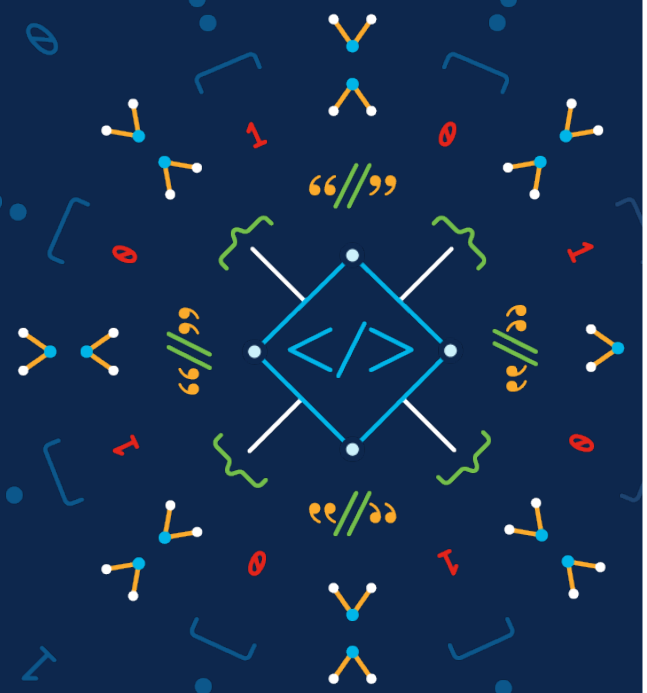
Developer Days
Automation

cisco
The bridge to possible

Model Everything – Automate Anything

Remove the Limits on What You Automate with NSO



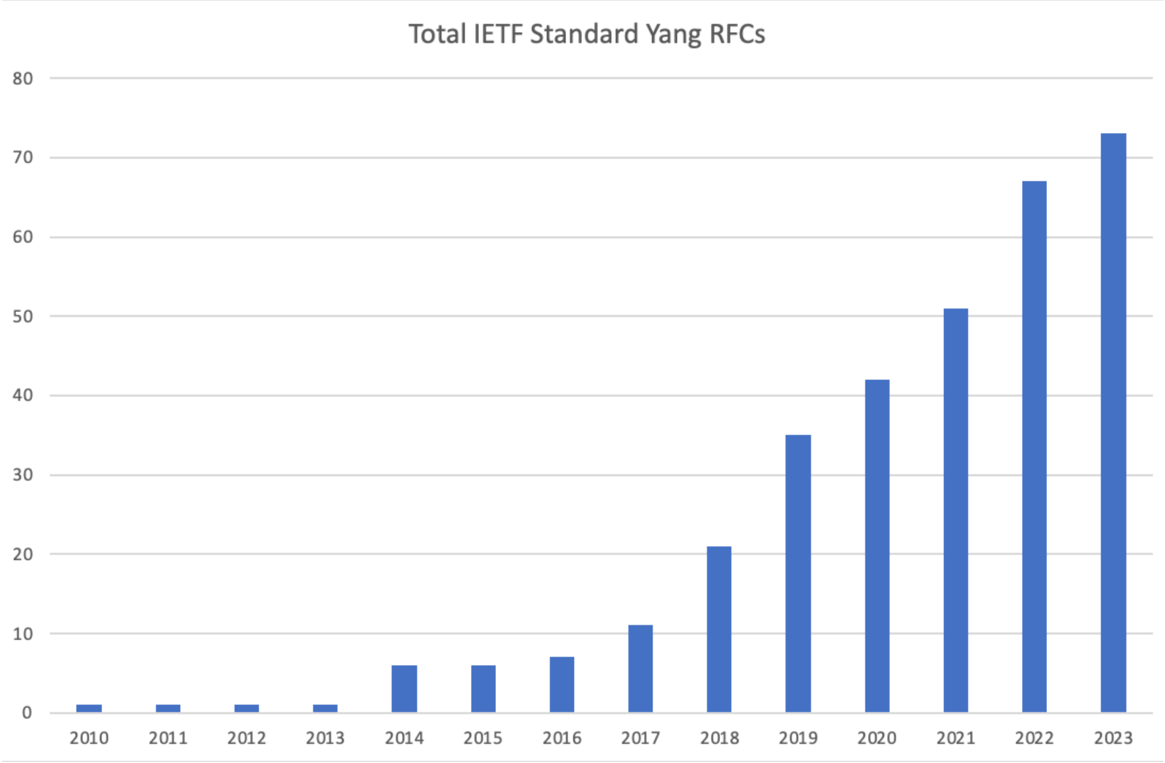
Scott Barvick
CTO – Data Ductus
December 2023

