



Smart Management for Robust Carrier Network Health & Reduced TCO

Colby Barth, Distinguished Engineer, Juniper Networks



PHILOSOPHICALLY SPEAKING ...

Only you know your network!

Enable smart network management through network
Orchestration with a published XML API

- Core element of router/switch OS ensuring that as new features are released they are immediately available to the orchestration system
- On-box & off-box solutions

How does this make the network better?

Eliminate User Errors

- Block configurations that break the rules
- Correct errors as they are detected
- Improve uptime

Leverage your knowledge

- Codify Customer-specific Business Rules
- Simplify configuration - Work in higher level constructs
- Define commands and command-line argument

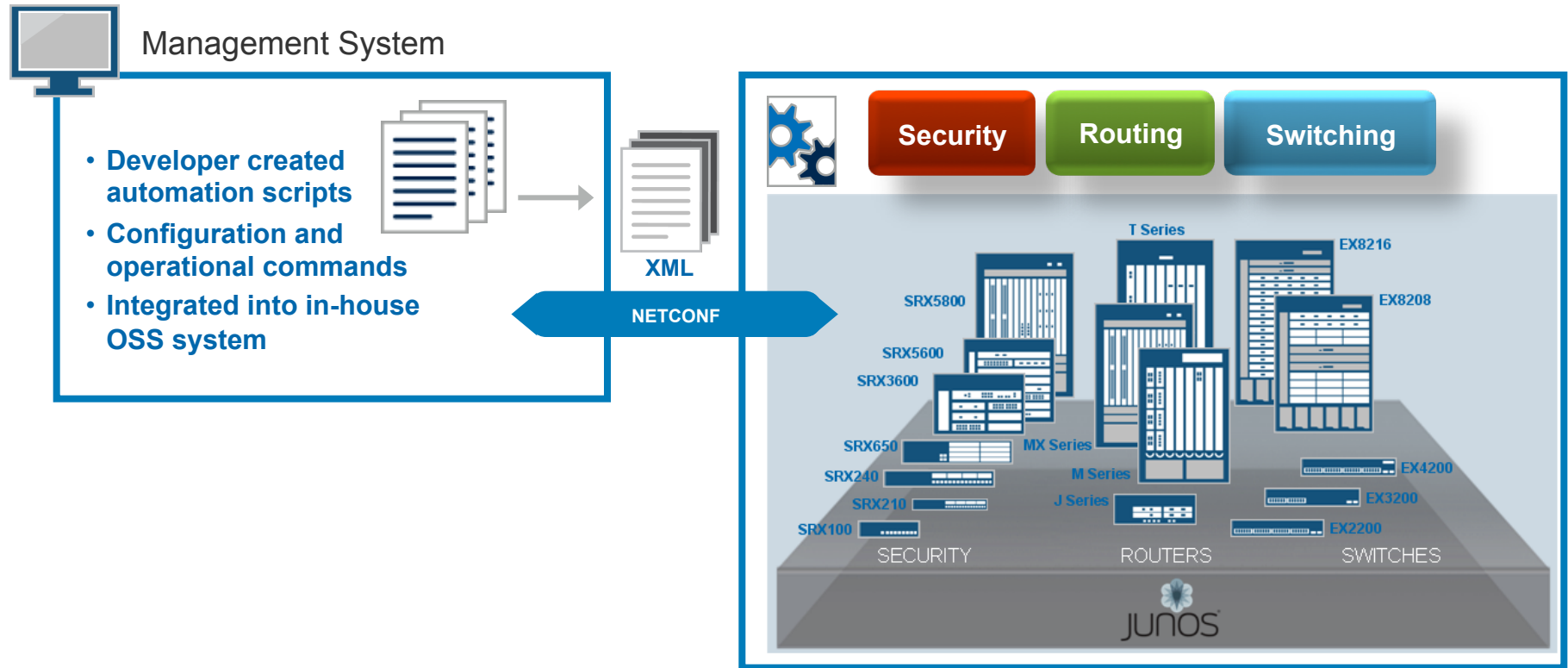
Translate in cost savings

- Reduce OPEX
- Reactions to live events enabling a self monitoring network
- Improving uptime

A FRAMEWORK FOR NETWORK ORCHESTRATION

The “off-box” automation scripts (“orchestration”) centrally located

These scripts communicate to the Network devices using the NETCONF protocol



“ON-BOX” AUTOMATION

“Scripting” for configuration automation, operations customization, and automated response to network event conditions



APPLICATION OF CONFIGURATION AUTOMATION

Commit Scripts

✓ Human Factors

- No accidental deletion of mandatory [interfaces] & [protocols]
- If default route is not filtered, fail the commit, better yet fix it

