



Service
Provider Summit
at **Cisco** *live!*

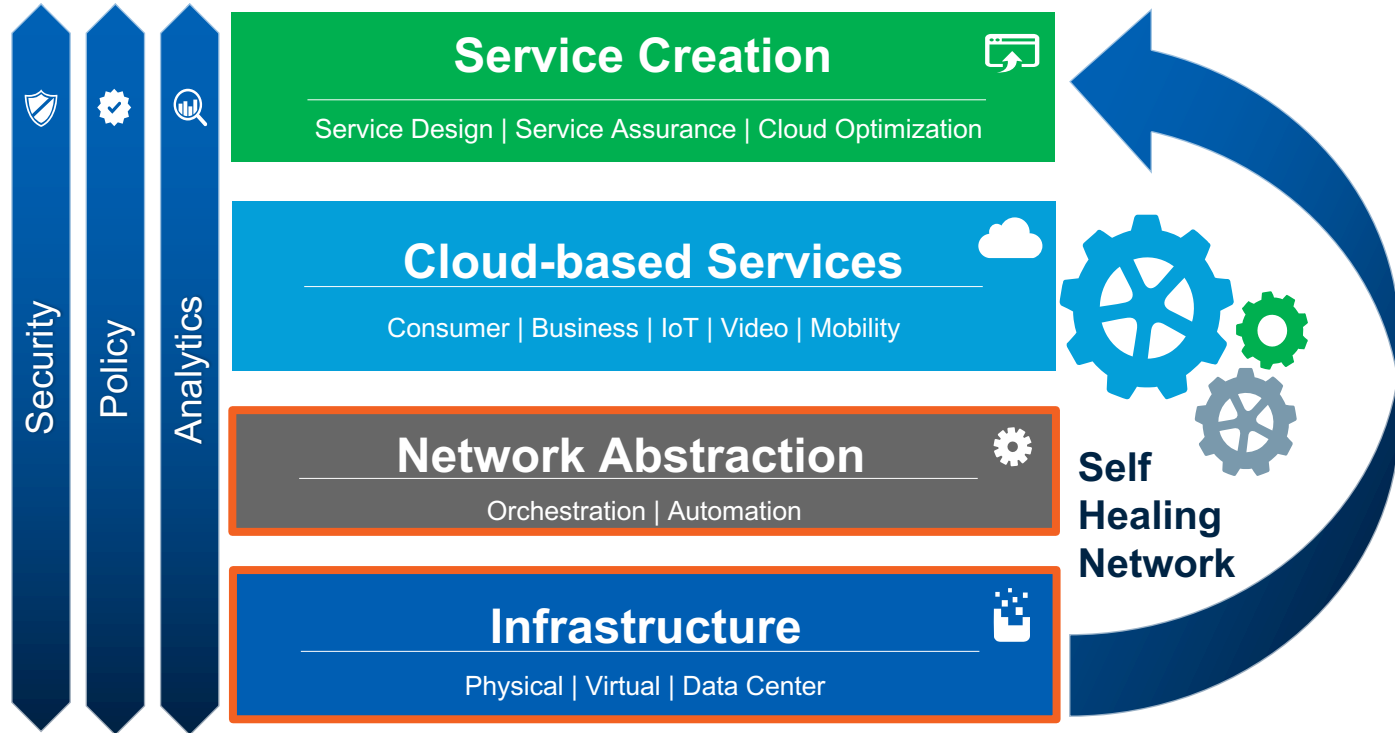
Your Time Is Now

Technologies for the future of Network Insight and Automation

Richard Wade (ricwade@cisco.com)

Technical Leader, Asia-Pacific Infrastructure Programmability

This Session's Context



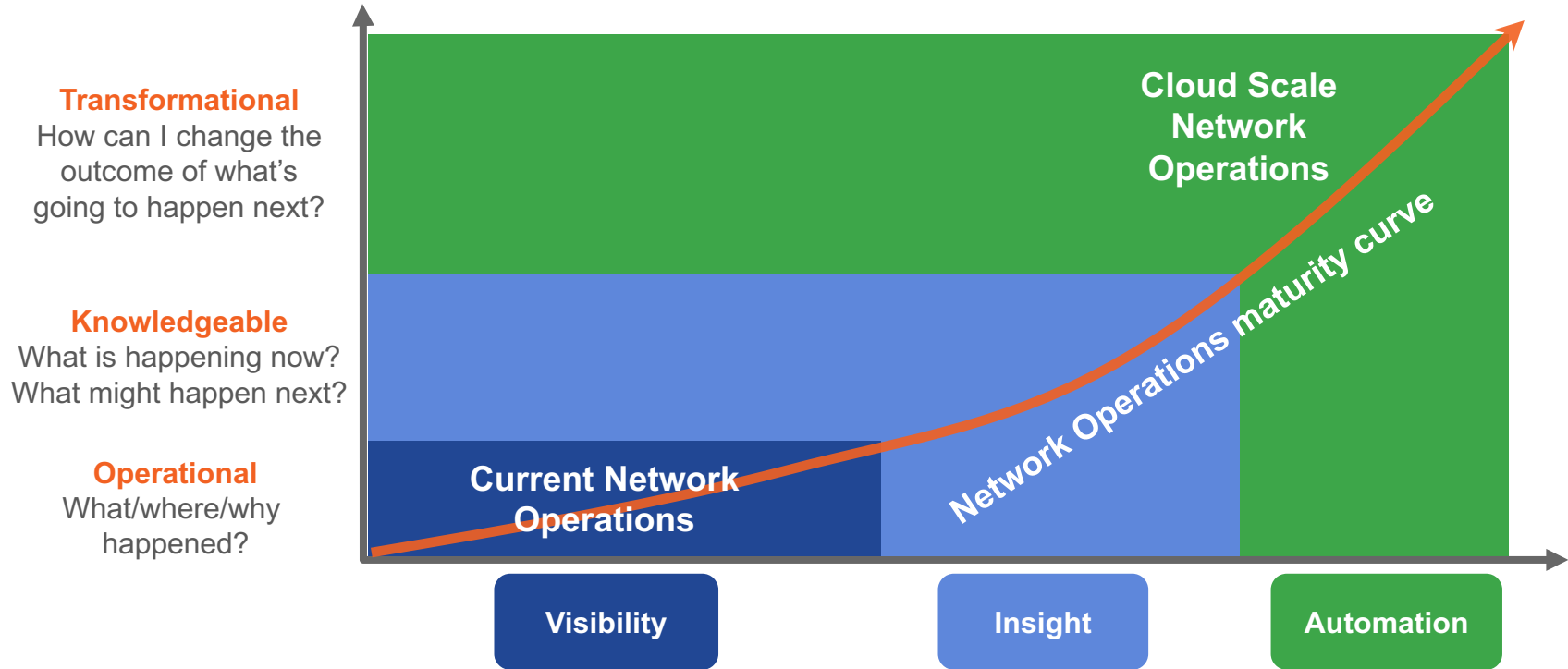
“What you need to know right now about infrastructure big data, analytics, and network automation.”