Opportunity – Extract more automation value from NSO

- Many things get touched in the network that do not have NEDs
- Start with a data model (YANG or other!) and you can generate the core infrastructure necessary to automate using NSO

Model Everything

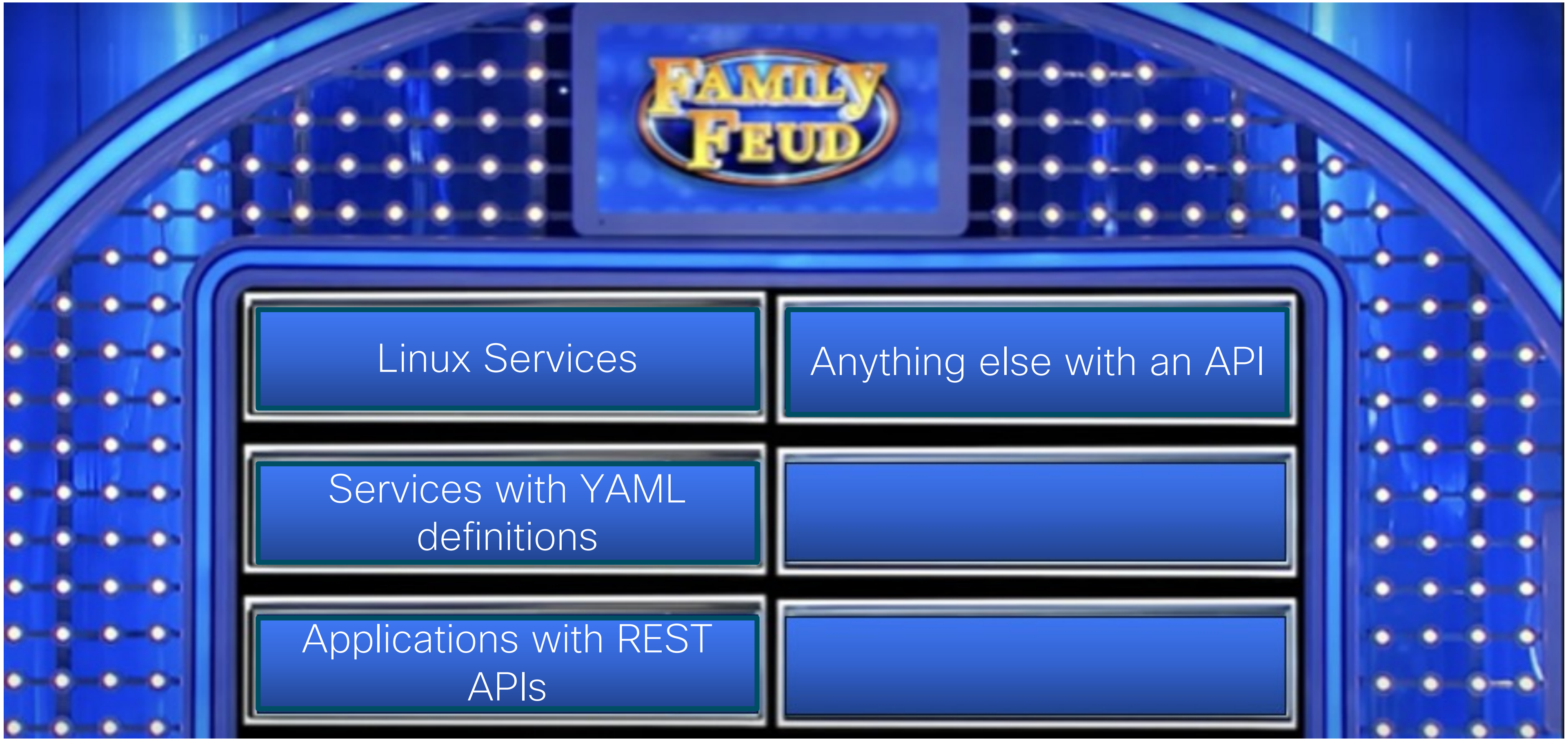
Great progress for network function YANGs



