ODTN and TIP Collaboration with Whitebox Transponder ‘Cassini’

2018.12.5
Hiroki Okui
NTT Communications
Disaggregated Transport Network

- Proprietary and closed API
- Vendor-specific data model

- Vertically integrated
- Single vendor

- Open and standard API
- Common data models

- Multi vendor
- Disaggregated
ODTN (Open Disaggregated Transport Network)

- Optical telemetry
- Power Management
- Protection/restoration
- Calendaring

ODTN Controller

- TAPI
- OpenConfig

Open Line System (OLS)

- Transponders from multiple vendors
- Book-ended transponders

WA

Edge Cloud

Onos
Open Line Systems

- Traditional optical line systems are integrated systems with a single vendor’s transponder, mux/demux, amp, ROADM

- Open Line Systems are disaggregated systems composed of multi-vendor transponders
- Possible to use preferred vendor’s transponder every time of wavelength expansion
Cassini & TAI
Whitebox packet transponder “Cassini”

- Broadcom Tomahawk+ ASIC (3.2T)
- 100Gbit/s QSFP28 x16
- 200Gbit/s CFP2-ACO x8
What is TAI?

- TAI is an interface between optical transponders and system software
- Allows system software to operate with any TAI-compliant transponders
- Allows transponders to operate in any system which supports TAI
- By decoupling the transponders from the rest of the system, it allows each to innovate independently
- Available here:
  - https://github.com/Telecominfraproject/oopt-tai
  - https://github.com/Telecominfraproject/oopt-tai-implementations
What is not TAI?

- TAI is not an API for operators like YANG models
- TAI is not trying to become a de jure standard or standardization body
Collaboration with ODTN

- Provision through OpenConfig common model
- IP Infusion provides OpenConfig NBI interface to us with OcNOS (Thanks!)
Development & Test
Device setup and TAPI representation

Cassini

transceiver (QSFP28) 逻辑-channel  transceiver (CFP2_ACO)

TAPI

Connectivity Service

OTSi layer (ignored)
Mapping from TAPI to OpenConfig

```
<logical-channels>
  <channel>
    <logical-channel-assignments>
      <assignment>
        <index>10101</index>
        <config>
          <index>10101</index>
          <assignment-type>LOGICAL_CHANNEL</assignment-type>
          <logical-channel>20101</logical-channel>
          <allocation>100.0</allocation>
        </config>
      </assignment>
    </logical-channel-assignments>
  </channel>
</logical-channels>
```

```
tapi-sample-step2-intermediate.xml

<connection xmlns="urn:onf:otcc:yang:tapi-connectivity">
  <uuid>00000000-0000-3000-0001-111000000000</uuid>
  <connection-end-point>
    <topology-id>...-100000000000</topology-id>
    <node-id>...-100000000000</node-id>
    <owned-node-edge-point-id>...-121000000000</owned-node-edge-point-id>
    <connection-end-point-id>...-121000000000</connection-end-point-id>
  </connection-end-point>
  <connection-end-point>
    <topology-id>...-100000000000</topology-id>
    <node-id>...-100000000000</node-id>
    <owned-node-edge-point-id>...-111000000000</owned-node-edge-point-id>
    <connection-end-point-id>...-111000000000</connection-end-point-id>
  </connection-end-point>
  <layer-protocol-name>DSR</layer-protocol-name>
</connection>
```

```
sbi-openconfig-sample-infinera.xml

<connection xmlns="urn:onf:otcc:yang:tapi-connectivity">
  <uuid>00000000-0000-3000-0001-111000000000</uuid>
  <connection-end-point>
    <topology-id>...-100000000000</topology-id>
    <node-id>...-100000000000</node-id>
    <owned-node-edge-point-id>...-121000000000</owned-node-edge-point-id>
    <connection-end-point-id>...-121000000000</connection-end-point-id>
  </connection-end-point>
  <connection-end-point>
    <topology-id>...-100000000000</topology-id>
    <node-id>...-100000000000</node-id>
    <owned-node-edge-point-id>...-111000000000</owned-node-edge-point-id>
    <connection-end-point-id>...-111000000000</connection-end-point-id>
  </connection-end-point>
  <layer-protocol-name>DSR</layer-protocol-name>
</connection>
```
Model-driven controller in ONOS: DCS