Topics

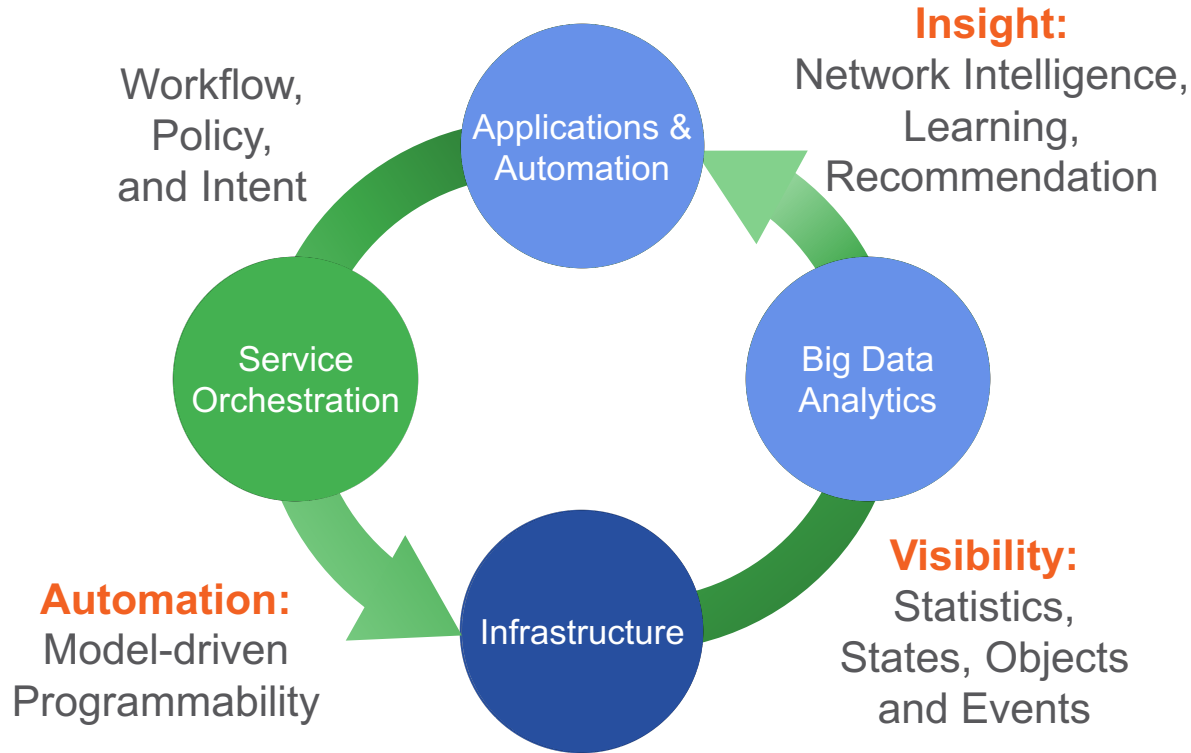
- The Relationship between Visibility, Insight & Automation
- Key Concepts and Technologies
- Streaming Telemetry, Infrastructure Big Data & Analytics
- Innovations in Network Automation
- Bringing it Together

The Relationship between Visibility, Insight & Automation

Objective: Evolving Network Operations

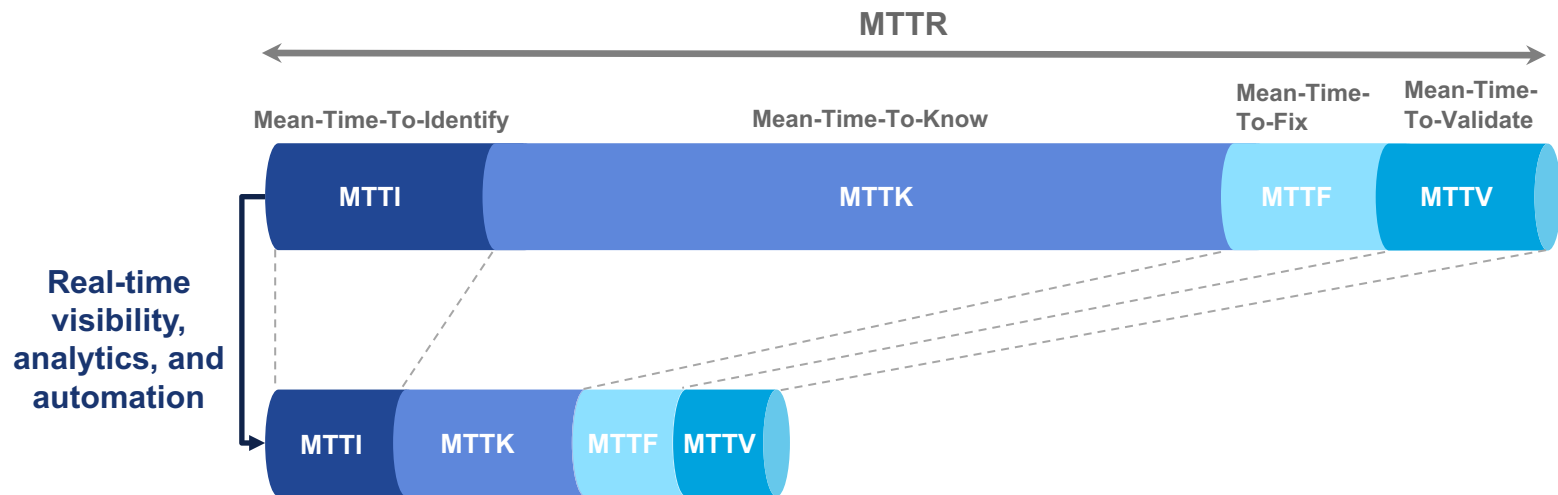


Cisco's Approach to Network Automation



Objective: Improving the Customer Experience

Mean-Time-To-Repair is one of the main **KPIs** impacting **customer experience**



Key Concepts and Technologies in Three Steps

Step 1: Streaming Telemetry

Where Data is Created



Sensing and
Measurement

Real
time

Removing Limitations and Complexity

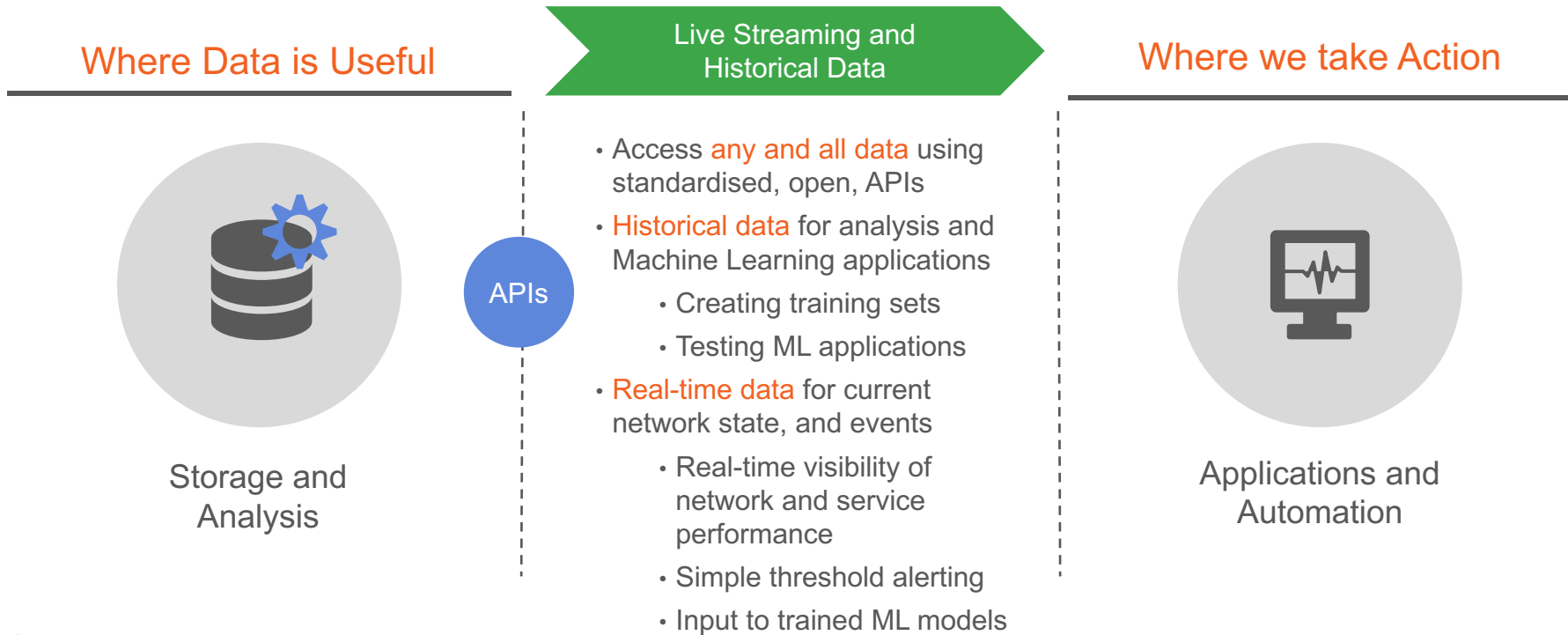
- **Push** paradigm
- One consistent way to produce Statistics, operating state and events at all layers
- High performance: <10 sec
- Multiple encodings and transport
- Available **NOW** in
 - IOS-XR
 - NX-OS

