



Taking SEBA into Production

NETSIA

Requirement Analysis & Design

- Requirement preparation
 - Business and Technical
- ONF's VOLTHA and SEBA study
- GAP Analysis
- Design

<input type="checkbox"/>	SPON-R1 (1)	TR10110 – Supported Line Termination Technologies	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R2 (1)	TR10120 – Multi T-CONT support	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R3 (1)	TR10130 – VLAN Models	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R4 (1)	TR10140 – QoS Profile Management for OLT	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R5 (1)	TR10145 – QoS Profile Provisioning	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R6 (1)	TR10150 – QoS Profile Management for ONT	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R7 (1)	TR10155 – QoS Profile Provisioning for ONT	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R8 (1)	TR10160 – PPPoE support	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R9 (1)	TR10170 – IPTV Multicast support	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R10 (1)	TR10180 – DHCP-L2 support	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R11 (1)	TR10190 – Alarming	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R12 (1)	TR10200 – Statistics and Counters	---	OLT Hardware Abstraction
<input type="checkbox"/>	SPON-R13 (1)	TR15110 – Subscriber Caching Application	---	ONOS Application
<input type="checkbox"/>	SPON-R14 (1)	TR15111 – Subscriber Caching configuration	---	ONOS Application
<input type="checkbox"/>	SPON-R15 (1)	TR15112 – Subscriber Caching time-to-live	---	ONOS Application
<input type="checkbox"/>	SPON-R16 (1)	TR15120 – Additional Caching data for Device and Subscriber	---	ONOS Application
<input type="checkbox"/>	SPON-R17 (1)	TR15130 – QoS caching for Subscriber	---	ONOS Application
<input type="checkbox"/>	SPON-R18 (1)	TR15140 – Integration of Caching Application to Platform API layer	---	ONOS Application
<input type="checkbox"/>	SPON-R19 (1)	TR15145 – Cache Persistency	---	ONOS Application
<input type="checkbox"/>	SPON-R20 (1)	TR15150 – vOLT Application	---	ONOS Application
<input type="checkbox"/>	SPON-R21 (1)	TR15160 – PPPoE Flow management	---	ONOS Application
<input type="checkbox"/>	SPON-R22 (1)	TR15170 – PPPoE Flow management configuration	---	ONOS Application
<input type="checkbox"/>	SPON-R23 (1)	TR15180 – Segment Routing Application	---	ONOS Application
<input type="checkbox"/>	SPON-R24 (1)	TR15190 – DHCP L2 Relay Application	---	ONOS Application

