service Abstraction

February 7, 2018

Today's Presenters



John Malzahn

Senior Manager, Cloud
and Virtualization
Solutions Marketing

Cisco Systems



Carl Moberg

Technology Director,
Cloud Solutions and
Platform Group

Cisco Systems

Today's Agenda

© 2017 Cisco and/or its affiliates. All rights reserved.

- 1 Introducing the Three Stages of Automation
- 2 Deep Dive on Stage #2
Service Abstraction
- 3 Using NSO for Service Abstraction
- 4 Demo Time!
- 5 Wrap-up and Q&A

Key Market Trend Observations

Execution at the speed of software



- Networks provides well-known utility abstractions
- Agility, DevOps, NFV, SDN drives new expectations

Changing customer behavior and new expectations



- Everything on demand
- New services with a press of a button

Rapidly changing business models



- Cloud services, virtualization, programmable networks
- New value chains including OTT Co-opetition

All of this requires successful, flexible automation.
But complexity has destroyed many automation initiatives.

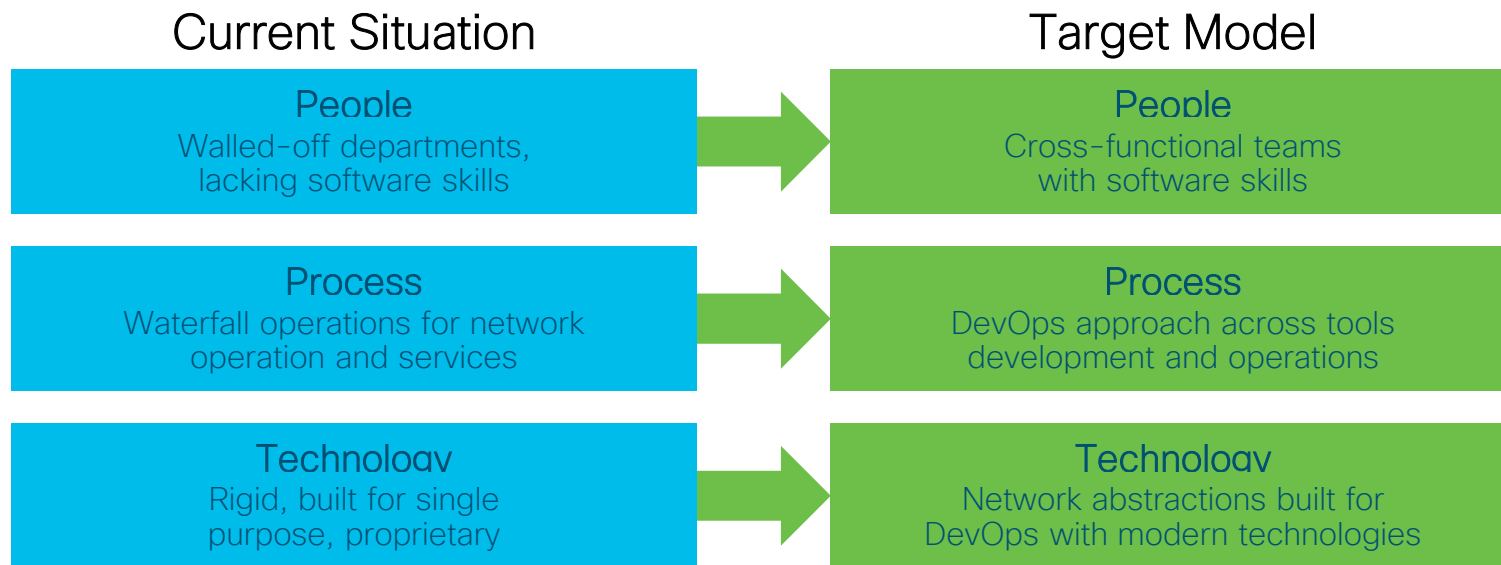
Departmental Pain Points

Network Engineer “Automation”	Ops and Provisioning Team “Customer Experience”	Service Developers “Time-to-Market”
<p>Day-to-day management of rapidly growing, complex networks</p> <p>Challenges</p> <ul style="list-style-type: none">• Error-prone manual tasks• Growing backlog• Virtualization is coming	<p>Provisions services and manages service quality in networks</p> <p>Challenges</p> <ul style="list-style-type: none">• No service insight• Lack of automation• Quality issues in delivery	<p>Develops new network services on demand</p> <p>Challenges</p> <ul style="list-style-type: none">• Implementation time• Cost of change• Lack of tooling

