Ansible x NAPALM

Automate the network lifecycle in a multi-vendor environment

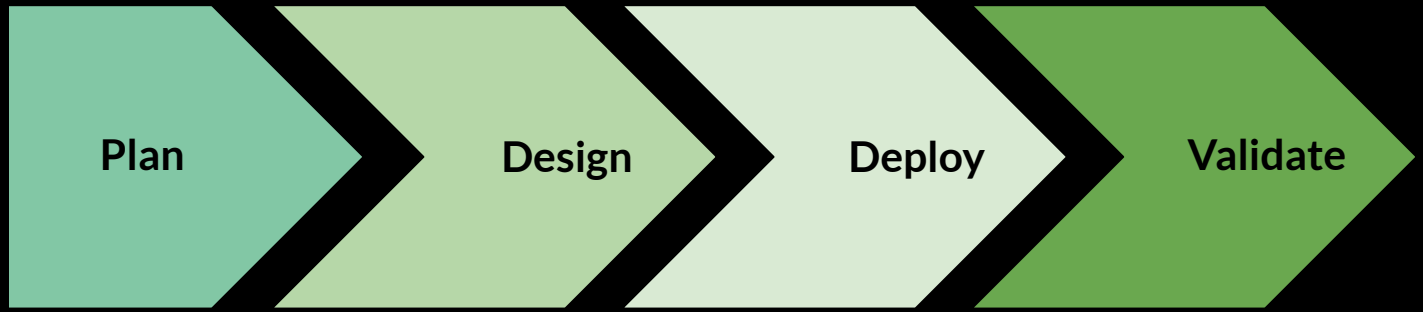
Overview

- Challenges
- The Network Lifecycle
- Scenario
- Demo

What are the challenges with modern network environments?

- Multi-vendor networks are the norm
- Need to do more with the same….or less
- Complex network environments - cloud, multi-cloud, multiple data centres, SD-WAN
- “Copypasta” is still the deployment tool of choice
- Network automation is hard

