Building Agile, Open-Source–Based Cloud Solutions With OpenDaylight

Colin Dixon
Distinguished Software Engineer, Brocade
Technical Steering Committee Chair, OpenDaylight
Abstract

OpenDaylight has become a nexus for open source project integration, creating the new open networking stack and enabling a new generation of open source, agile IT infrastructure. The fifth “Boron” release provides new tooling and documentation to support application developers, as well as greater integration with larger industry frameworks from OPNFV and OpenStack to CORD and Atrium Enterprise. Boron also brings a strong practical focus on two leading types of deployments: (1) direct control of virtual switches to provide network virtualization and NFV and (2) management and orchestration of existing networks to provide new features and automation.

This talk will cover trends in open SDN and cloud networking, with a focus on Boron milestones. In particular, it will focus on the architecture that is being developed across the OpenStack and OpenDaylight project to enable OpenStack service function chaining support in OpenDaylight controller.
Outline

1. OpenDaylight
2. What did we really want?
   1. Agility
   2. At all levels of the stack
3. Common use patterns
   1. OpenFlow + OVSDB
   2. NETCONF + BGP
We’ve been building SDN for years

- Commercially since at least 2012

- Why?

- Decouple elements of the network stack:
  - lower 成本 (cost)
  - faster innovation

- How are we doing?

敏捷 (Agility): quickly get new features with minimum dependence on others
How are we doing with SDN?

- Avoiding vendor lock-in
- Achieving a foothold in the market
- Faster innovation
- Interoperability and integration

You will notice that I haven’t mentioned cost yet...
Why Open Source?

• Avoid vendor lock-in
• Have a seat at the table
• Faster Innovation
• Interop & Integration

• You’ll note I didn’t say cost
Let’s focus on 成本 (agility)

- We need to be able to add features
- Anywhere in the stack
- Without help from others
- And deal with the consequences of the changes
The new open networking stack

- Networking Apps
- Mgmt / Analytics App
- ORCHESTRATION (NFVO, ...)

Rest API

SDN Platform

- OpenFlow
- Netconf
- OVSDB
- BGP
- Custom

YANG

Open NOS

- OpenFlow
- Netconf
- OVSDB
- BGP
- Custom

Legacy HW

Optical

VMs / Cont.
“OpenDaylight fundamentally changed the Linux Foundation’s world. It’s been wildly successful. It’s the *de facto* standard open source SDN controller for the industry today.”

- Dave Ward, Cisco CTO

- Mature, Open Governance
- 800+ Contributors
- Over 100 deployments
- Leading use cases identified
- Dozens of ODL-based solutions
- Mature code base
- Focus on performance, scale and extensibility

*SDxCentral, 9/7/16*
Boron Features and Capabilities

Integration - industry frameworks
• OPNFV
• OpenStack enhancement
• CORD/vCO
• ECOMP
• ONF/Atrium

Common SDN toolchains
Net Virtualization + SFC:
• OF + OVSDB + OVS/FD.io
Mgmt plane programmability:
• BGP + PCEP + MPLS + NETCONF

Operational tooling
• Cardinal health monitoring
• Data analytics (TSDR & Centinel)
• OCP (Open radio I/F)
• Documentation

App developer tooling
• YANG-IDE toolkit
• NetIDE for cross-OSS controller interoperability
• NeXt UI toolkit
• “Singleton app” HA
• Documentation
OpenStack/OpenDaylight Integration

- Multiple Neutron implementations
- Target different use cases, southbound drivers
  - FD.io/VPP
  - OVS
  - Open Overlay Router (née LISPmob)
- Provide distributed implementations of scalable network virtualization for OpenStack
OpenStack/OpenDaylight Integration

- L2: ML2 plugin
- L3: ODL L3 plugin
- services
  - FWaaS
  - L2Gateway
  - QoS
  - LBaaS
  - BGPVPN
  - networking-sfc
  - trunk

```
Neutron Server

<table>
<thead>
<tr>
<th>Type Manager</th>
<th>Mechanism Manager</th>
<th>ODL L3 Plugin</th>
<th>Service Plugins</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRE TypeDriver</td>
<td>...</td>
<td>ODL mech</td>
<td>FWaaS</td>
</tr>
<tr>
<td>VLAN TypeDriver</td>
<td>...</td>
<td>ODL driver</td>
<td>L2GW</td>
</tr>
<tr>
<td>VXLAN TypeDriver</td>
<td>...</td>
<td>ODL driver</td>
<td>QoS</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

networking-odl
```
OpenDaylight in vCO and ROBO
What does Central Office do?

- **Subscriber management capabilities**: Gateway, authentication and authorization, event and subscriber information logging
- **Optical Line Termination (OLT) for PON/GPON** (Passive Optical Net.)
- **Service functions**: self-service portals, NAT, FW, routing, IP addr mgmt, QoS, quotas, video caching, mail and file stores

**A Virtualized Central Office (vCO):**
- Uses general-purpose compute, storage and network capabilities to deliver the above services
- Added agility (spin up VMs vs. rack and stack hardware)
- Cost savings (via increased automation and commodity servers)
vCO Data Center Architecture

Physical elements are divided into

- Network: provides fabric/underlay
- Servers: provides computer/storage for VNFs
Controllers and orchestrators use overlay networks to form service chains of VNFs
vCO Data Center Software Architecture

**OSS/BSS**

- **Policy** (NIC, NEMO, GBP, Neutron)
- **VNFO** (ECOMP, Open-O, OSM, ...)
- **VNF Spec** (TOSCA)
- **VNF Catlog**
- **VNFM** (Tacker, Cloudify, ...)
- **VIM** (OpenStack, Kubernetes, ...)

**Fabric/Underlay (Network)**

- SDN Controller (OpenDaylight)

**Servers/VNFs (Compute, Storage)**

- Overlay Network
- Fabric
ROBO: Using vCO Blueprint in Enterprises

- vCO for Enterprises to provide for Remote/Branch offices
  - Maybe offered by ISPs as a service
  - Integrating with public cloud will likely involved some form of vCO (either aaS or Enterprise-deployed)
  - Hybrid Cloud will almost certainly involve vCO

Automating this requires integration with many other projects and legacy devices (NETCONF + BPG + MPLS + PCEP)
Think Beyond the Controller

Product

Enabling solution component
Thank you