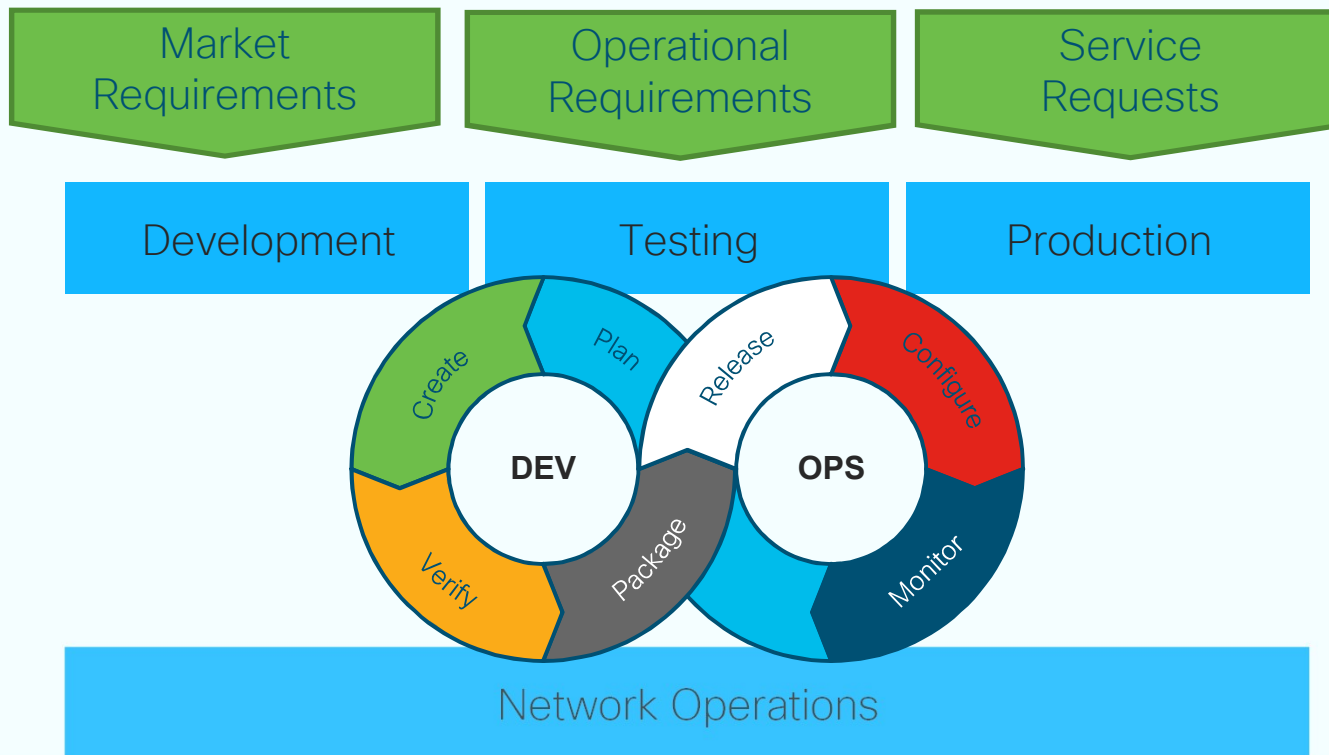
Transition Towards Automation

Network Engineer “Automation”	Ops and Provisioning Team “Customer Experience”	Service Developers “Time-to-Market”
<p>Day-to-day management of rapidly growing, complex networks</p> <p>Challenges</p> <ul style="list-style-type: none">• Error-prone manual tasks• Growing backlog• Virtualization is coming	<p>Provisions services and manages service quality in networks</p> <p>Challenges</p> <ul style="list-style-type: none">• No service insight• Lack of automation• Quality issues in delivery	<p>Develops new network services on demand</p> <p>Challenges</p> <ul style="list-style-type: none">• Implementation time• Cost of change• Lack of tooling
<p>Network API Utilize a single interface to all network devices</p> <p>1</p>	<p>Service Abstraction Leverage one central API for all services</p> <p>2</p>	<p>Transformation Develop your own services</p> <p>3</p>

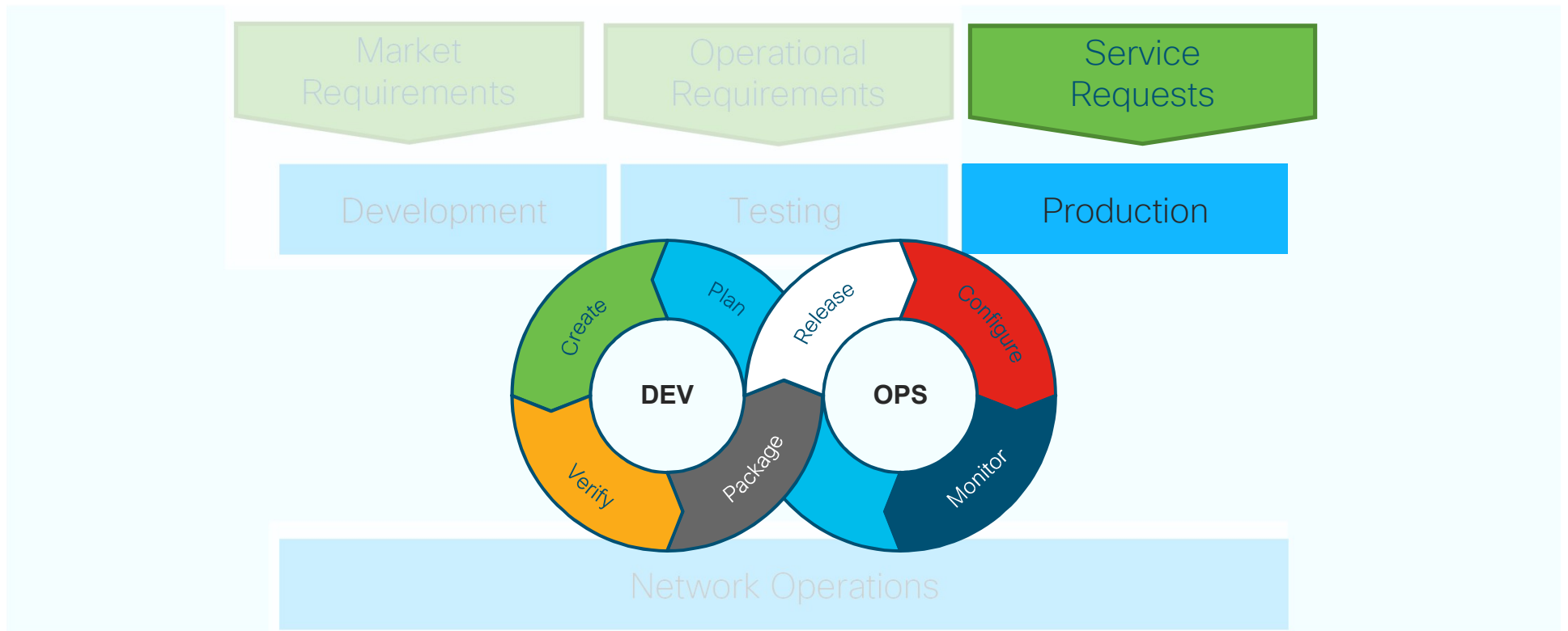
Change Requirements



Devops Virtual Cycle



Devops Virtual Cycle (Today's Focus)