The Network Lifecycle

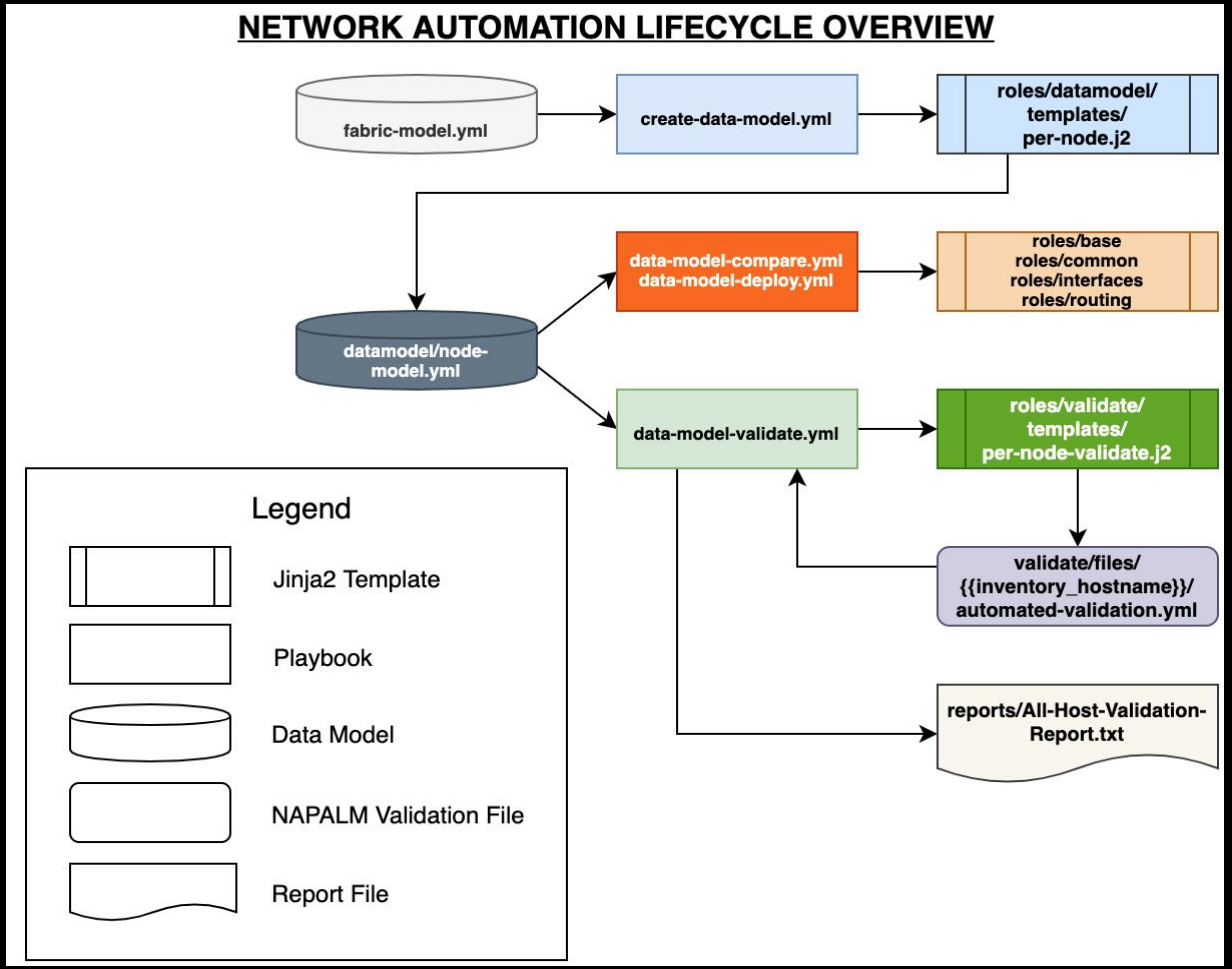


Scenario

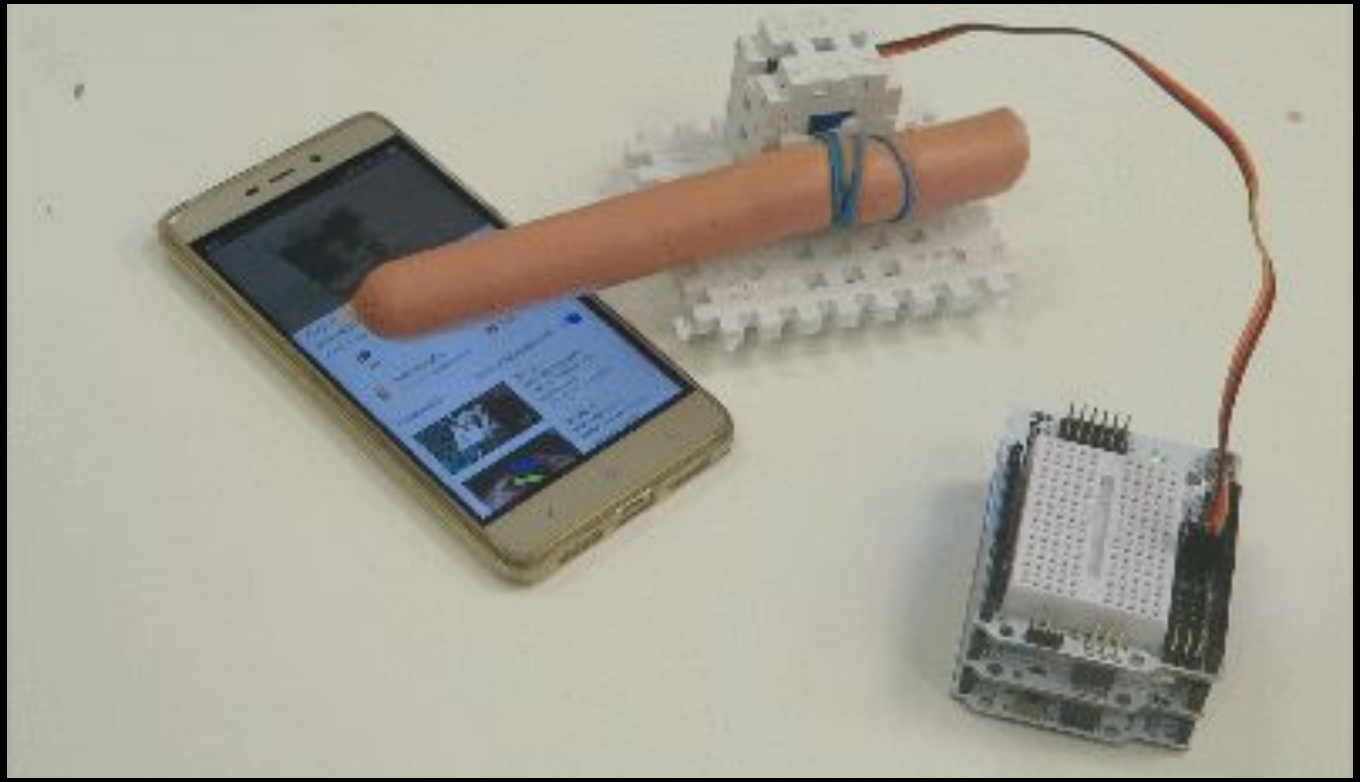
Create a multi-vendor BGP fabric

- Should be able to add devices to the fabric.
- Should be able to replace Vendor X with Vendor Y and the fabric still works.
- Interface and BGP neighbor descriptions are standardised.
- Validation tests must be automatically generated and validated.

Network Topology


NETWORK AUTOMATION LIFECYCLE OVERVIEW
fabric-model.yml
create-data-model.yml
roles/datamodel/
templates/
per-node.j2
data-model-compare.yml
data-model-deploy.yml
roles/base
roles/common
roles/interfaces
roles/routing
datamodel/node-
model.yml
data-model-validate.yml
roles/validate/
templates/
per-node-validate.j2
validate/files/
{{inventory_hostname}}/
automated-validation.yml
reports/All-Host-Validation-
Report.txt
Legend
Jinja2 Template
Playbook
Data Model
NAPALM Validation File
Report File

Demo Time