Things to Do

➤ New Features

- Service related features(TT Workflow)
- Northbound APIs
- UI
- FCAPS

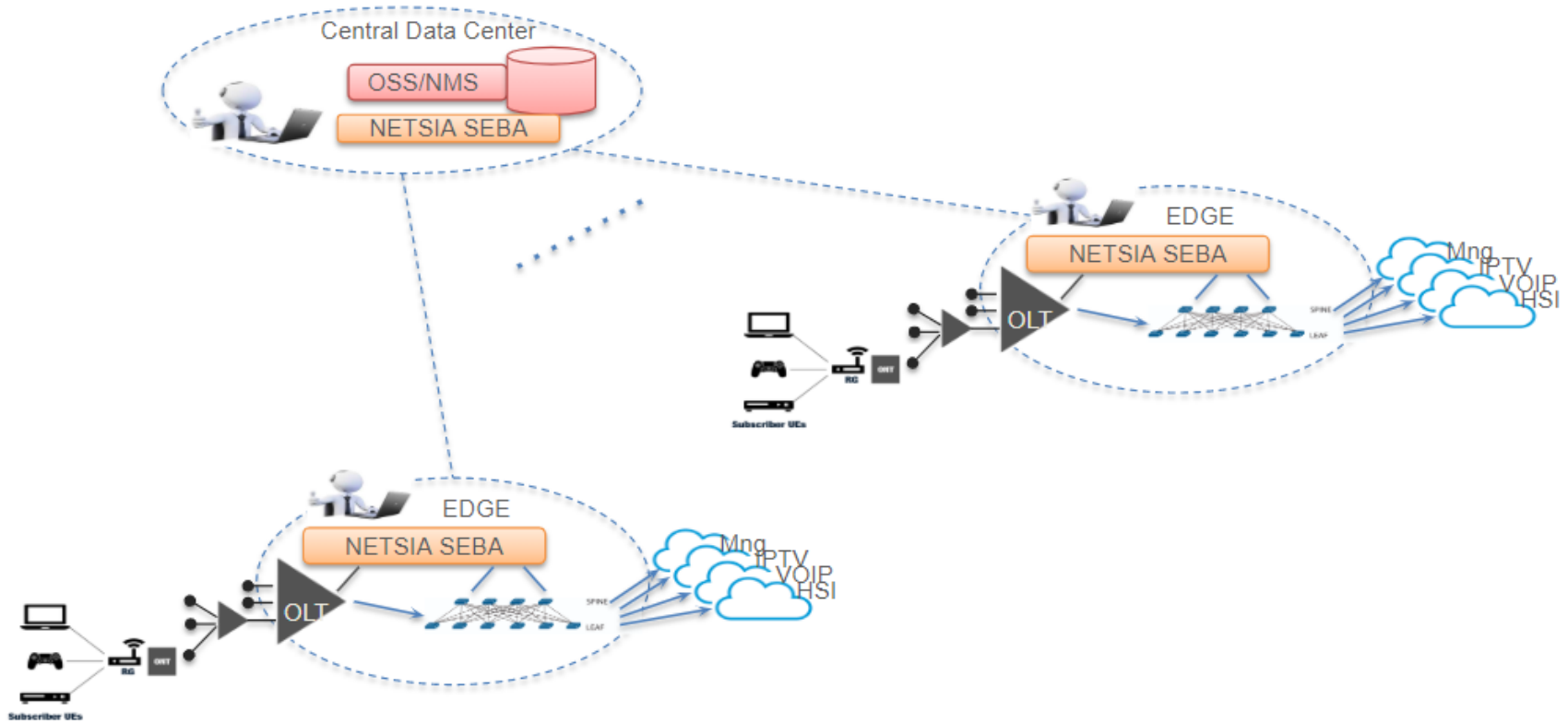
➤ Network Integration

- Provisioning
- PM
- Alarm

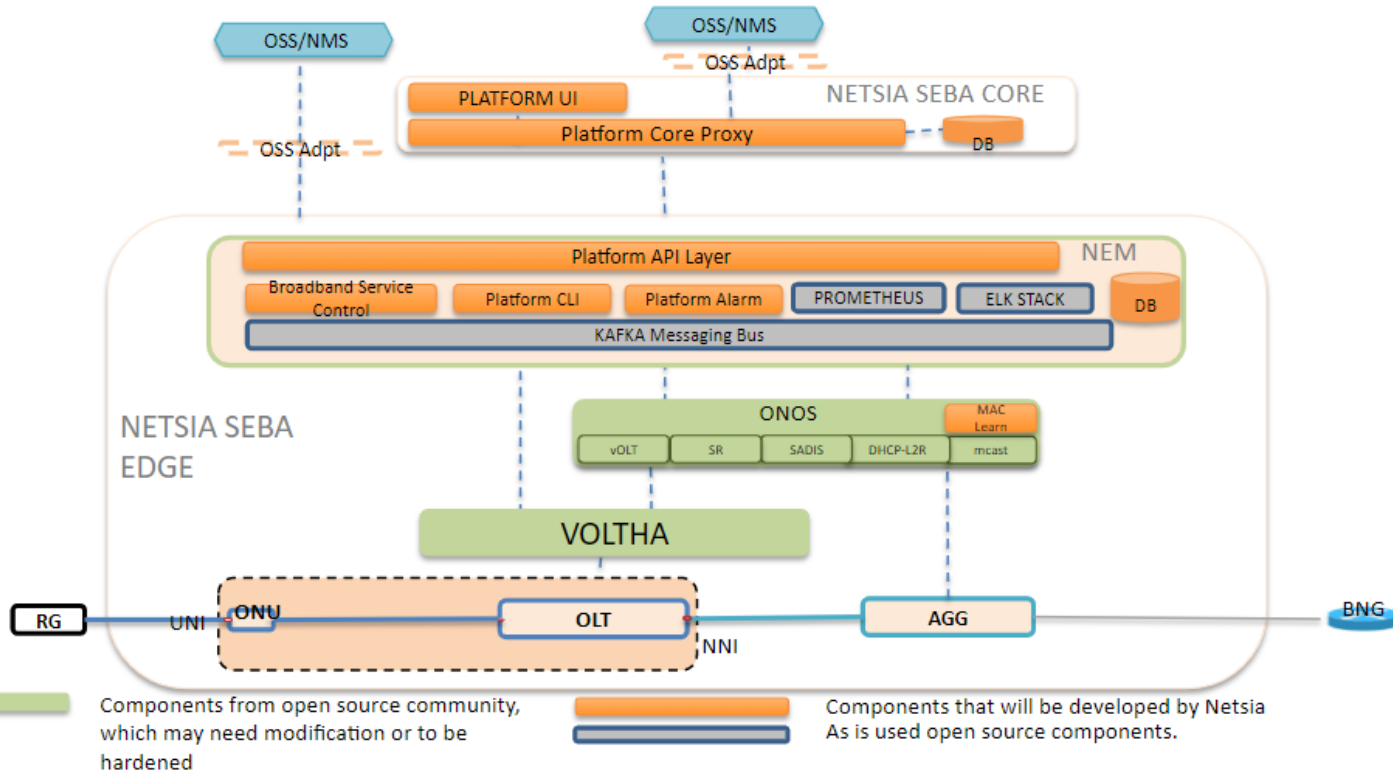
➤ Hardening

- Scaling services for n ONUs bringup scenario
- compute nodes/micro-services restarts
- add/remove, reboot, enable/disable devices
- Backup/Restore

Decided Architecture



Decided Architecture



Development Activities

- Voltha 1.x
 - Technology profile Voltha 1.6
 - Multi t-cont, VLAN translation
 - Multicast and BAL brigades
- Voltha 2.x
 - Performance & Scalability tests
 - Containerization brigade
 - Stabilization brigade
 - BBSIM
- Netsia Components
 - Service Control
 - O&M Modules
 - APIs
 - User Interface

Test Activities

- Test Case Documentation
- Automated tests started by Jenkins
 - **Functionality**
- Scalability and performance tests with BBSIM (South-to-north)
 - **OMCI Initialization**
 - rw_core performance
 - onu-adapter performance
 - olt-adapter performance
 - **DHCP and authentication**
- DHCP traffic tests with tcpreplay (South-to-north)
- ONOS performance on Rest APIs (North-to-south)
- Failover tests

CI/CD Activities

- For CI operations in our development environment
 - Periodic Jenkins Jobs are used
- For CD operations
 - Jenkins
 - Test automation frameworks(JUnit, Robot Framework)
 - Binary repository manager(Nexus 3)
 - Docker, Kubernetes
 - Custom developed tools/scripts
- Additional tools
 - Bitbucket for source code management
 - Jira issue tracking system
 - SonarQube for code quality

Process

- Created a mirror of all upstream projects under Bitbucket
- Periodic(three times a day) sync with Jenkins jobs
- Netsia repos:
 - Netsia specific components
- Forks from mirror for upstream projects
- Mid-day sync from fork to development branches.
- Maintain a GitLab for 3rd party dependencies.
- Build code and upload the artifacts to Nexus 3 by Jenkins jobs.
- Use artifacts in Nexus 3 during deployments.