Poll Question #1

Have you, or are you planning to provide an API for services to your network

Transition Towards Automation

Network Engineer “Automation”	Ops and Provisioning Team “Customer Experience”	Service Developers “Time-to-Market”
<p>Day-to-day management of rapidly growing, complex networks</p> <p>Challenges</p> <ul style="list-style-type: none">• Error-prone manual tasks• Growing backlog• Virtualization is coming	<p>Provisions services and manages service quality in networks</p> <p>Challenges</p> <ul style="list-style-type: none">• Lack of automation• Quality issues in delivery• No service insight	<p>Develops new network services on demand</p> <p>Challenges</p> <ul style="list-style-type: none">• Implementation time• Cost of change• Lack of tooling
<p>Network API Utilize a single interface to all network devices</p> <p>1</p>	<p>Service Abstraction Leverage one central API for all services</p> <p>2</p>	<p>Transformation Develop your own services</p> <p>3</p>

Challenge Mapping

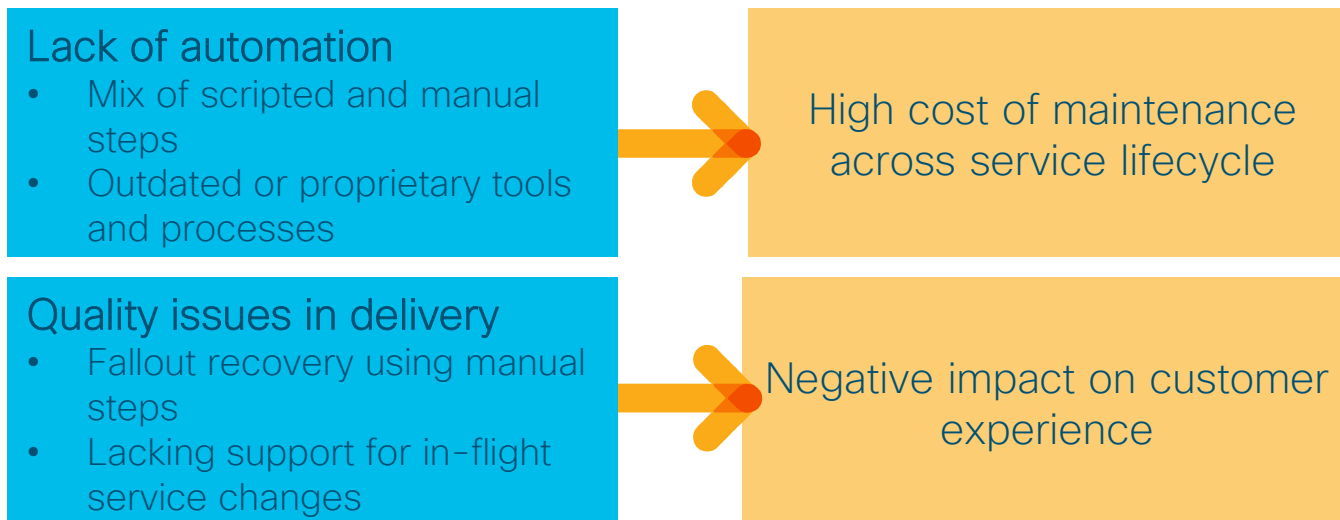
Lack of automation

- Mix of scripted and manual steps