Where Data is Useful

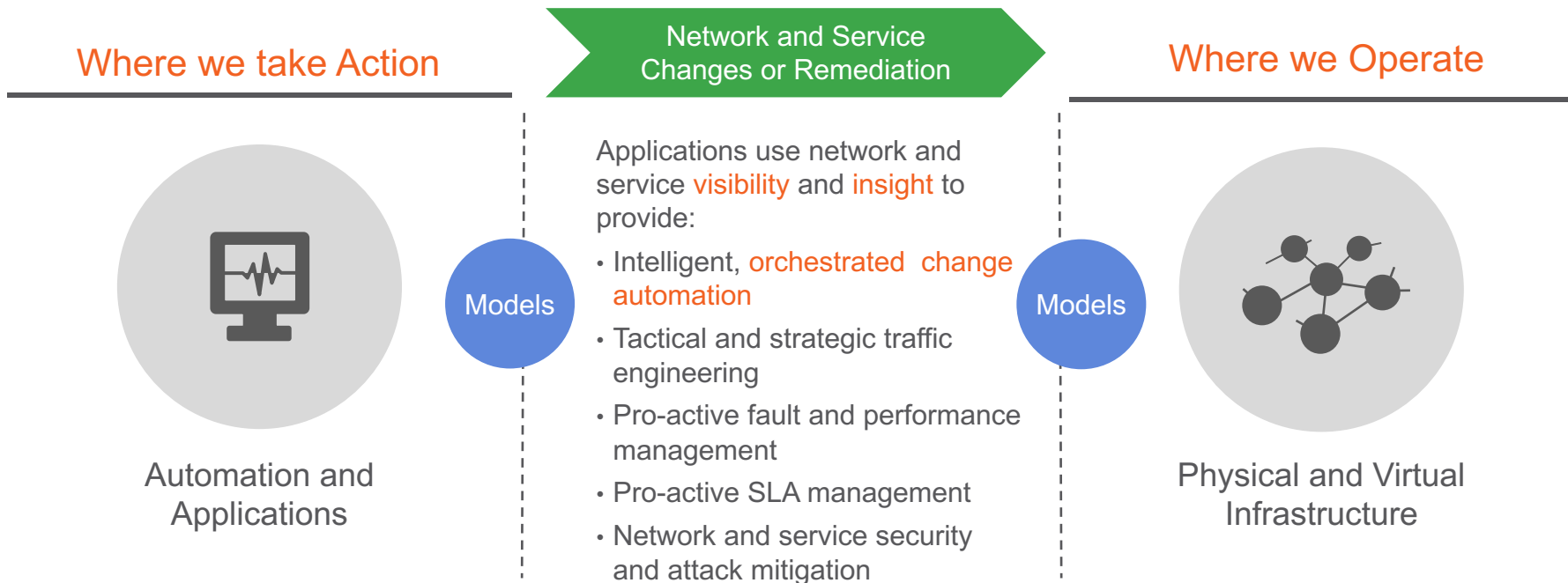


Storage and
Analysis

Step 2: Infrastructure Big Data & Analytics



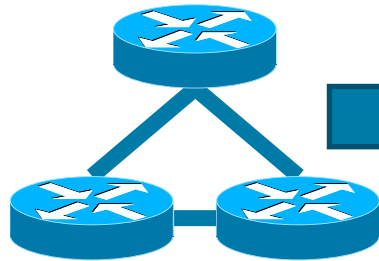
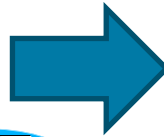
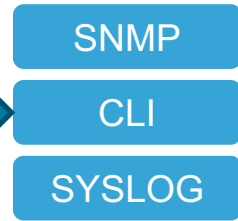
Step 3: Network Automation and Applications



Streaming Telemetry, Infrastructure Big Data, and Analytics

Traditional Monitoring is Showing its Age

Where Data is Created



Static Data

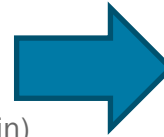
Low volume (MB)

Low velocity (15-30min)

Low variety (MIBs, CLI)

- ❗ Incomplete
- ❗ Scale Issues
- ❗ Unstructured
- ❗ Non real-time

Where Data is Useful



SNMP server

CLI repository

SYSLOG server

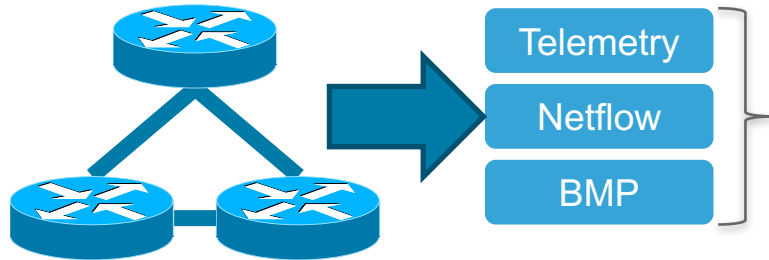
Strong burden on back-end

Normalise and correlate encodings,
timestamps, data models

- ❗ Poor value extraction

Streaming Telemetry is a Game Changer

Where Data is Created



Streaming Data

High volume (GB-TB)

High velocity (10-90sec)

High variety (flows, topology, routes, logs)

Where Data is Useful



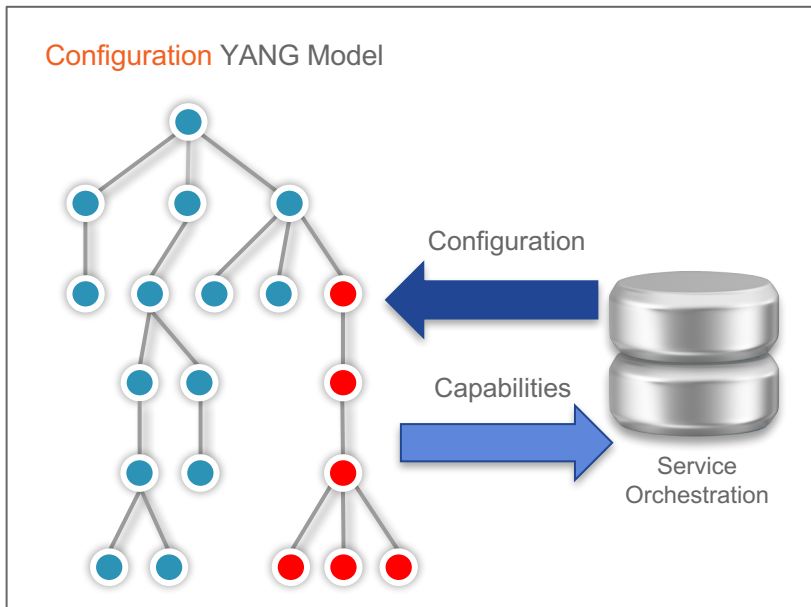
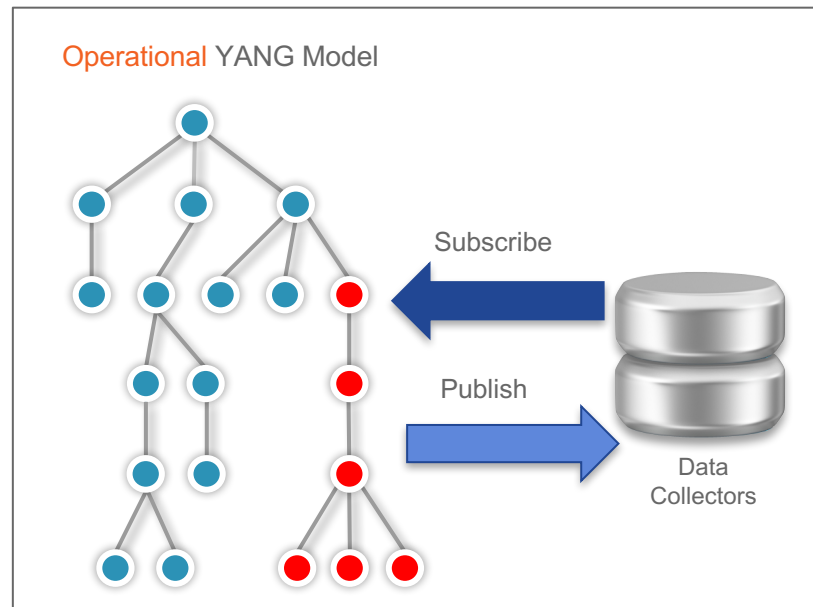
Big-Data Architecture