Multi T-CONT

- TT workflow requires each service has different T-CONTs
- High complexity
 - For each subscriber service BW limitation might be different
 - e.g. Gold/Bronze HSI service
 - pbit marking, VLAN translation, MAC learning
- It was too difficult to debug
 - No clue to understand what is NOT working
 - No way to debug at the HW level
 - Logs are not very informative

IPTV Service

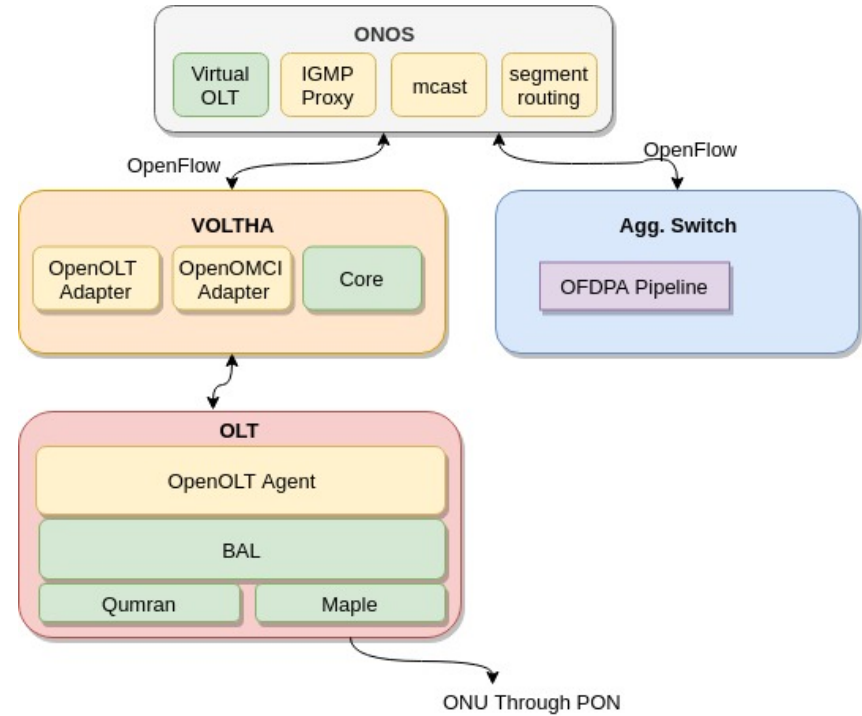
- Video On Demand & Multicast
 - Both uses the same T-CONT thus same Technology ID
 - Changes required on SADIS, ONOS vOLT
 - TP has two GEMs
 - Need VOD, IGMP Trap and IPTV DHCP flows to be defined
 - VOD flow is not required in case of MAC learning

Multicast

- Multicast brigade
 - Netsia ONF Sterlite Infosys
- Implementation has been completed on Voltha 1.7. Community reviews are ongoing.
 - Jira Epic is “Enabling multicast functionality in VOLTHA”
- Multicast has been implemented along with the Multi T-Cont
 - Design was dependent on the Multi T-Cont and BAL traffic model and it’s limitations.
 - No support for destination IP address - IP to MAC conversion on adapter
- IGMPv2 done, but TT requires also v2 and v3 as hybrid.

Multicast

- ONOS apps has been updated to work with the ONOS 1.13.x
- New Multicast Store usage
- OLT's to manage by mcast and fabric to manage by segment routing



VLAN Translation

- TT workflow requires VLAN translation at the ONU level
 - SADIS, vOLT, OpenOMCI adaptor
 - Extended VLAN Tagging Operation at OMCI
 - OF worked well as NB API
- Not open-sourced yet
 - It's hard to implement this without affecting other workflows

BAL

- BAL 3.x provides the flexibility changing packet pbits
 - mapping to queues by pbits - not tested yet!
- BAL 2.6 multicast limitations
 - same V-LAN for different multicast groups - supported by BAL 3.1(patch) & BAL 3.2
- BAL 2.6 is re-architected, must use BAL 3.x for production
 - 3.0 is an early release, 3.1 and later stable releases



Thank You

Follow Up Links:

XXXX