- Outdated or proprietary tools and processes

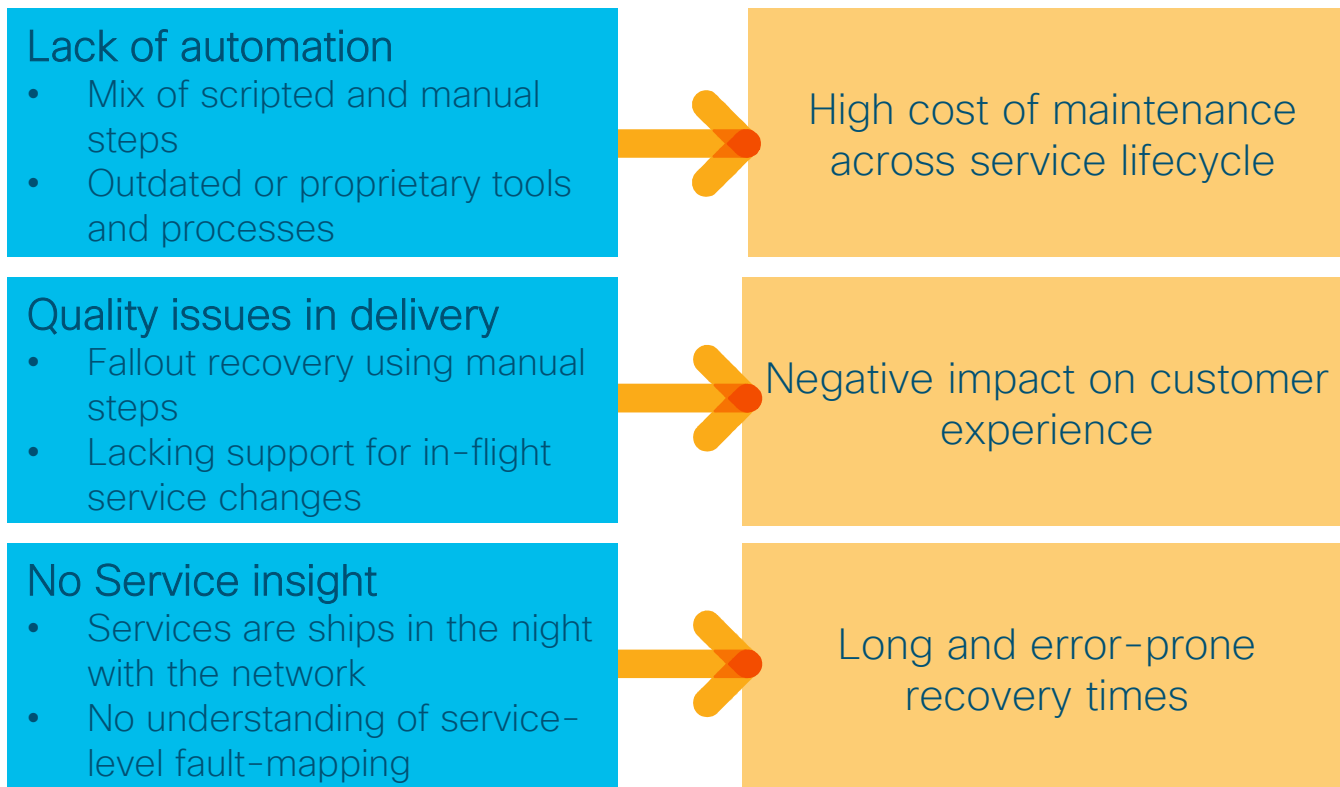


High cost of maintenance for services

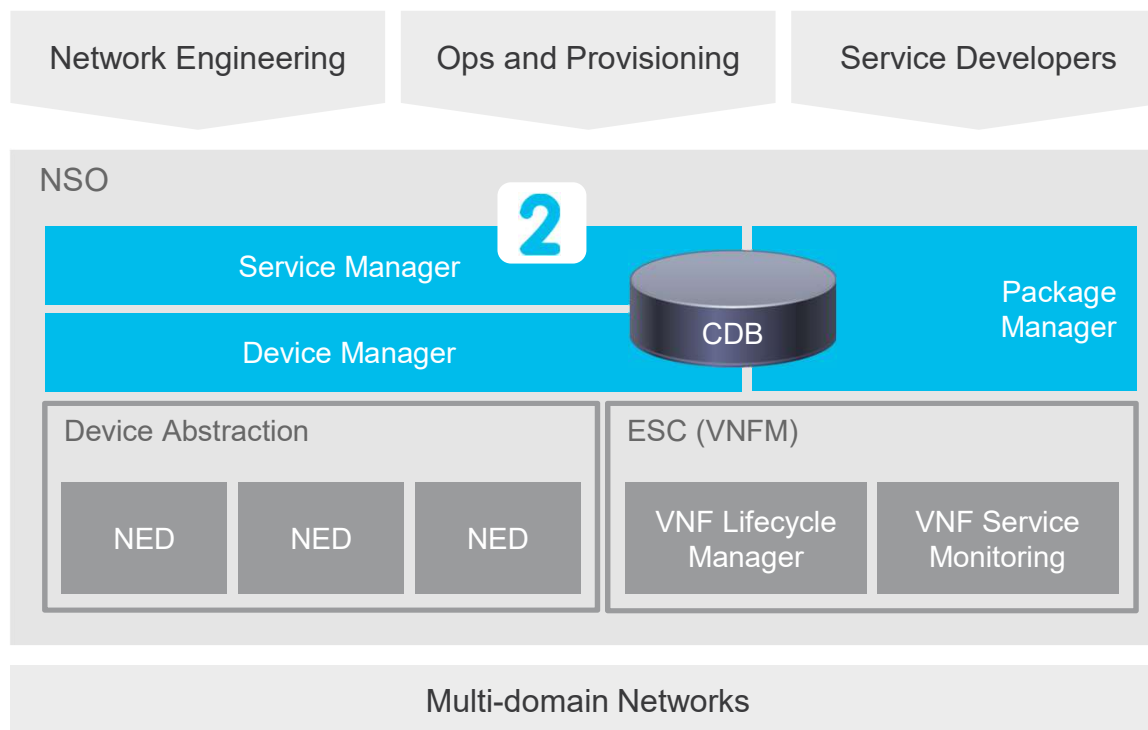
Challenge Mapping



Challenge Mapping



Quick System Overview



- Model-driven end-to-end service lifecycle and customer experience in focus
- Seamless integration with existing and future OSS/BSS environment
- Loosely-coupled and modular architecture leveraging open APIs and standard protocols
- Orchestration across multi-domain and multi-layer for centralized policy and services across entire network

Feature Mapping #1

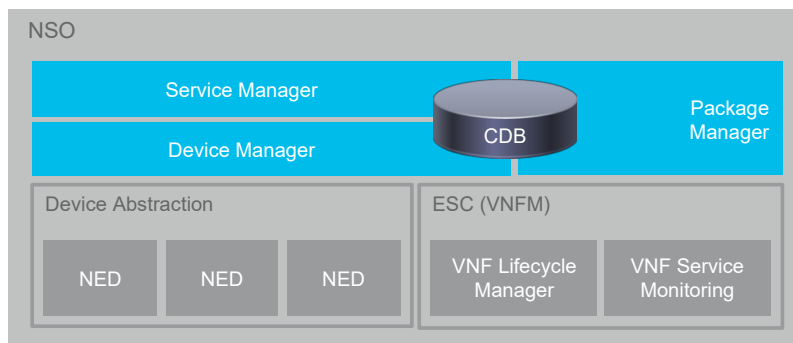
Service Models and APIs

The Challenge

- We need:
 - A formal and well understood means of understanding and working with the service layer...
 - allowing for use of modern, mainstream tools and practices
- On:
 - Brownfield networks
 - Across place in network, vendor, device type and protocol

Expose the services layer using formal language and approaches that match the technology choices of modern development teams

Model-driven Means Choice!