- ✓ Near Real-time
- ✓ Push not pull
- ✓ Structured (Analytics Ready)
- ✓ Consistent access to state and events:
Driven by data models

- ✓ Superior insight

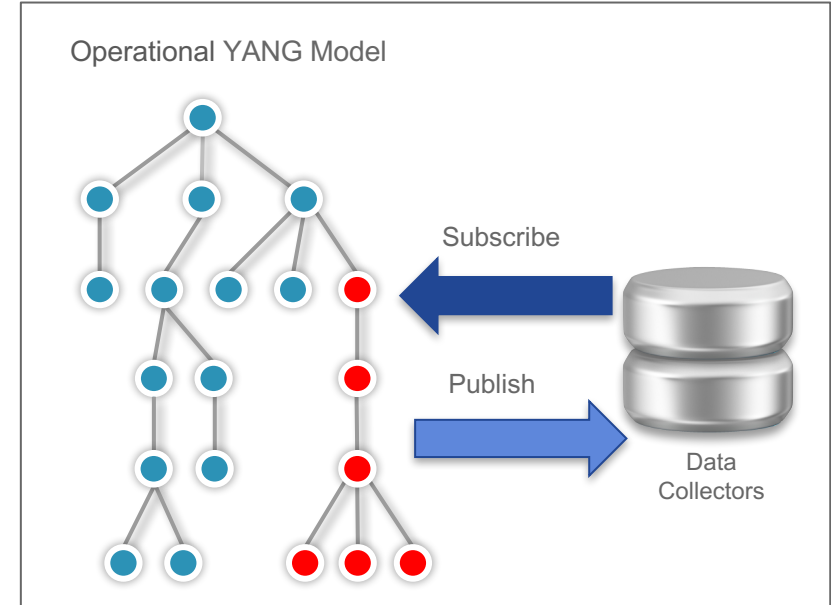
Key Technology: Models

- Consistent way to both:
 - **configure infrastructure**, and
 - access statistics, **operational state**, and events at all layers



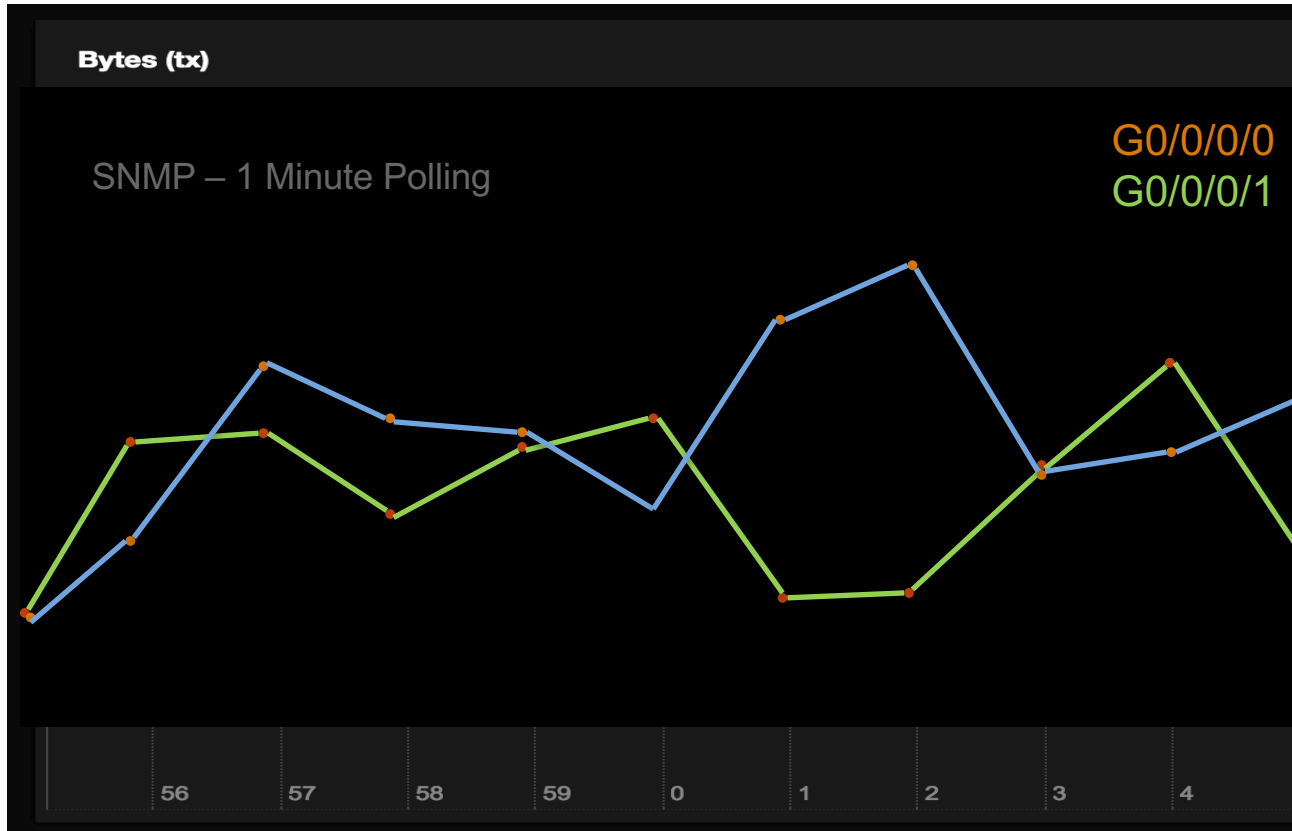
The Role of Models in Telemetry

- Consistent way to access **statistics**, **operational state** and **events** at all layers
- Models describe the metrics we can stream
- **Publish-Subscribe** architecture
- Subscribe to receive metrics from one or more models
- **High Performance**: 10 sec cadence
- Enabling closed loop automation



Telemetry vs SNMP: Link Bundle Monitoring

Is the
bundle
balanced?



Near Real-Time Insight on Hundreds of Metrics

Network Health / Troubleshooting

- Environmental health
- IP Routing health / stability
- Routing Black Hole detection
- Packet loss trending
- ECMP / LAG balance

Optimisation, SLA

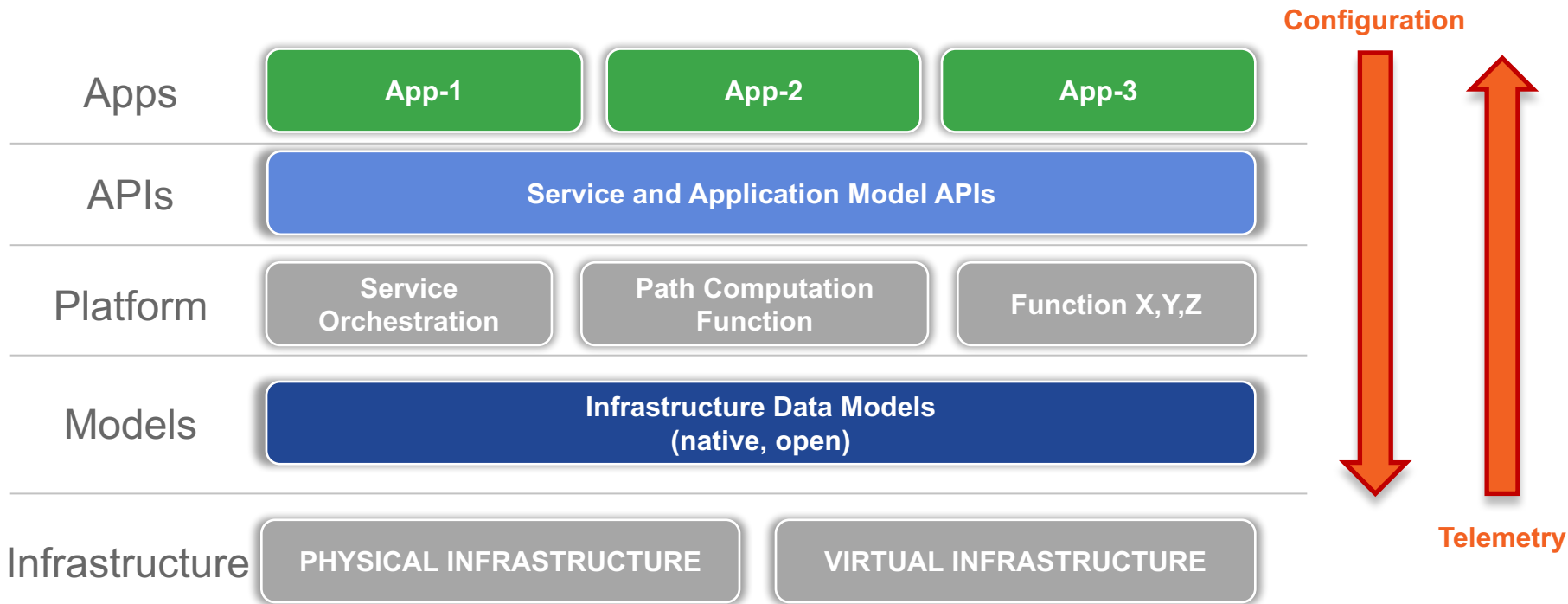
- Tactical traffic engineering
- QoS / Policy

