

ONAP CDS to Manage OLT Configuration

Tigo - Michel Franchesco
Aarna Networks - Vivekanandan M

 THE **LINUX** FOUNDATION

Agenda

- › Problem statement
- › Why we prefer ONAP ?
- › ONAP OLT Configuration Management Workflow
- › CDS Blueprint JSON
- › Demo

Problem Statement

- › With FTTx rollouts progressing, there is an increasing need to manage 10s or 100s of thousands of OLT switches from a centralized management platform
- › Current alternatives are not effective
 - › Manual steps: not scalable from an OPEX point of view, too much human resource.
 - › Inhouse management tool: don't work in a heterogeneous environment; it was deployed without software architectures' best practices.
 - › Custom software from an external provider: very expensive, all additional requirements depends for a provider.

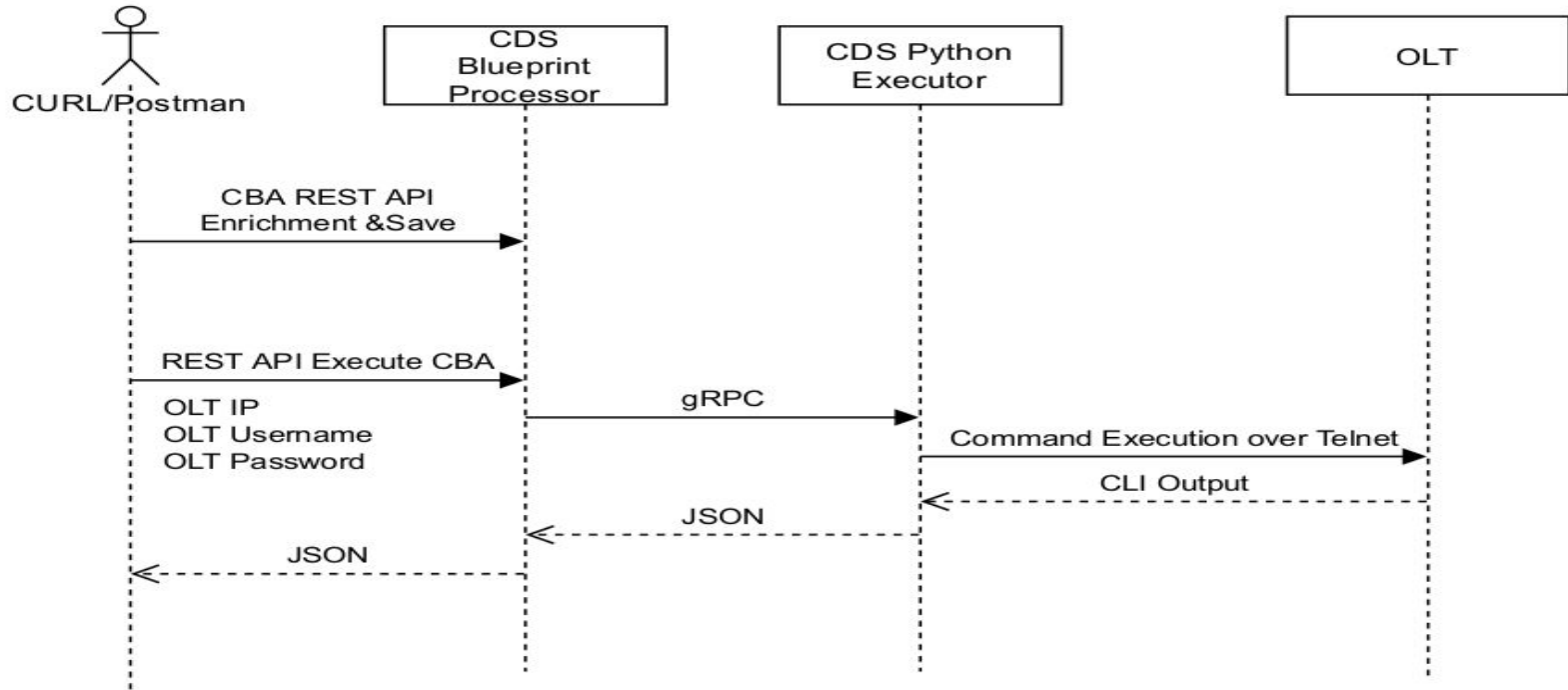
Solution

- › The solution is to utilize open source software
- › ONAP offers an elegant, **vendor-neutral** management framework. This allows one to build a centralized and scalable management solutions with multi-vendor **device discovery, monitoring, failure detection, automatic correction/recovery** of end to end services
- › ONAP CDS offers a very lightweight framework to interface with various south-bound protocols such as RESTconf, SSH, NETCONF, Telnet and SNMP