Protocols and language-bindings are used to lifecycle the service instances

The *service models* define the valid content of *service instances* in CDB

Pick the Tools that Work for You

```
with m.start_read_trans() as t:  
    rt = ncs.maagic.get_root(t)  
    for svc in rt.ncs__services.service:
```

```
GET /api/config/services HTTP/1.1
```



```
netconf-console --get-config \  
-x '/services'
```



```
1 module: eline  
2   augment /ncs:services:  
3     +--rw eline* [name]  
4       +--rw name string  
5       +--rw serviceinstance? uint16  
6       +--rw a-side  
7         +--rw device? -> /ncs:devices/device/name  
8         +--rw GigabitEthernet? string  
9         +--rw remote-ip? inet:ipv4-address  
10      +--rw z-side  
11        +--rw device? -> /ncs:devices/device/name  
12        +--rw GigabitEthernet? string  
13        +--rw remote-ip? inet:ipv4-address
```

Create

- Easy
- Given a set of service-level inputs, provide a known and valid output to network
- May require some additional resource collection to fulfill the configuration set



Update

- Challenging
- Allow arbitrary changes to the network service
- May require collecting or handing back resources to fulfill configuration set



Delete

- Hard
- Delete any given instance of a service and clean up the resources
- May require reference counting for shared resources

The ability to dry-run all operations is key for trust

Feature Mapping #2

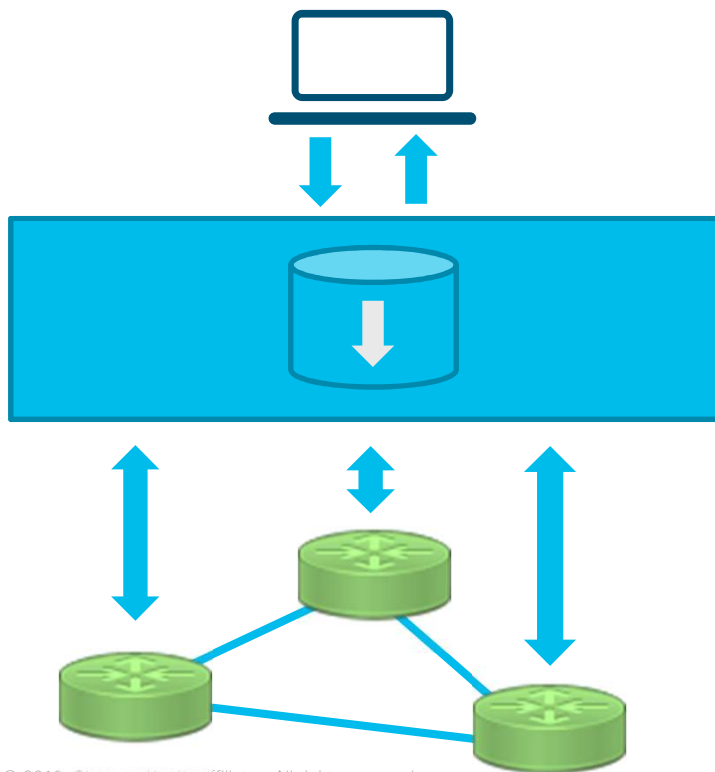
Fail-safety with Transactions

The Challenge

- We need:
 - A comfortable programming environment on the northbound interface...
 - implementing well known data manipulation abstractions (create, read, update, delete)
- On:
 - Brownfield networks
 - Across place in network, vendor, device type and protocol

Make the network look like a single database with all service instances represented in a single, easy-to manage data tree

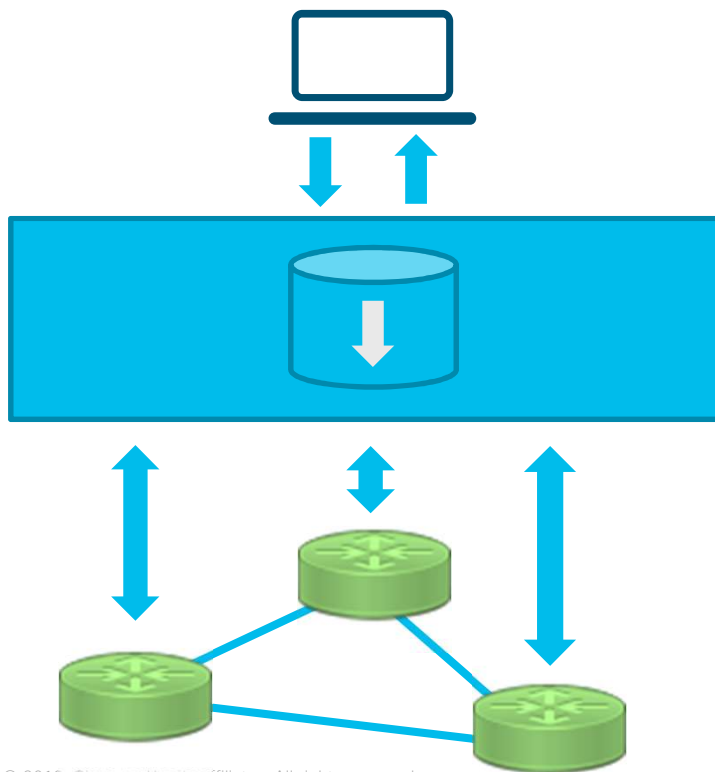
Fully Synchronous



© 2018 Cisco and/or its affiliates. All rights reserved.