Security / Audit / Compliance

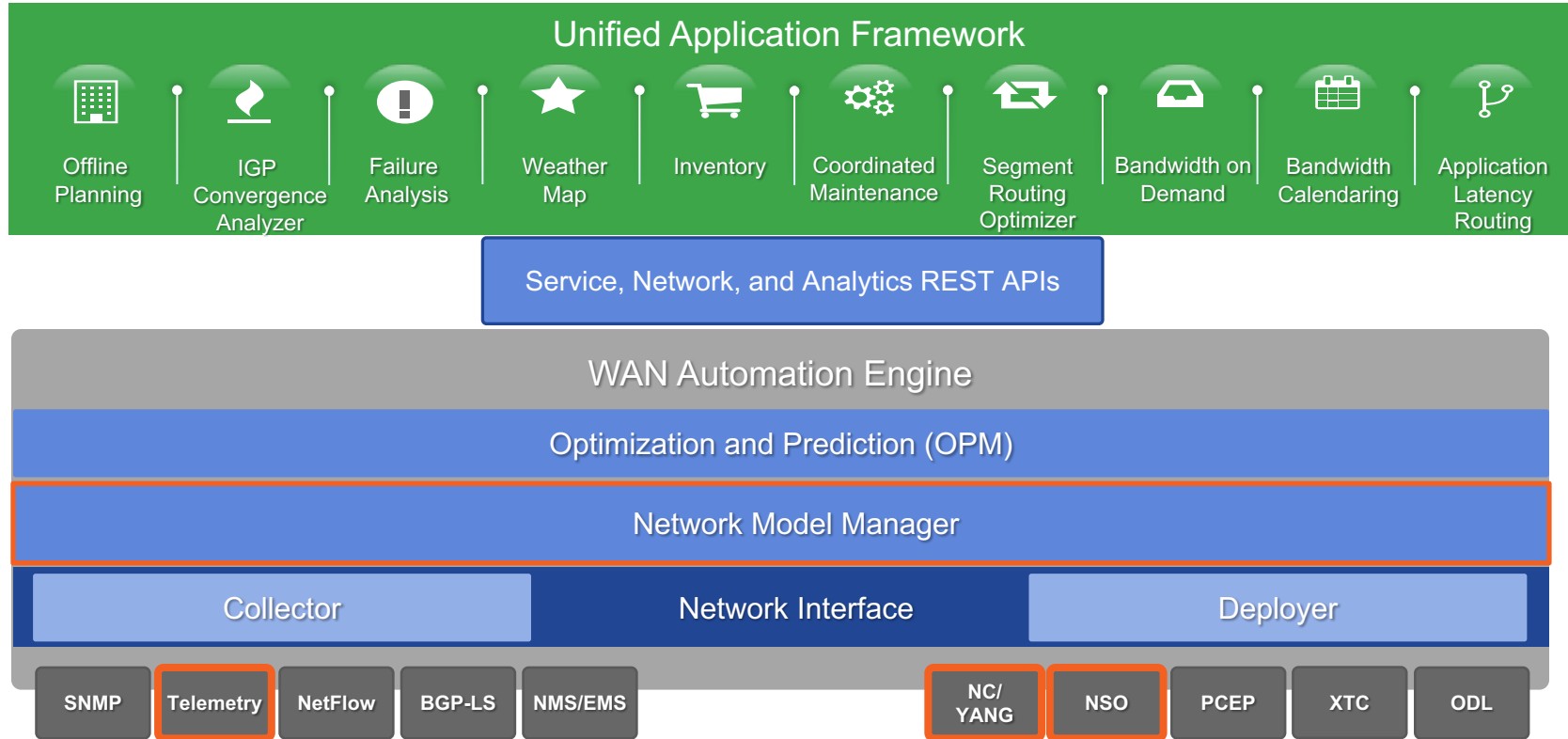
- Integrity verification
- Immutable audit trail
- Dynamic lawful intercept

