NEM: Overview and ISSU plans
Overview

AbstractOLT

Network Edge Mediator (NEM)

ONOS

VOLTHA

OLT

ONU

UNI

NNI

AGG

Compute

BNG

KAFKA

XOS

LOGGING (elk stack)

MONITORING (prometheus)
XOS Architecture

- **Northbound Interface**
  - xos-gui
    - xos-tosca
    - chameleon
    - xos-core
    - synchronizers

- **Core**
  - kafka
    - xos-db

- **Southbound Interface**
  - VOLTHA
  - ONOS

- **Backend**
  - VNFs
  - xos-ws
  - chameleon
XOS in SEBA

An example operation, OLT provisioning.

Models (OLTDevice) -> vOLT Synchronizer

- Pre-provision
- Activate
- Wait for activation

VOLTHA

NB Interface
XOS in SEBA

An example operation, Subscriber authentication.

Workflow -> Subscriber -> vOLT -> Crossconnect

Workflow Synchronizer -> KAFKA

VOLT Synchronizer

Fabric-xconnect Synchronizer

ONOS -> VOLTHA -> ONOS
The synchronizer framework allows XOS to be extended in service-specific ways.

- Service-specific models
- Service-specific business logic
- Abstractions and logic that span multiple services

XOS supports diverse heterogeneous services. Different kinds of services naturally need different models and logic.
Synchronizers specify models, and implement policies and steps
Types of Steps

- XOS -> External Component
  - Sync Step
  - Delete Step
- External Component -> XOS
  - Pull Step
  - Event Step
- XOS -> XOS
  - Model Policy
Synchronizers: moving to a library

The synchronizer framework was refactored as a python library.

- **Developer benefits**
  - Compliant with python best-practices
  - Developer friendly (IDEs)

- **Community benefits**
  - Ease of re-use promotes adoption

- **Operational benefits**
  - De-layering of containers -> Smaller containers
Anytime a model evolves actions needs to be take, mainly:

- Bring the database schema up to date
- Make sure data are kept in a consistent state

Best practices:

- Migrations are treated as code
- Migrations can be executed both ways
Migrations: example

Model v1:
- string firstName
- string lastName

Model v1.1:
- string firstName
- string lastName
- string fullName

Model v2.0:
- string fullName

A field is added (autogenerated)
Data are changed (custom logic)
Migrations: XOS

xos-migrate: https://guide.opencord.org/xos/dev/xosmigrate.html

- Generate standard migrations base on xProto changes
- Allow developers to extend migrations with custom logic
Migrations: XOS

CORE

Models v1 → Models v2

Models v1

Migrations

xProto

Synchronizer v1

Synchronizer v2

Migrations

xProto

DB