1. Client performs operation on service instance
2. Orchestrator resolves dependencies, resource situation, creates per-device intent
3. Orchestrator performs intent-driving operations to affected resources
4. Orchestrator returns result

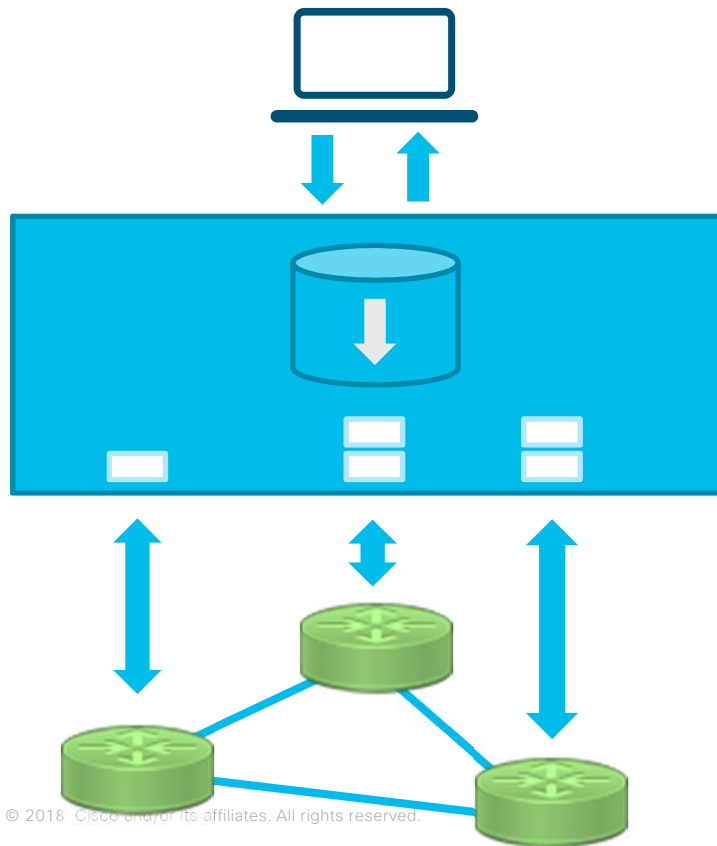
Fully Synchronous



© 2018 Cisco and/or its affiliates. All rights reserved.

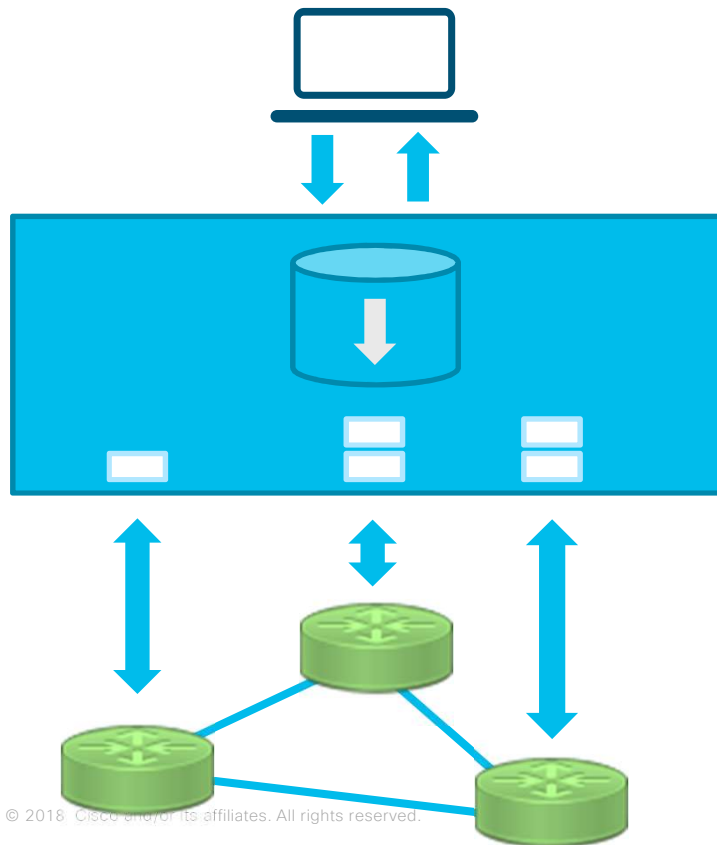
1. Client performs operation on service instance
2. Orchestrator resolves dependencies, resource situation, creates per-device intent Lock
3. Orchestrator performs intent-driving operations to affected resources
4. Orchestrator returns result

Eventually Consistent (*Commit Queues*)



1. Client performs operation on service instance
2. Orchestrator confirms the reception of the operation
3. Orchestrator:
 - Creates per-resource intent
 - Puts queue items in per-resource queues
4. Orchestrator works through queues and updates queue item status
5. Client polls or gets notified on queue item completion

Eventually Consistent (*Commit Queues*)



© 2018 Cisco and/or its affiliates. All rights reserved.

1. Client performs operation on service instance
2. Orchestrator confirms the reception of the operation
3. Orchestrator:
 - Creates per-resource intent
 - Puts queue items in per-resource queues
4. Orchestrator works through queues and updates queue item status
5. Client polls or gets notified on queue item completion

Lock

Two Modes for Network Programmability

Fully Synchronous

- Big Pro
 - Very comfortable for the developer, no fallout coding
- Somewhat Con
 - Leverages global locks, may lead to contenting

Eventually Consistent

- Big Pro
 - Parallel execution, minimal to no lock contention
- Somewhat Con
 - Potentially tricky fallout situations and client state management

Need to consider usage patterns, service types, frequency of change, size of network, characteristics of clients, etc.