Plus -
Total # of YANG files in git:

- IETF: 160
- OpenConfig: 200
- Broadband Forum: 300
- IEEE: 46
- + MEF, ETSI, etc

What else should be modeled?



Linux Services

Anything else with an API

Services with YAML definitions

Applications with REST APIs

Modeling Linux Services

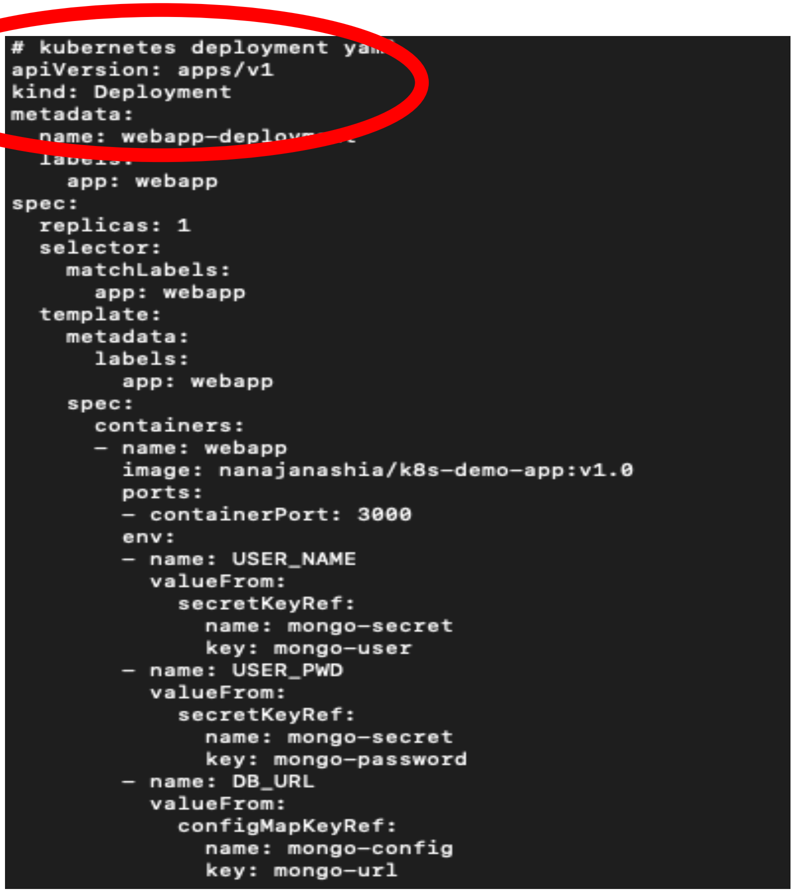
NTP, DNS, SSHD

- Easy to understand
- Text files to configure
- Modeled with OpenConfig YANGs

Open-source routing/networking

- More complex
- Different configuration options
- Probably can use OpenConfig or IETF YANGs