Data Models

- All vendors have their own implementation of the same thing
- Need to normalise the data using a data model
- Provide a translation layer between user and network device

NX-OS - Cisco Nexus Operating System

```
interface Ethernet1
   description To router-01 - Interface1
   no switchport
   ip address 192.168.30.1/30
!
```

JUNOS - Juniper Operating System

```
interfaces {
    ge-0/0/1 {
        unit 0 {
            description "To router-01 - Interface1";
            family inet {
                address 192.168.30.1/30;
            }
        }
    }
}
```

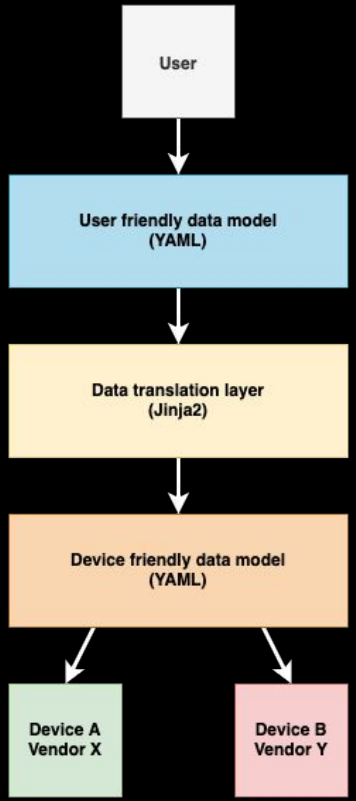
IOS - Cisco Operating System

```
interface GigabitEthernet1
 description To router-01 - Interface1
 ip address 192.168.30.1 255.255.255.252
!
```

EOS - Arista Operating System

```
interface Ethernet1
   description To router-01 - Interface1
   no switchport
   ip address 192.168.30.1/30
!
```

Data Models (cont.)