✓ Best Practices

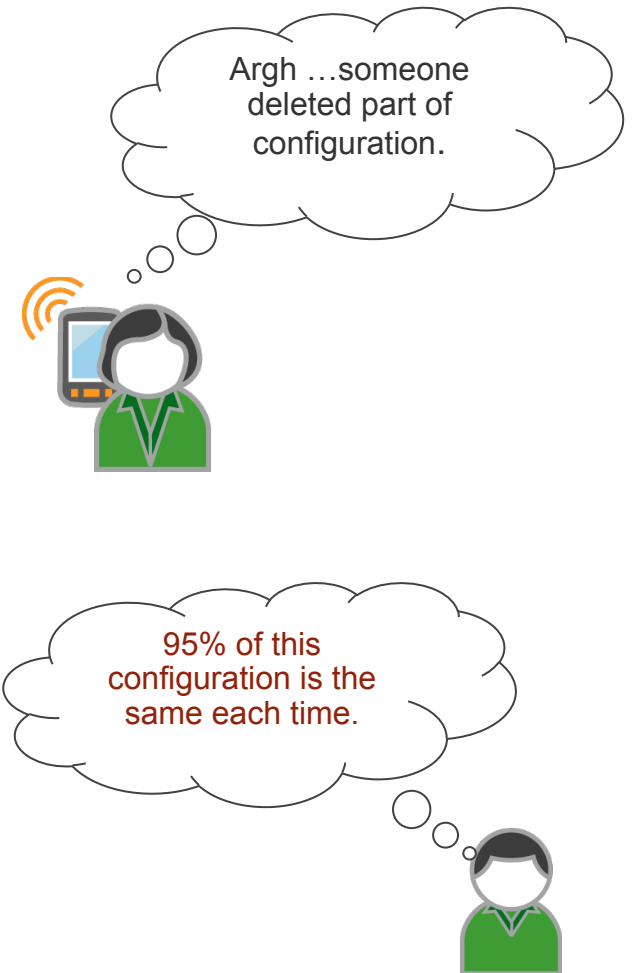
- Every interface must have a description
- Loopback interface configuration is always present

✓ Security/Performance Guidelines

- Firewall must have trailing explicit deny
- Restrict guest access to a floor by reject guest VLAN tag config on access switch trunk ports

✓ Abstract Configuration

- Abstract set up of VPLS from many steps to one macro
- Apply pre-defined VoIP port template on any switch port that gets a matching description “data-phone”



SIMPLIFYING COMPLEX CONFIGURATIONS[1]

```
apply-macro ipvpn-cust1 {  
  cust-name customer-1;  
  description "Customer #1";  
  egress-processing cust-output-dscp-4q-profile-1;  
  egress-shaper 500k;  
  inet-address 10.1.3.1/30;  
  unit 551;  
  ingress-processing cust-input-dscp-profile-A;  
  route-distinguisher 65001:301;  
  igp ospf;  
}
```

```
[edit class-of-service]  
traffic-control-profiles {  
  ipvpn-customer1-4-queue-shaper {  
    scheduler-map cust-output-dscp-4q-profile-1;  
    shaping-rate 500k;  
  }  
}
```

```
[edit interfaces lsq-1/2/0]  
unit 551 {  
  description "Customer #1";  
  family inet {  
    filter {  
      input cust-input-dscp-profile-A;  
    }  
    address 10.1.3.1/30;  
  }  
}
```

```
[edit routing-instances]  
root# show | display commit-scripts  
ipvpn-customer1 {  
  instance-type vrf;  
  interface lsq-1/2/0.551;  
  route-distinguisher 65001:301;  
  vrf-target target:65001:301;  
  vrf-table-label;  
  protocols {  
    ospf {  
      area 0.0.0.0 {  
        interface lsq-1/2/0.551;  
      }  
    }  
  }  
}
```

Example of a configuration macro

- Embed user-defined data set
- Expansion done by commit script
- Provides an abstracted, customized interface to your configuration needs

SIMPLIFYING COMPLEX CONFIGURATIONS[2A]

```
root@pe-router-1> op provision-cpe-profile
Entering data for CPE Configuration:
  Enter Customer name [string] [mandatory]: cust-1
  Enter Interface name [interface] [mandatory]: xe-0/2/0
  Enter VLAN Id [integer] [mandatory]: 111
  Enter Customer address [ip-address] [mandatory]: 1.1.1.1
  Enter Security profile [string] [mandatory]: default
```

Configuration:

CPE Configuration:

Customer name: cust-1
Interface name: xe-0/2/0
VLAN Id: 111
Customer address: 1.1.1.1
Security profile: default

Is this configuration accurate? (default: no) [yes/no]: yes
[loading configuration ...]
[finished]

CPE cust-1 has been provisioned
load complete

Example op script to build a higher level configuration

- User prompted small set of input
- Creates high-level data-structure (apply-macro)
- Expanded configuration driven by the apply-macro

```
apply-macro cust-1 {
  cust-name customer-1;
  description "Customer #1";
  interface-name xe-0/2/0;
  cust-public-ipaddr 1.1.1.1;
  security-profile default;
}
```



"Expanded
configuration"