Modeling YAML-based Services



```
# kubernetes deployment yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: webapp-deployment
  labels:
    app: webapp
spec:
  replicas: 1
  selector:
    matchLabels:
      app: webapp
  template:
    metadata:
      labels:
        app: webapp
    spec:
      containers:
      - name: webapp
        image: nanajanashia/k8s-demo-app:v1.0
        ports:
        - containerPort: 3000
        env:
        - name: USER_NAME
          valueFrom:
            secretKeyRef:
              name: mongo-secret
              key: mongo-user
        - name: USER_PWD
          valueFrom:
            secretKeyRef:
              name: mongo-secret
              key: mongo-password
        - name: DB_URL
          valueFrom:
            configMapKeyRef:
              name: mongo-config
              key: mongo-url
```

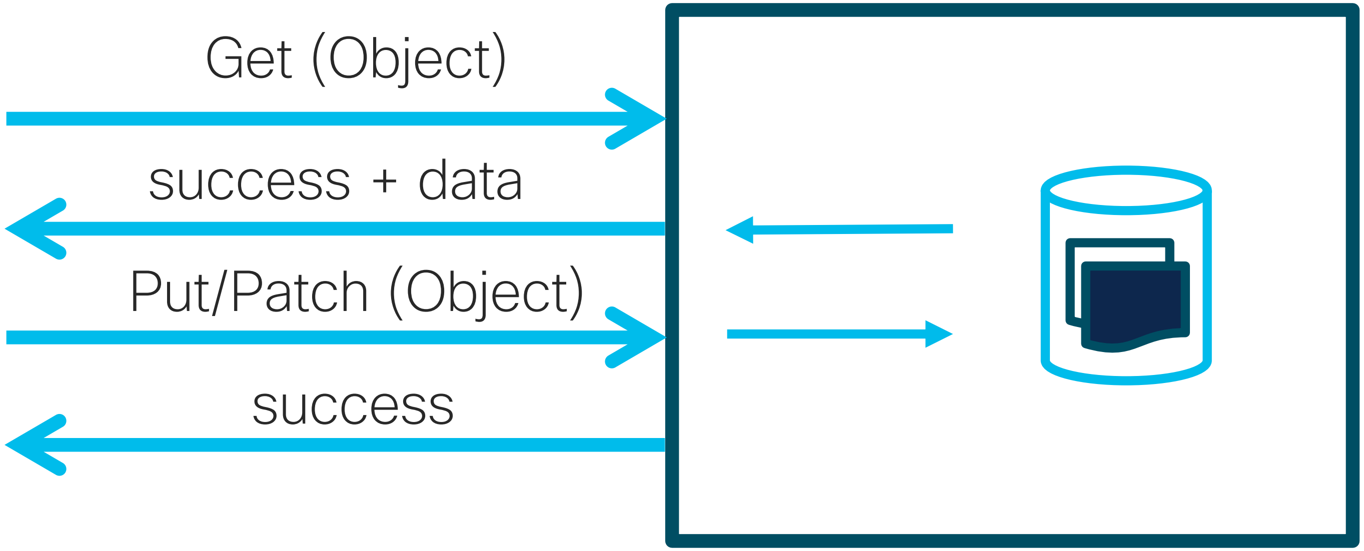
YAML is a data serialization format, not a data model

e.g. what are the allowed values for the ‘kind’ field?

Kubernetes is a great example

- Mature, well-understood capabilities
- Rich implementation ecosystem
- YAML deployment with kubectl

