Increasing Adoption of Network Automation

• 70% still use manual CLI-based network configuration
• automating 70% of network configuration changes can cut the number of unplanned outages by 50%
• **RFC 8199** data model classification
  • Device models
    • 100s of models available
      • IETF, OpenConfig, vendors
  • Service models
    • IETF - **L2SM**, **L3SM**
    • MEF – available only to members
YANG Model Layers

Network Service YANG data models
- VPWS L2VPN
- VPLS L2VPN
- L3VPN

Network Device YANG data models
- BGP
- MPLS
- Routing
- Interfaces
Network Automation

• Easy migration from Device Management to Network Service Management

• New Management API (Device and Network Service)
  • Operator programs directly changing Op States of Control-Plane or Forwarding-Plane elements

• YANG data-models (Device and Network Service)
  • Operators only need to load the required network-service data-models in their network (L2 switching, L3 routing, VPN, Firewall, Class-of-service, etc..)
  • Lots of choices of data-models (vendors, IETF, OpenConfig, user-customized, etc)

• Same capabilities regardless of the methods chosen
L2VPN data-model

module: l2vpn

+-rw network-service
  +-rw l2vpn* [model-id instance-id]
    +-rw model-id uint64
    +-rw instance-id uint64
    +-ro service-id? string
    +-rw l2vpn-type enumeration
  +-rw provider-edge* [router-address interface]
    +-rw router-address union
    +-rw interface string
    +-ro service-id? string
  +-rw neighbor* [router-address]
    +-rw router-address inet:ip-address
L2VPN Network Instance (device view)

network-service: {
  l2vpn: {
    model-id: 177,
    instance-id: 11,
    l2vpn-type: vpls,
    service-id: 101, # system generated
    provider-edge: [
      {
        router-address: "1.1.1.1",
        interface: "xge3.101",
        neighbor: [ { router-address: "2.2.2.2" } ]
      },
      router-address: "2.2.2.2",
      interface: "xge6.101",
      neighbor: [ { router-address: "1.1.1.1" } ]
    }
  }
}

- When devices are managed individually, users need to specify all parameters
Network Automation: L2VPN Network Instance (network-service view)

```
network-service: {
  l2vpn: {
    model-id: 177,
    instance-id: 11,
    l2vpn-type: vpls,
    service-id: 101,    # system generated
    provider-edge: [
      {
        router-address: "1.1.1.1",
        interface: "xge3.101",
        neighbor: [ { router-address: "2.2.2.2" } ]    # system generated w/ Topology discovery
      },
      router-address: "2.2.2.2",
      interface: "xge6.101",
      neighbor: [ { router-address: "1.1.1.1" } ]    # system generated w/ Topology discovery
    }
  }
}
```

- When using the network-service view, network topology is discovered by system Discovery Service, the l2vpn tunnel connections are calculated by the system SW
Volta, from the Italian word voltare which means turn. Now is the time to turn networking around to new architecture.