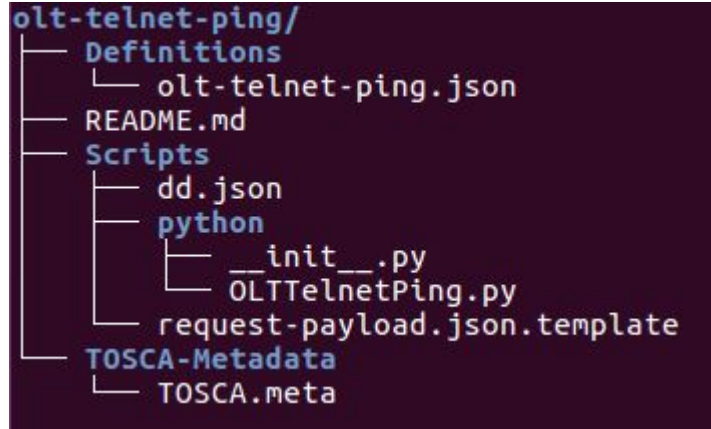
Why we chose ONAP ?

- Vendor-neutral, Open Source solution, with vibrant community support
- Easy to port our existing automation scripts
 - Legacy platform based python scripts
- Horizontal scalability
 - Configure 100s and 1000s of Physical Equipments (PEs or PNFs)
 - Multiple py-executor instances across regions
- Device Vendor specific CBA capabilities
 - Execute Configuration and Diagnostics tests
- Extensible to configure multi-vendor/multi-domain systems
 - Service Orchestrator (SO) Camunda workflows
 - End to End Service Provisioning

ONAP OLT Configuration Management Workflow



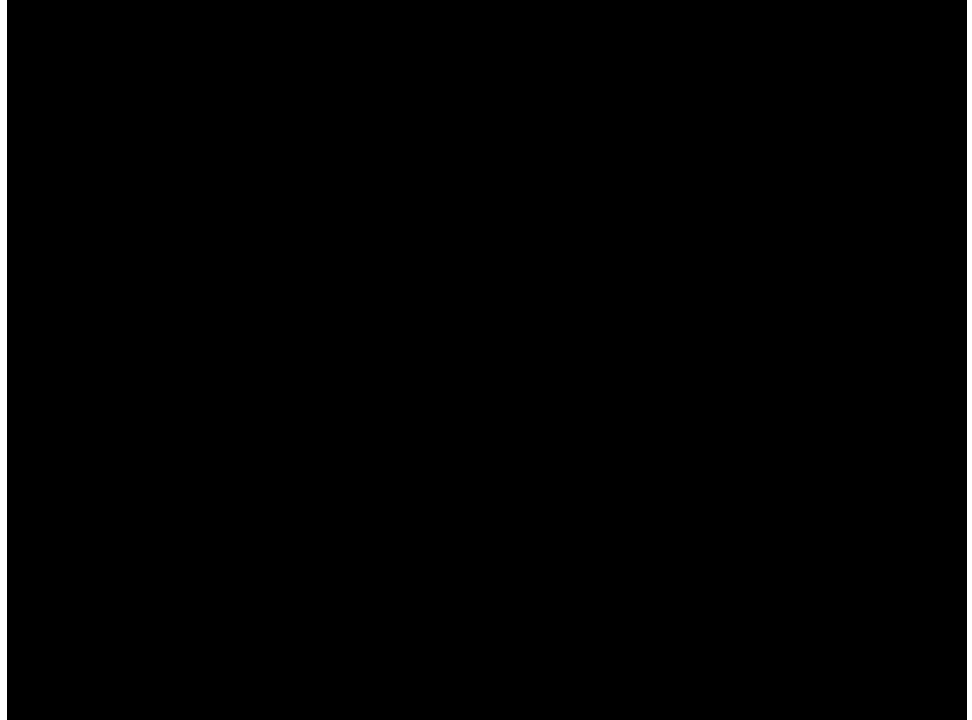
CDS Blueprint JSON



```
{  
  "metadata": { ...  
},  
  "dsl_definitions": {  
    "py-executor": { ...  
  },  
  "parameters": {  
    "olt-ip-address": { ...  
  },  
    "olt-user-name": { ...  
  },  
    "olt-password": { ...  
  }  
},  
  "topology_template": {  
    "workflows": {  
      "remote-python": { ...  
    }  
  },  
  "node_templates": {  
    "execute-remote-python": {  
      "type": "component-remote-script-executor",  
      "interfaces": {  
        "ComponentRemoteScriptExecutor": {  
          "operations": {  
            "process": { ...  
          }  
        }  
      }  
    }  
  }  
}
```

Demo

- › Recorded Video



Thank You!