Feature Mapping #3

Service Insight

The Challenge

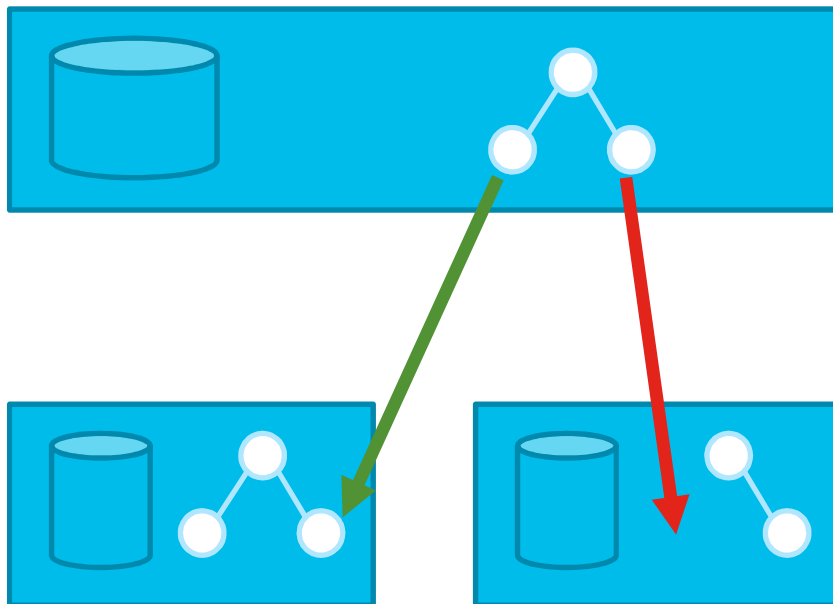
- We need:
 - A way to understand the relationship between the service lifecycle and the resources in the network (configuration, runtime state)...
 - Using easily accessible constructs (references) both up- and down the stack
- On:
 - Brownfield networks
 - Across place in network, vendor, device type and protocol

Provide referential integrity across services and resources in a fashion that allows actionable operations (planning, what-if, fallout, capacity management)

Three Common Service Operations Issues

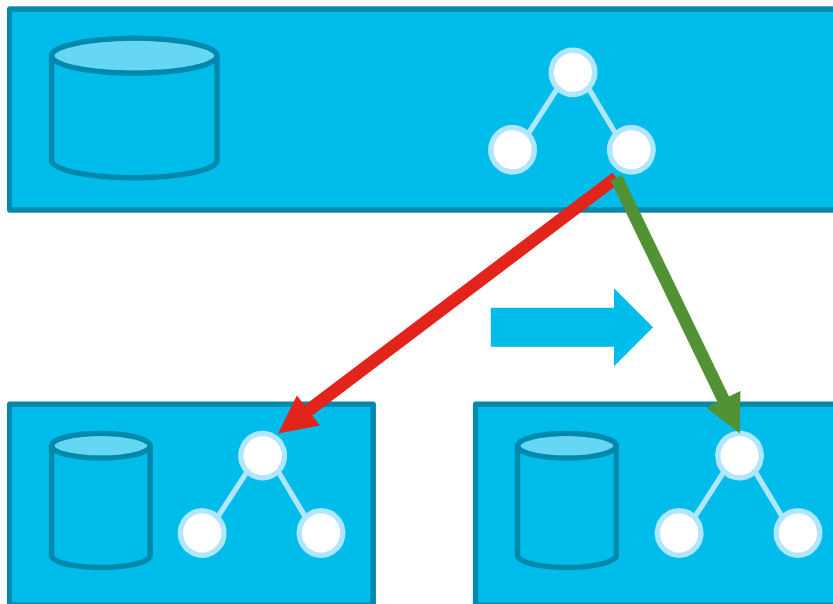
- Recovering from out-of-band changes
- Relocating services in an upgrade/migration scenario
- Create resource views for service-oriented fault management

Recover From Out-of-band Changes



- Referential integrity is broken between service layer and device layer
- Redeploy service instance to reproduce resource intent
- Produce diff between intent and actual
- Remediate using diff

Relocate Service for Upgrade or Migration

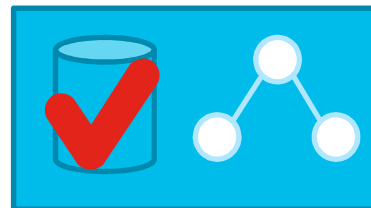
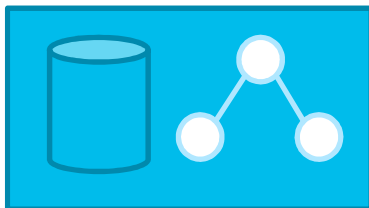


- Service resources need to be moved across devices
- Redeploy the service with the new device input
- Resulting diff will include:
 - Configuration to be removed from evacuated device
 - Configuration to be added to arriving device

Resource Views for Fault Management



- Device is out of order and not able to support service
- Service operations team queries service instances for references to lost device
- Remediation activities may include moving service, or replacing device



