

Migrating from CLI-driven to Model-driven Programmability



Nick Russo

NETWORK ENGINEER

@nickrusso42518 www.njrusmc.net



Agenda



NETCONF/YANG refresher

YANG visualization

- DevNet yangexplorer
- Advanced NETCONF Explorer (anx)

NETCONF on IOS-XE using ncclient

YANG Refresher

**Data modeling
language**

**Hierarchical
structure**

Many components



```
module interfaces {  
  container interface-cont {  
    list switchport-list {  
      key "name";  
      leaf name {  
        type string;  
      }  
      leaf vlan {  
        type int;  
        default 1;  
      }  
    }  
    list another-list {...}  
  }  
}
```

- ◀ YANG module definition
- ◀ Holds different items
- ◀ Key means "required"
- ◀ Names must be strings
- ◀ VLANs must be integers



NETCONF Refresher

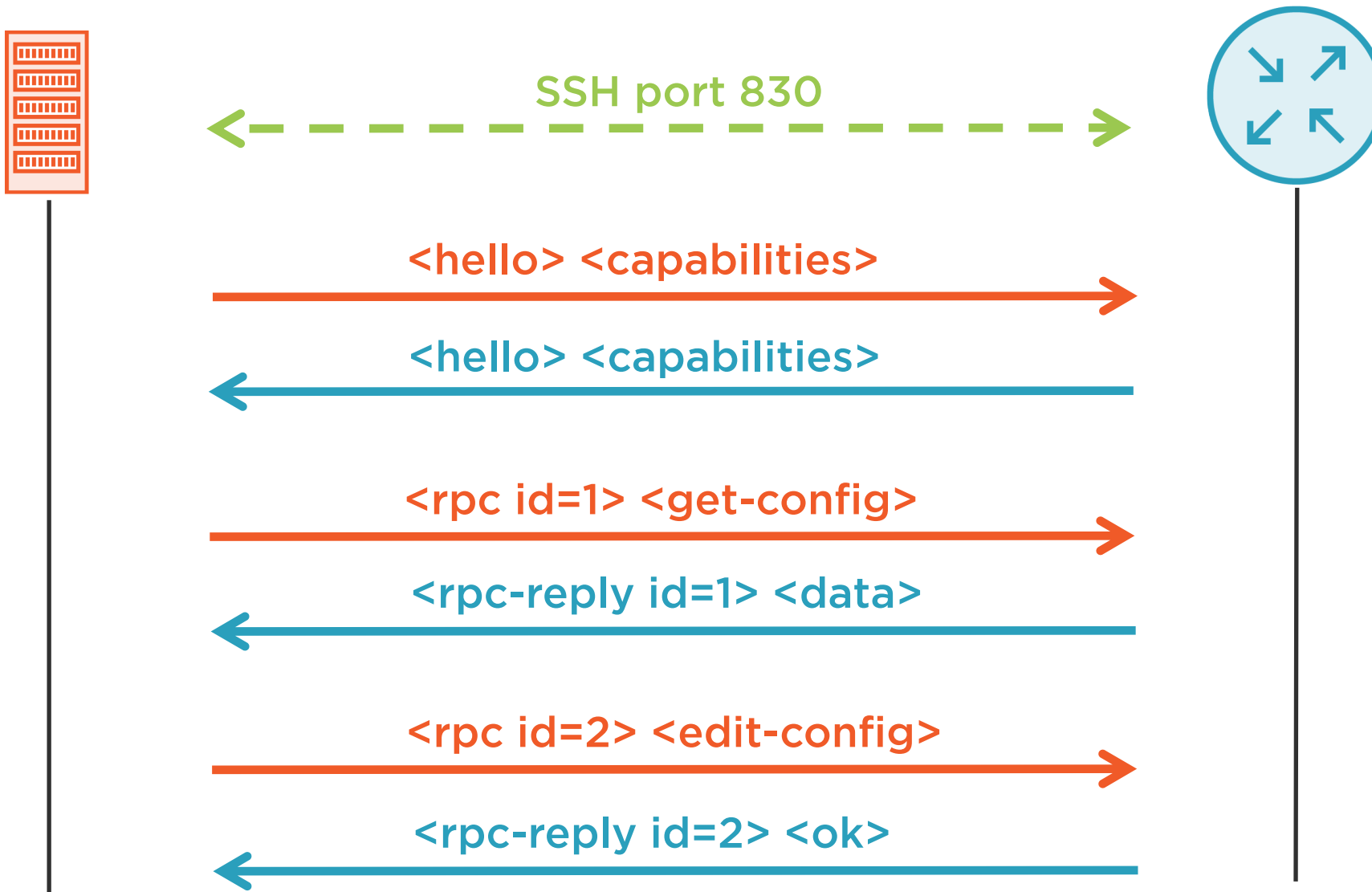
**Simplified
implementation**

Structured data

Easily idempotent



Introducing NETCONF



Demo



Using Cisco DevNet's yangexplorer



Demo



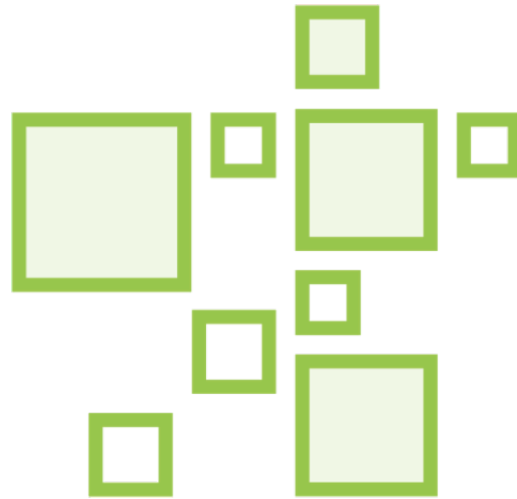
Using the Advanced NETCONF
Explorer (anx)



Common Migration Mistakes



Feature parity



Ubiquity



Framework
abstraction



Testing

Demo



Collecting network data using NETCONF



Demo



Migrating from CLI to NETCONF



Summary



Simplified operations, but harder to learn
"YANG first, management second"

Challenge:

- Combine Ansible and NETCONF
- "Automating Networks with Ansible the Right Way"