SIMPLIFYING COMPLEX CONFIGURATIONS[2B]

```
apply-macro cust-1 {  
  cust-name customer-1;  
  description "Customer #1";  
  interface-name xe-0/2/0;  
  cust-public-ipaddr 1.1.1.1;  
  security-profile default;  
}
```

```
[edit services]  
stateful-firewall {  
  rule sf_rule {  
    match-direction input-output;  
    term 1 {  
      then {  
        accept;  
        syslog;  
      }  
    }  
  }  
  ...  
  service-set ServiceSet_1 {  
    stateful-firewall-rules sf_rule;  
    nat-rules rule_1;  
    next-hop-service {  
      inside-service-interface sp-0/0/0.1;  
      outside-service-interface sp-0/0/0.101;  
    }  
  }  
}
```

```
[edit services]  
nat {  
  pool public1 {  
    address 1.1.1.1/32;  
    port {  
      automatic;  
    }  
  }  
  rule rule_1 {  
    match-direction input;  
    term translate {  
      from {  
        source-address {  
          192.168.1.0/24;  
        }  
      }  
    }  
  }  
  ...  
}
```

```
[edit routing-instances]  
customer-1 {  
  instance-type virtual-router;  
  system {  
    services {  
      dhcp-local-server {  
        group ifs {  
          interface xe-2/0/0.1;  
        }  
      }  
    }  
  }  
  interface sp-0/0/0.1;  
  interface xe-2/0/0.1;  
  routing-options {  
    static {  
      route 0.0.0.0/0 next-hop sp-0/0/0.1;  
    }  
  }  
}
```

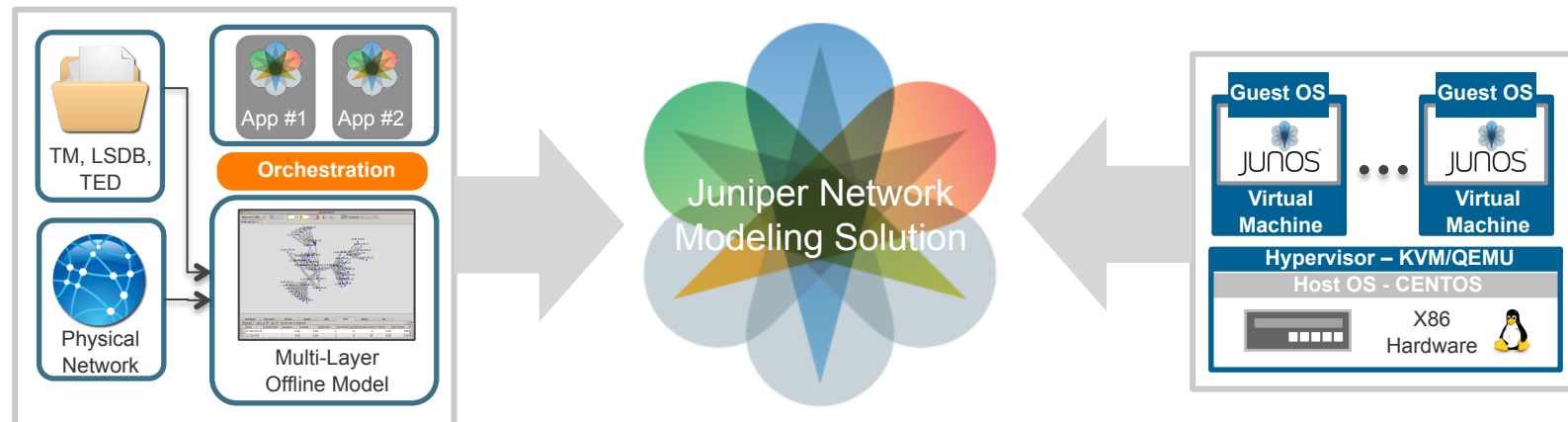
```
[edit interfaces]  
xe-2/0/0 {  
  vlan-tagging;  
  unit 1 {  
    vlan-id 1;  
    family inet {  
      address 192.168.1.1/24;  
    }  
  }  
  ...  
  sp-0/0/0 {  
    unit 1 {  
      family inet;  
      service-domain inside;  
    }  
    unit 101 {  
      family inet;  
      service-domain outside;  
    }  
  }  
}
```


NETWORK VIRTUALIZATION

On-demand Virtualization environment

- Enables real network design, testing and training
- Accessed on demand via the Internet
- Scale when you need it; pay for what you use

Design, test and learn with the flexibility, cost-efficiency & simplicity inherent of a cloud based delivery model



REFERENCES

Previous NANOG presentations:

- http://www.nanog.org/meetings/nanog47/presentations/Monday/RAS_Scriptrouter_N47_Mon.pdf
- http://www.nanog.org/meetings/nanog47/presentations/Monday/Shافر_Scriptrouter_N47_Mon.pdf

Juniper.net automation home:

- <http://www.juniper.net/us/en/community/junos/script-automation/>

Juniper contacts:

- Phil Shafer, phil@juniper.net
- Colby Barth, cbarth@juniper.net
- Roy Lee, roy@juniper.net

FIND US ONLINE



<http://www.juniper.net/jnet>



<http://www.juniper.net/facebook>



<http://www.juniper.net/youtube>



<http://www.juniper.net/twitter>



CONFIGURATION AUTOMATION