Modeling functions with REST APIs



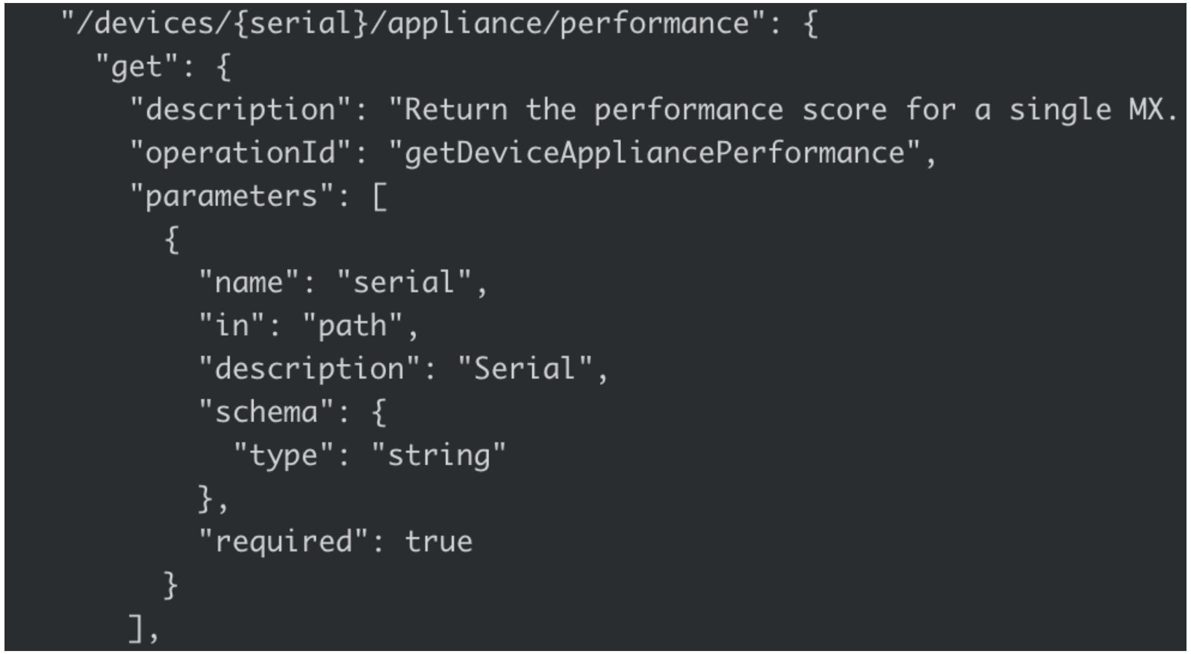
How is “Object” defined?

- Ad hoc API documentation – historical and no modeling
- Swagger->OpenAPI – growing with modeling!
- YANG – None!

Generating YANG from OpenAPI (fka Swagger)

- OpenAPI spec includes a ‘data model’ – similar to YANG
- 120+ “generators” generate client and server code in every language – and documentation and schema
- No YANG – until now

(https://openapi-generator.tech)



```
"/devices/{serial}/appliance/performance": {
  "get": {
    "description": "Return the performance score for a single MX.
    "operationId": "getDeviceAppliancePerformance",
    "parameters": [
      {
        "name": "serial",
        "in": "path",
        "description": "Serial",
        "schema": {
          "type": "string"
        },
        "required": true
      }
    ],
```