Innovations in Network Automation

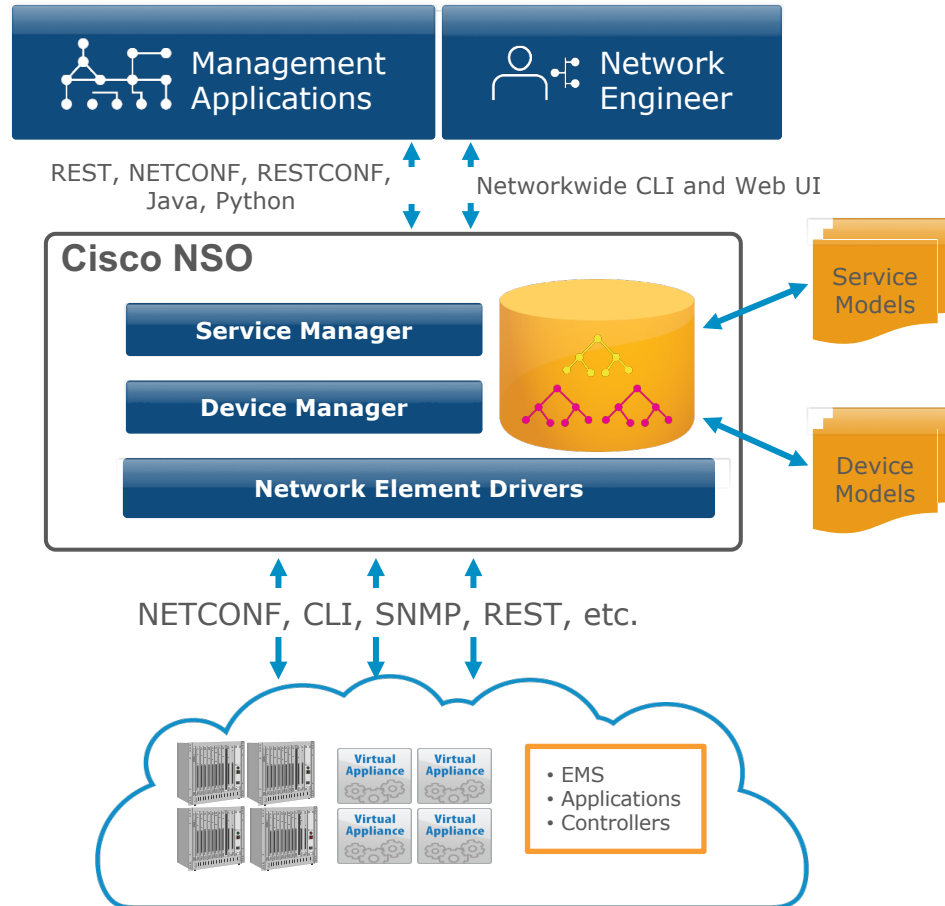
Network Automation using Models



Cisco WAN Automation Engine: Path Computation

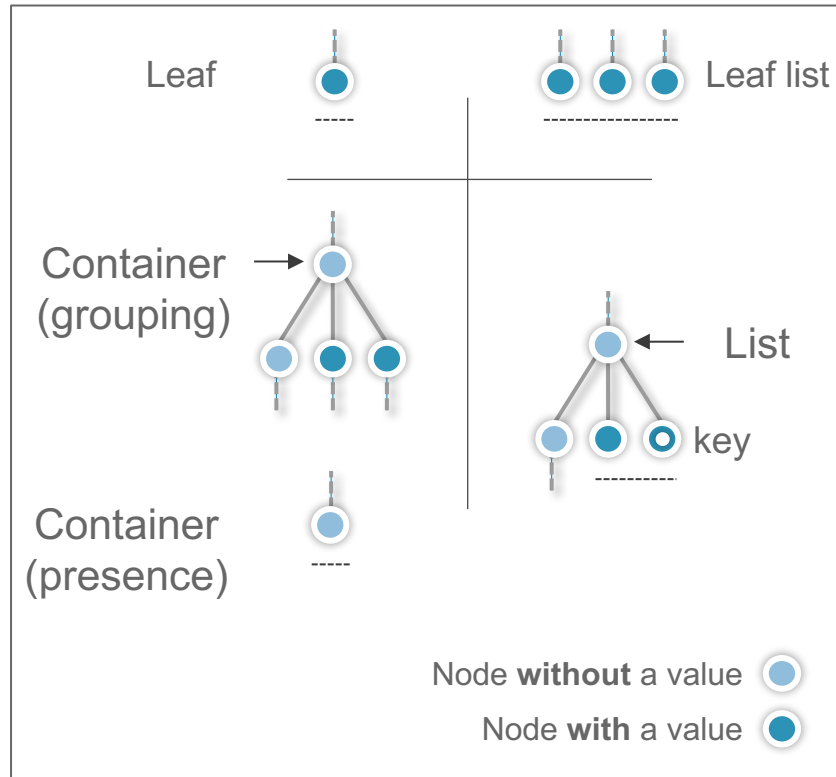


Cisco NSO: Model-Driven Service Orchestration



YANG Overview

- An **IETF data modeling language**
 - Configuration data
 - State data
- Plain text, human readable
- Tree structure
- Main node types:
 - **Leaf**: node with name and value of certain type
 - **Leaf list**: sequence of leaf nodes
 - **Container**: groups nodes
 - **List**: Sequence of records with key leaf nodes
- <https://tools.ietf.org/html/rfc6020>

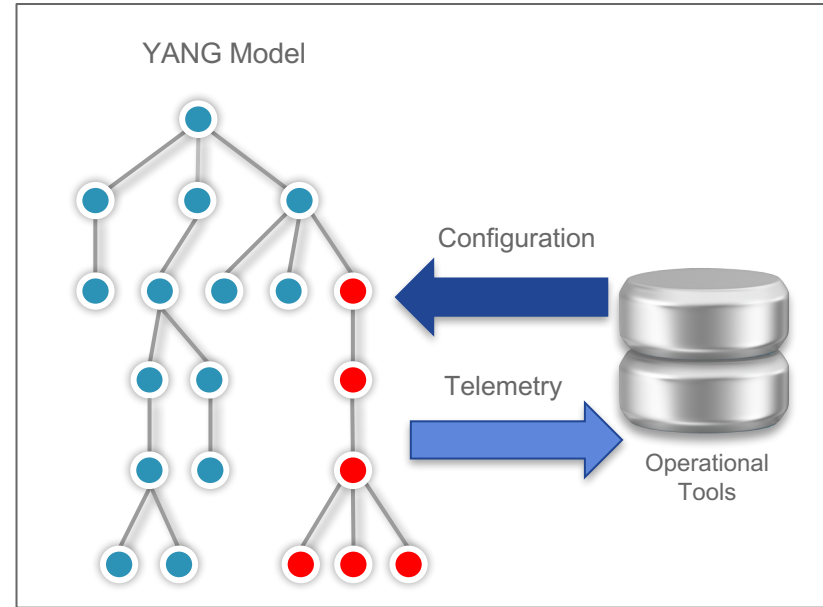


YANG Layer 3 VPN Service Model Example

```
34  container vpn {
35
36      list l3vpn {
37          description "Layer3 VPN";
38
39          key name;
40          leaf name {
41              tailf:info "Unique service id";
42              tailf:cli-allow-range;
43              type string;
44          }
45
46          list endpoint {
47              key "id";
48              leaf id {
49                  tailf:info "Endpoint identifier";
50                  type string;
51              }
52              leaf as-number {
53                  description "AS used within all VRF of the VPN";
54                  tailf:info "MPLS VPN AS number.";
55                  mandatory true;
56                  type uint32;
57              }
58          }
59      }
60  }
```

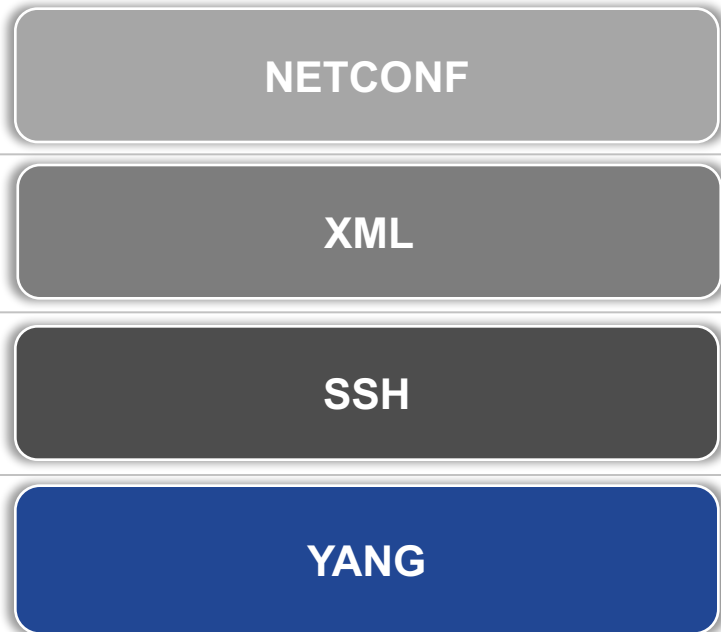
The Role of Models in Programmability

- 'Plug and Play' discovery of device capabilities
- Distributed as part of Operating System
- Also **openly published** (GitHub)
 - Open Models (OpenConfig / IETF)
 - Native / Vendor Models
- Can be used to automatically generate programmatic APIs
- Modeled in **YANG**
- YANG Maps easily to XML, seamless integration with **NETCONF RPCs**
- Enables **automation**