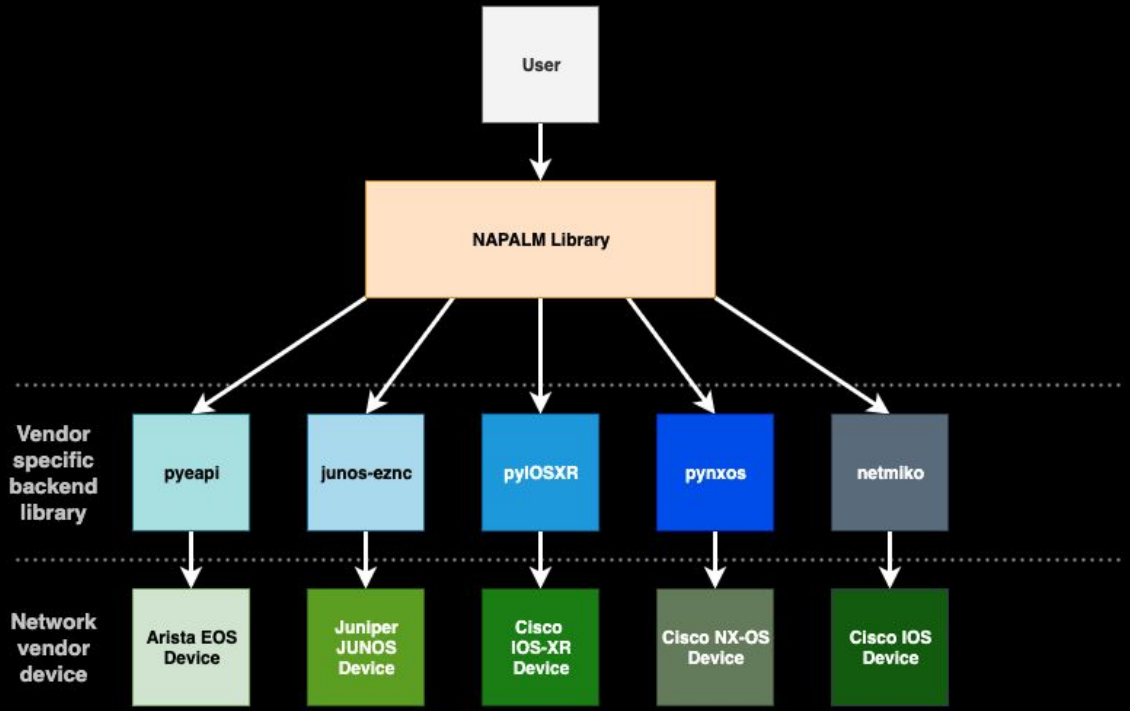


NAPALM

(Network Automation and Programmability Abstraction Layer with Multivendor support)

- NAPALM is a Python library which provides a unified API across network devices from various vendors.
- Write code once, then reuse it across vendors.
- napalm-ansible - Ansible modules which uses napalm to perform operations on network devices

NAPALM (cont.)


User
NAPALM Library
Vendor specific backend library
pyeapi
junos-eznc
pyIOSXR
pynxos
netmiko
Network vendor device
Arista EOS Device
Juniper JUNOS Device
Cisco IOS-XR Device
Cisco NX-OS Device
Cisco IOS Device

NAPALM Validate (cont.)

- YAML based validation files
- Use to validate state of the device, rather than configuration
- Can use regular expressions to validate values

```
# Dynamically generated NAPALM validation rules
---
- get_facts:
    fqdn: lab-junos-01.lab.dfjt.local
    hostname: lab-junos-01
- get_bgp_neighbors:
    global:
      peers:
        192.168.30.2:
          description: lab-csr-01.lab.dfjt.local
          is_enabled: true
          is_up: true
          local_as: 65015
          remote_as: 65016
      router_id: 192.168.40.15
- get_interfaces:
    ge-0/0/1.0:
      description: To lab-csr-01.lab.dfjt.local - GigabitEthernet3
      is_enabled: true
      is_up: true
    lo0.0:
      description: Loopback_Interface
      is_enabled: true
      is_up: true
- get_interfaces_ip:
    ge-0/0/1.0:
      ipv4:
        192.168.30.1:
          prefix_length: 30
    lo0.0:
      ipv4:
        192.168.40.15:
          prefix_length: 32
- get_lldp_neighbors_detail:
    ge-0/0/1.0:
    - remote_system_name: lab-csr-01
```

Further Resources

- Code repository: https://github.com/writememe/ansible-mel-meetup-2020
- NAPALM Website: https://napalm-automation.net/
- NAPALM Documentation: https://napalm.readthedocs.io/en/latest/
- NAPALM Support: https://networktocode.slack.com - #napalm channel