- Subsystem to support NETCONF/YANG ecosystem
- Launched in 2016 and has been developed to realize model-driven ctrl.
ODTN Implementation

Framework:
- ONOS YANG compiler, runtime
- Dynamic config subsystems

Features:
- NBI(RESTCONF) auto-generation
- SBI(NETCONF) auto-generation
- Java library which enable easy implementation of Service Application
- Distributed config store of NBI service configuration and device configuration
Test Result

• Made all component work together successfully
  • Mapping between OpenAPIs: TAPI => OpenConfig => TAI
  • Got confidence that Cassini/OcNOS/TAI are promising devices/NOS/API as transport whitebox

• Still work in progress
  • Some features of TAI are not implemented as of now
  • Exposed configurations are limited as well
  • Will be able to have a full capability of TAI in a year

• Some critical issues are found in DCS
  • Should be addressed from its design
  • Good requirement for next DCS
Takeaways

• ODTN and TIP collaboration
• ODTN and Cassini is a good starting point to realize whitebox transponder/controller
• Come and join us!
Thank you
Backup Slides
Our Expectation for Disaggregated Transport Networks

- Flexibility and agility
  - Integration from “in hardware” to “in software”
  - Innovations and upgrades from “per all-in-one nodes” to “per each component”

- Target Domain
  - metropolitan
  - DC/Cloud Interconnect
Open Communities Efforts

Open Line Systems

OpenConfig

Transport API

http://www.openconfig.net/projects/models/


http://www.opennetworking.org/open-transport/

Copyright © NTT Communications Corporation. All rights reserved.
Towards Full Open Architecture

- Existing communities are focused on each specific target
- No “Integrated Solution” in open source community

→ Build a reference implementation by using those communities outputs
ODTN (Open Disaggregated Transport Network)

- Optical telemetry
- Power Management
- Protection/restoration
- Calendaring

ODTN Controller

- TAPI
- OpenConfig

Open Line System (OLS)

- MUX
- WSS
- AMP
- WSS
- MUX

Transponders from multiple vendors

Book-ended transponders

Edge Cloud

WA N

Copyright © NTT Communications Corporation. All rights reserved.
ODTN Members

- 5 operators
  - China unicom 中国联通
  - COMCAST
  - NTT Communications
  - Telefonica
  - TIM

- 12 vendors
  - ciena
  - Edge-core
  - NOKIA
  - NEC
  - ZTE
  - OPLINK
  - infinera
  - LUMENTUM
  - ADVA
  - CoAdna
  - Juniper
  - Coriant
Current progress and next step

- **Current progress**
  - Implementation and testing for Transponder provisioning with OpenConfig: Done
  - Design OLS and optical media layer provisioning with latest TAPI and OpenConfig: On going

- **Next step**
  - Implement path and config computation feature with leveraging onos optical-intent
  - Design mesh solution towards Phase 2.0

---|---|---
Phase 1.0 | **We are here** | Phase 2.0
P2P, only transponder | P2P, transponder + OLS | mesh, transponder + OLS

Phase 1.5

P2P, transponder + OLS

Jan. 2018
Challenges

- The journey to Software Integration of multi-vendor dis-aggregated devices is long and difficult
  - Lots of features to be realized among multi-vendor devices
  - Discovery, path computation, power control, protection, monitoring, etc..
- Common Open API is needed
  - TAPI is the most possible candidate, but there are some missing parts from the software integration perspective
  - ODTN is collaborating with OTCC/TAPI and growing into each other
- Multi-device transaction and config state management features are needed
  - But there are no candidates in current Open SDN controllers
  - Now considering to implement these features in ONOS