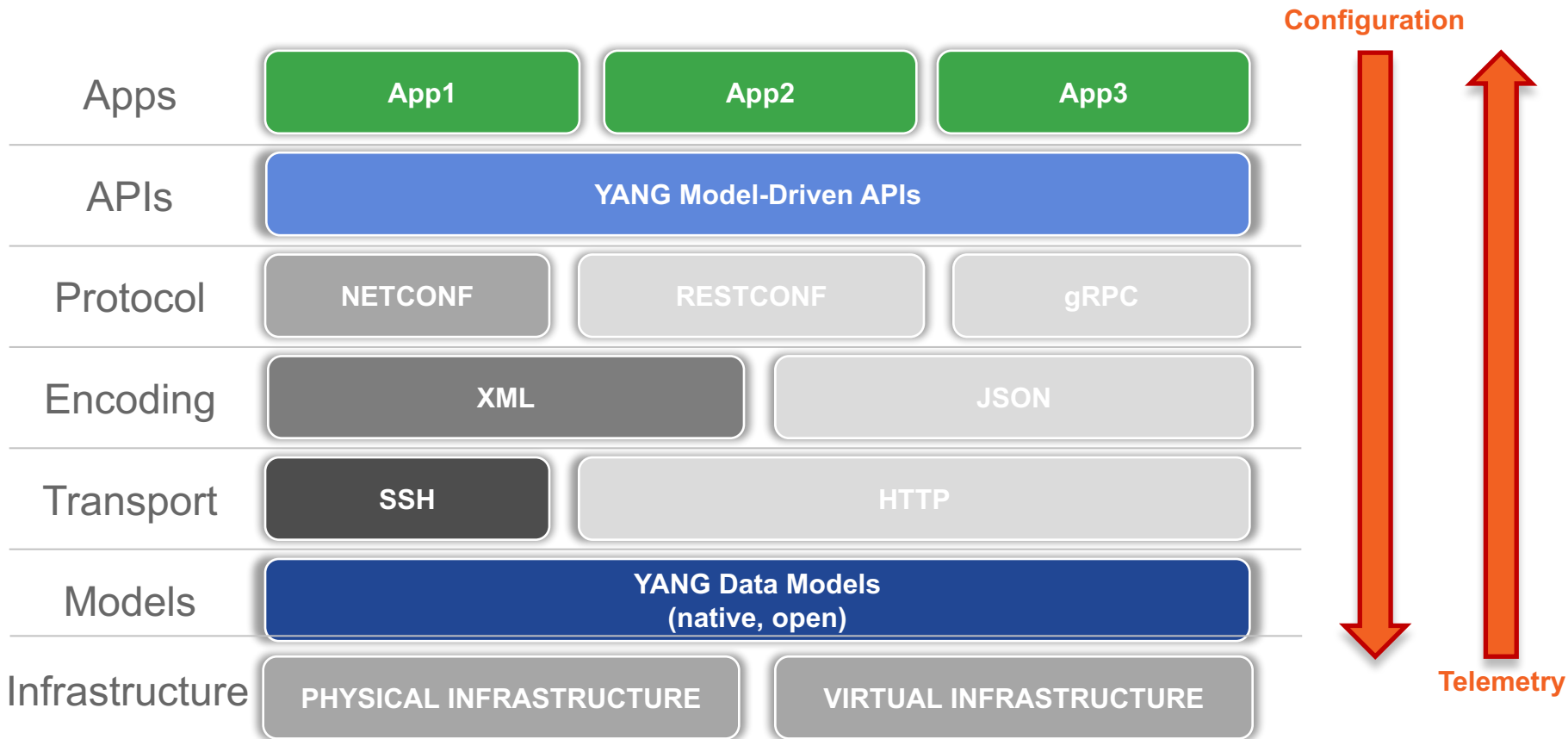


NETCONF Overview

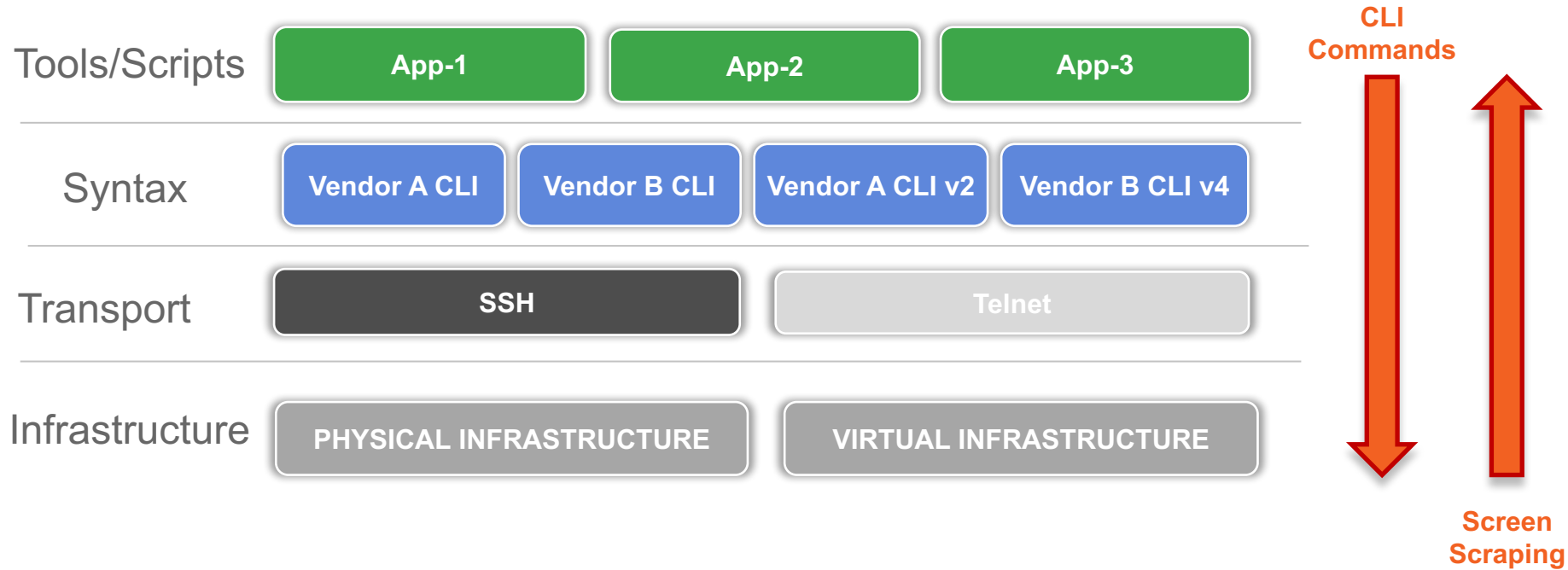
- An **IETF network management protocol**
- Device capabilities exchanged during session initiation
- Rich functionality to manage configuration and operational (state) data
- Supports **running**, **candidate** and **startup** configurations
- Client/app initiate request towards server/device
- Operations defined as **RPCs** (request / reply) in XML
- <https://tools.ietf.org/html/rfc6241>



Model-Driven Programmability Stack



Command Line Interface (CLI) Scripts



Command Line Interface (CLI) Script Issues

- Lack of transaction management: Can leave network “half configured”
- No structured error management: Very difficult to cover all failure cases
- Lack a common data model (across platforms, even same vendor)
- Ever changing structure and syntax of commands: New OS versions
- Scripts fragile and costly to maintain: Each vendor, Each OS, Each OS version

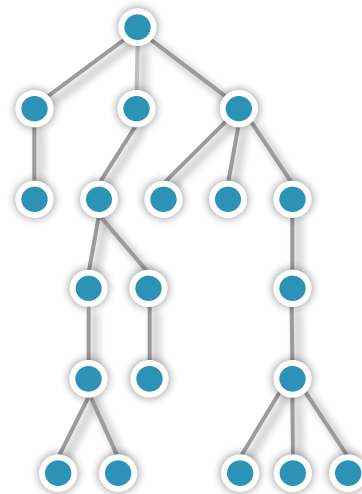
“CLIs are designed to be used by humans
and not an API for programmatic access.”

Model-Driven APIs: YANG Development Kit (YDK)

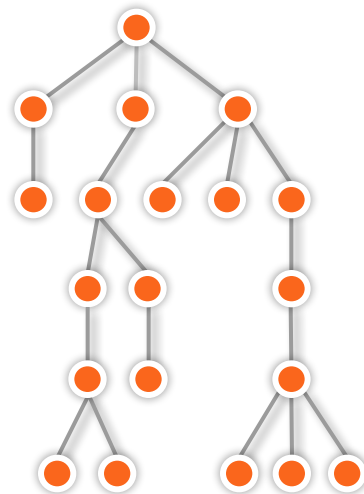


- Enable DevOps teams to use models in their **in-house tools**
- Enable developers to **write applications** or tools using infrastructure YANG models
- Create programming language **APIs from YANG models**
- Simplify application development
- One-to-one correspondence between model and class hierarchy
- Multi-language
 - Python, C++, Ruby, Go, and more.
- <https://developer.cisco.com/site/ydk/>