Modeling Summary

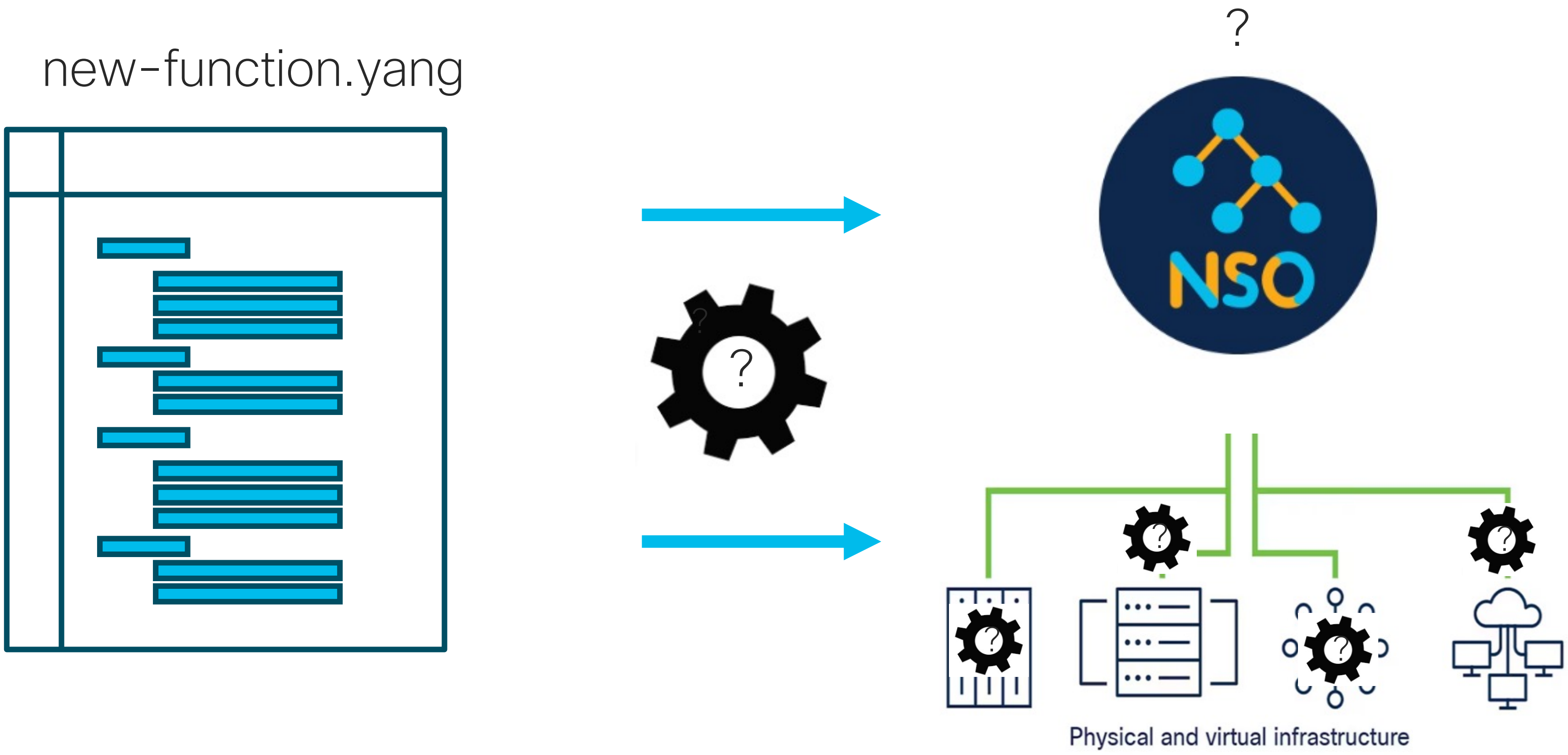
- If there is a function out there you want to automate through NSO, start with a model

- Modeling with YANG is not that difficult for the NSO community if none exists – we do it all the time with service models

- Many models exist if you know where to look

Automate Anything

OK, I have a model, now what?



new-function.yang

?