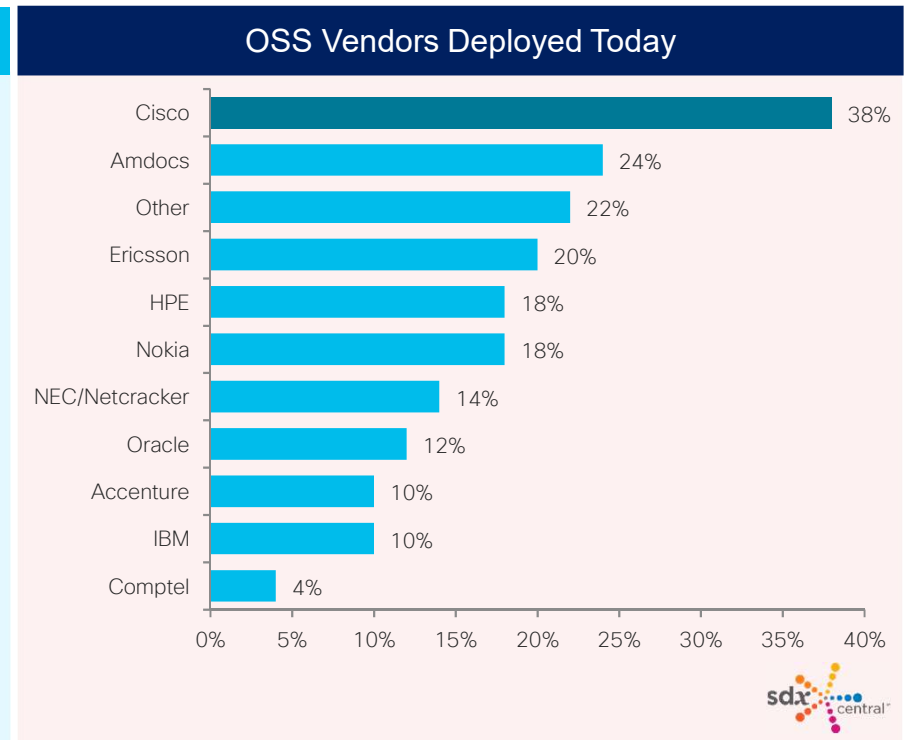
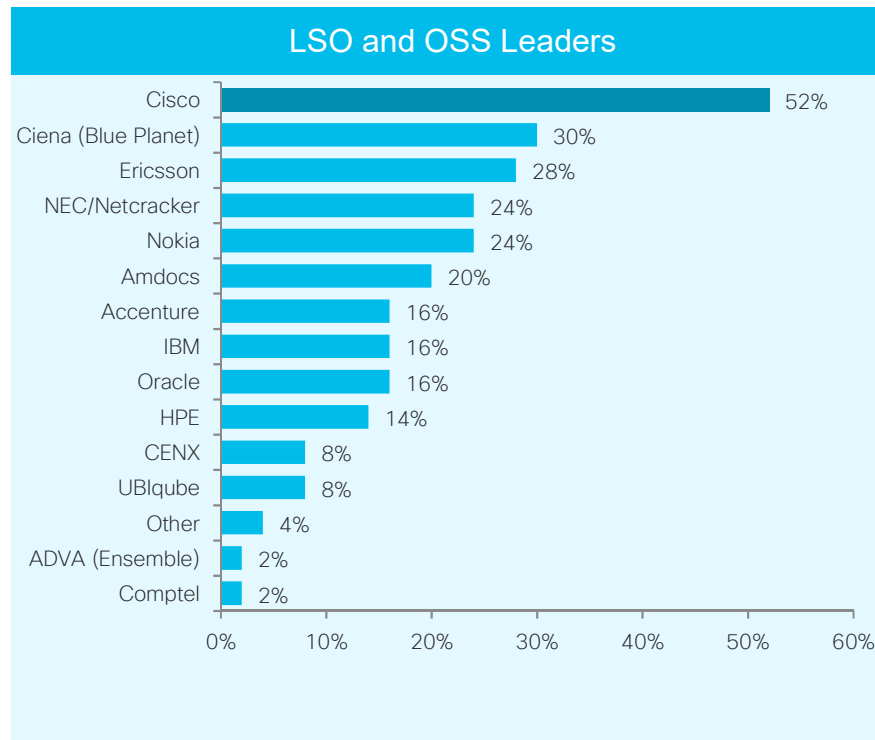
Poll Question #2

How do you currently or plan to handle your API development?

Demo Time

Operators Voted

Cisco Leads Industry In Lifecycle Service Orchestration






Source: SDxCentral 2017 Next-Gen OSS and the Rise of LSO Report

What You Gain with Cisco Network Services Orchestrator, Enabled by Tail-f

- Agility throughout service lifecycle
 - Strict YANG model-driven solution
 - Auto-rendered business logic results in 90% less code
 - Effortlessly re-deployment of updated service and device models
 - DevOps for differentiation
- Full automation
- Robust and proven in tier-1 deployments
- Industry's broadest multivendor support
- Relevant in today's and tomorrow's networks



Next Webinar in the *Three Stages* Series

Network Engineer “Automation”	Ops and Provisioning Team “Customer Experience”	Service Developers “Time-to-Market”
<p>Day-to-day management of rapidly growing, complex networks</p> <p>Challenges</p> <ul style="list-style-type: none">• Error-prone manual tasks• Growing backlog• Virtualization is coming	<p>Provisions services and manages service quality in networks</p> <p>Challenges</p> <ul style="list-style-type: none">• No service insight• Lack of automation• Quality issues in delivery	<p>Develops new network services on demand</p> <p>Challenges</p> <ul style="list-style-type: none">• Implementation time• Cost of change• Lack of tooling
January 10 	February 7 	March 7 

NSO DevNet – Key Highlights

The one place to use for sharing, finding and collaborating on NSO public knowledge!



Light start
through
DevNet
content
page and
Learning-
Labs



Constant
news and
updates to
help you
keep up to
date



Large
searchable
content pool



Cisco
customers,
partners
and
employees
all have
access



Got a
question,
ask! We will
help ensure
a fast
response



Easy to
share and
find public
content



Code
sharing
through
public
GitHub

developer.cisco.com/site/nso

For More Information

Visit:

www.cisco.com/go/nso

and contact your Cisco account representatives



Visit Us at Mobile World Congress – Feb 26–Mar 1
Cisco Booth Hall 3, Space 3E30

Questions?