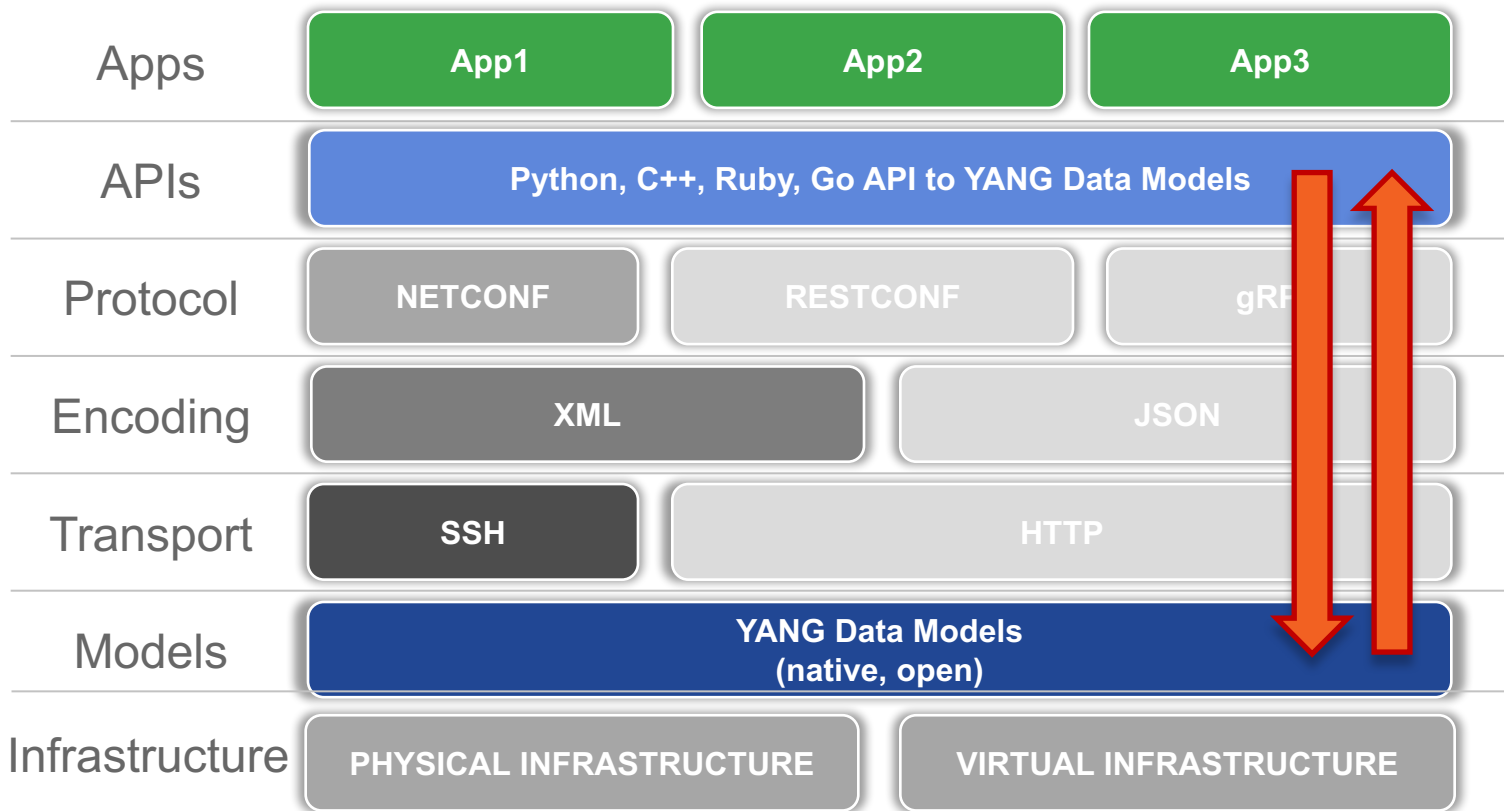
YANG Model



Class Hierarchy
(Python, C++, Ruby, Go)



Model-Driven API Programmability Stack



YDK Example: Telemetry Configuration **IOS-XR**

```
48 telemetry model-driven
49   destination-group DGroup1
50     address family ipv4 192.168.10.3 port 5432
51     encoding self-describing-gpb
52     protocol tcp
53   !
54 !
```

YDK Example: Telemetry Configuration Python

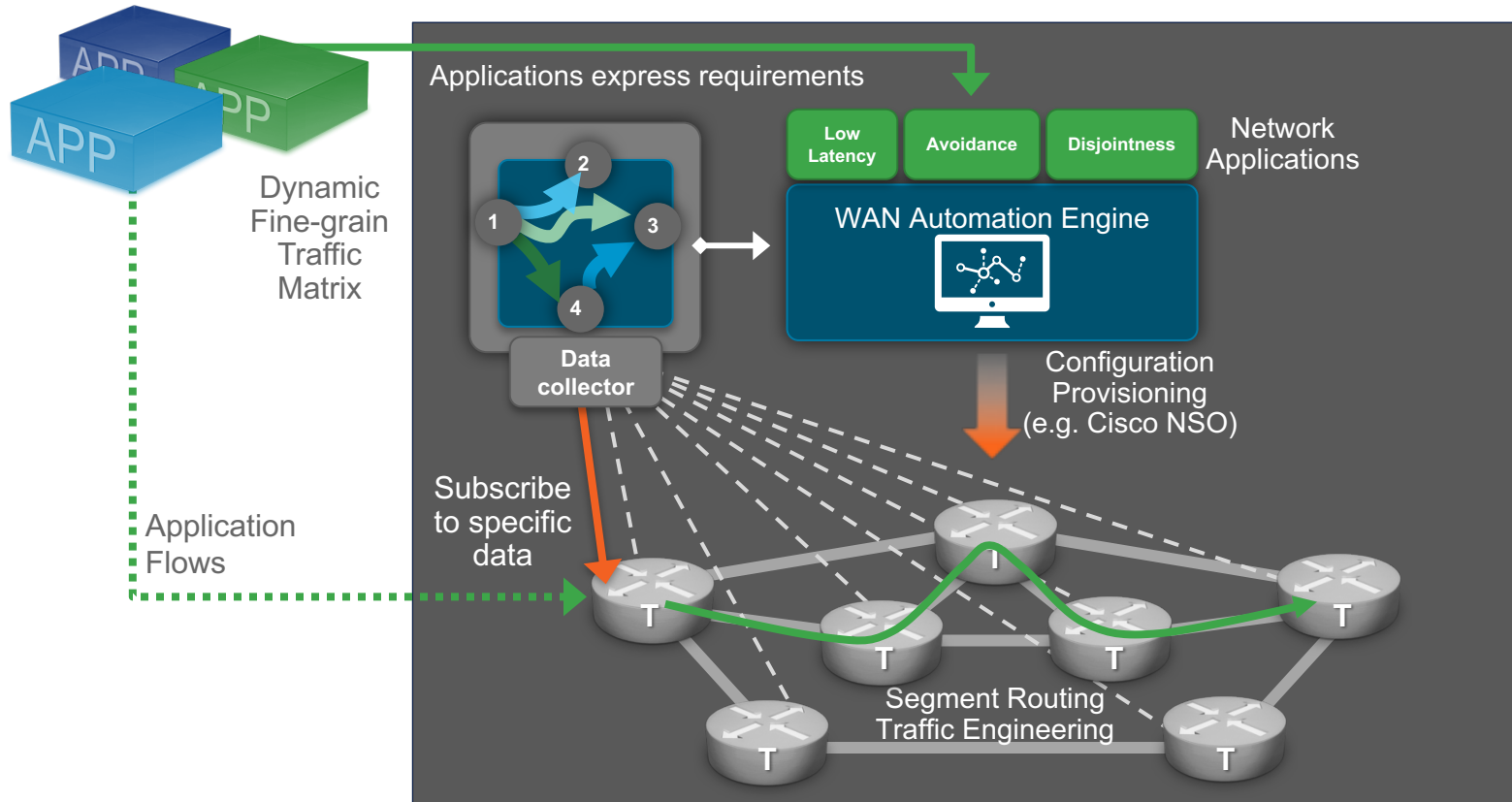
```
39 def config_telemetry_model_driven(telemetry_model_driven):
40
41     """Add config data to telemetry_model_driven object."""
42     # destination group
43     destination_group = telemetry_model_driven.destination_groups.DestinationGroup()
44     destination_group.destination_id = "DGROUP1"
45
46     # set the destination group target
47     ipv4_destination = destination_group.ipv4_destinations.Ipv4Destination()
48     ipv4_destination.destination_port = 5432
49     ipv4_destination.ipv4_address = "192.168.10.3"
```

Benefits of Model-Driven Automation & Programmability

- Standardised way of **discovering and interacting** with APIs
- Significantly reduces manual customisation and configuration, therefore **lower integration costs**
- Multiple model types (native, common, OpenConfig, IETF, etc.)
- Models are structured, **human-readable but computer friendly**
- Models **decoupled** from transport, protocol and encoding
- Automation platforms and code benefit from Model-driven APIs for **abstraction and simplification**

Bringing it Together

The Vision: Closed Loop Network Automation



Thank you



Service Provider Summit

at **Cisco** *live!*