Physical and virtual infrastructure

“Low code” is possible with build/tools and generation

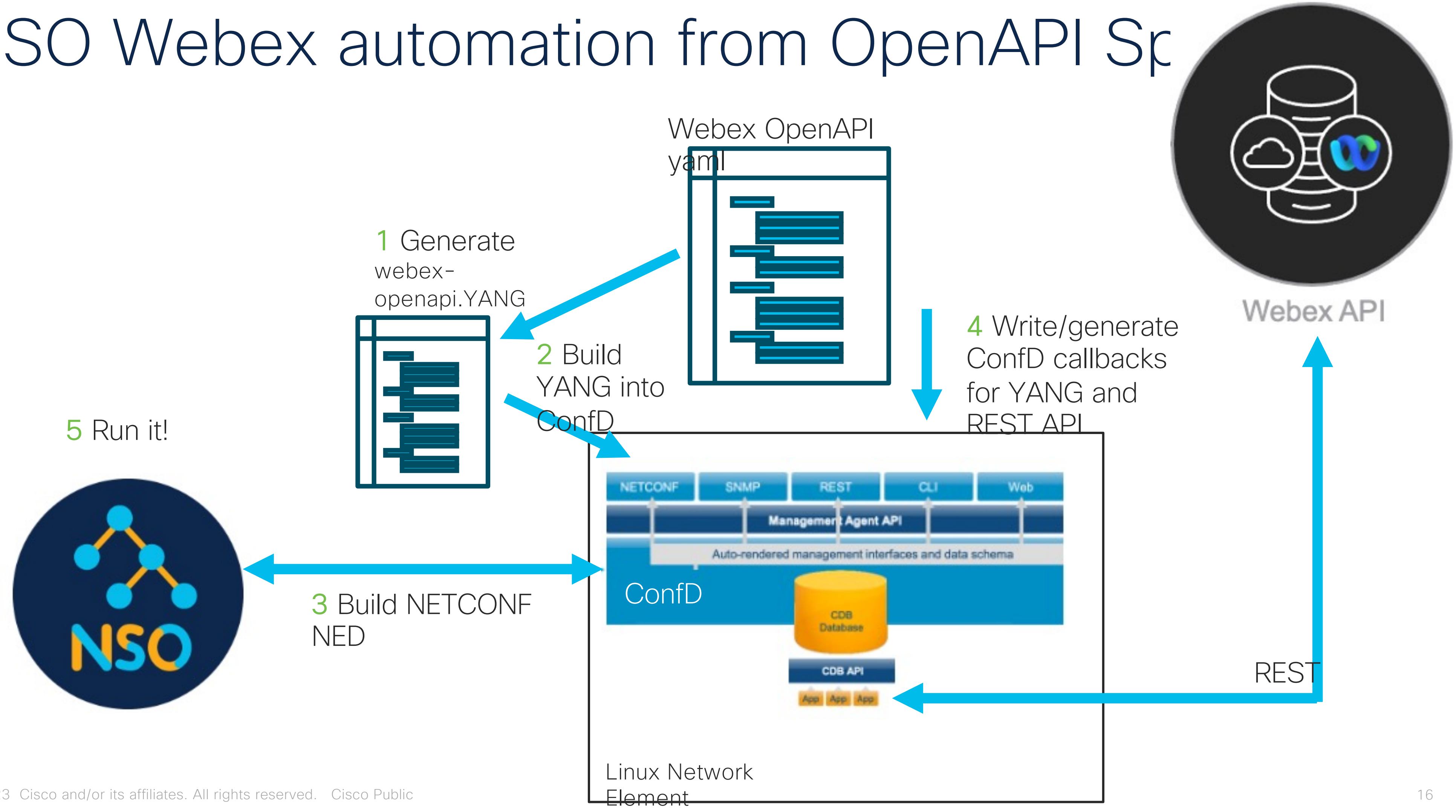
Device NETCONF server	ConfD YumaWorks netopeer2 libnetconf2	✓
YANG into NETCONF server	Build	✓
NETCONF NED	netconf-ned-builder pioneer	✓
NETCONF server callbacks for YANG	Code Development -> Generation	✓

Demo

Goal

Demonstrate ease of building and deploying code to handle simple REST API automation through NSO to the REST server using NETCONF

NSO Webex automation from OpenAPI Sp



Webex OpenAPI yaml

1 Generate webex-openapi.YANG

2 Build YANG into ConfD

3 Build NETCONF NED

4 Write/generate ConfD callbacks for YANG and REST API

5 Run it!

Webex API

REST

Linux Network Element

Summary

- Get more out of NSO with models for network functions, writing them if they don’t exist
- Start with tools and basic functionality and build from there
- Push for OpenAPI specs for REST APIs – they are surprisingly hard to find
- Watch for rapid advances in generating the interface code to make process of YANG to automation go even faster.

CISCO

The bridge to possible