ETSII ENI Technology Workshop on 25 September 2020
The relation of ENI policy management to network intelligence

Relation to Open Source MANO (OSM)

Francisco Javier Ramón (Telefonica)
Subhankar Pal (Altran)
Open Source MANO is an ETSI-hosted project developing an Open Source NFV Management and Orchestration (MANO) software stack aligned with ETSI NFV.
OSM Architectural Principles

Layering

Abstraction

Modularity

Simplicity

Architectural Principles
OSM’s Approach Aims To Minimize Integration Efforts

- A well-known **Information Model (IM)**, aligned with ETSI NFV, that is capable of modelling and automating the full lifecycle of Network Functions.

- A unified **northbound interface (NBI)**, based on NFV SOL005

- The **extended concept of “Network Service”** in OSM, so that an NS can span across the different domains identified and therefore control the full lifecycle of an NS interacting with VNFs, PNFs and HNFs.

- The lifecycle management of **Network Slices**, assuming if required the role of Slice Manager, or integrating with an external Slice Manager
OSM Architecture Overview

### Common Services
- OSM IM
- Common Database (NoSQL)
- Object Storage
- TSDB (Metrics)
- Auth

### Integrated Components
1. Unified message bus for async communications
2. Integrated components for placement, policy, fault and performance management
3. VCA controller for Generic NF configuration & indicator management
4. End to end orchestrator: LifeCycle Management (LCM) component
5. Unified Northbound Interface
6. Complete control through CLI and stand-alone UI

### Key Features
- osmclient
- light-ui
- ng-ui
- OSM IM
- NBI
- LCM
- VCA
- RO
- PLA
- POL
- MON

© ETSI 2020
Network Intelligence Maturity Model

- Degree of Automation
- Business Impact & Productivity
- People Driven
- KPI & Rules Driven
- Predictive
- Proactive Automation
- Open Loop Automation
- Closed Loop Automation
- Human Machine Interface
- Autonomous Network
- OSM Rel 8
Current State of OSM Service Assurance

MON
Covers the exporters based monitoring, with a solid architecture to expand them to more exporters.

- Static thresholds for alerting policies
- No correlation of metrics

POL
Designed around basic auto scaling & alerting

- No correlation of alerts and events

Dashboard
Automated dashboards for monitoring VNF timeseries metrics using Grafana

- Log analytics using Elastic stack

Placement
Placement optimization of network services
Closed Loop Automation Vision

Closed loop automation drives autonomous network.

1. Observe
Collect network metrics through different telemetry interfaces.

2. Decide
Processes collected metrics to determine the network status, decides action to be taken based on network policies. This phase is not responsible for executing the action.

3. Act
Acts upon orchestrated object and implements given lifecycle action.
Self Learning Through Closed Loop Automation

1A. Append to Historical Dataset
1B. Train ML Models
1C. Validate ML Models
2. Predict
3. Prescribe
4. Act

- Business Intents
- Update Scorecard
- Calculate Reward
- Get Previous Action’s Outcome
- Action Scoring
- Secondary Feedback Loop
- Prescribe New Action
- Primary Feedback Loop
- Network Intents
- Resource Orchestrator
- Realtime & Forecasted Network Anomalies
- Trained Models
- Network Data
- Realtime Network Data
- Model Training
- Secondary Feedback Loop
- Network Data

© ETSI 2020

Iterative process

1. Observe

2. Define and train AI model

3. Provide Forecasts

4. Act (scaling operations)

Define and train AI model

Training Stage

Get NS metrics

Production Stage

AI-Agents

NFVI

AI-Models Server Framework

New metrics insights

Model validation

Model optimization

Model update
Predictive Threshold

Feature Description: https://osm.etsi.org/gerrit/#/c/osm/Features/+9178/

Why it means?

• Metrics will be forecasted in future using timeseries forecasting algorithm.
• Threshold will also be applicable in forecasted period
• VNFD will define how much forecasted value will be considered as valid for prediction

Why we care?

• Move away from reactive to proactive
• Take action before actual problem occurs.
Dynamic Thresholds

Step 1
Model the normal behavior of the metric(s) using a statistical model (e.g. trend and seasonality).

Step 2
Devise a statistical test to determine if samples are explained by the model.

Step 3
Apply the test for each sample. Flag as anomaly if it does not pass the test.
Integrating ML in OSM – Implementation Options.
Auto Healing

**Feature Description**: https://osm.etsi.org/gerrit/#/c/osm/Features/+/9617/

**What it means?**
- Automatically heal the VNF when it fails. The recovery policy specified during deployment controls the recovery.

**Why we care?**
- Act on and fix unresponsive or dead VDUs
- High availability of network service

**How will we implement?**
- Automatically heal the VNF when it fails. The recovery policy specified during deployment controls the recovery.
- Metrics –
  - New Metrics Type — ICMP PING — first attempt to reboot the affected VDU; if this fails, then it attempts to redeploy the affected VNFD (on the same host)
  - Polling Frequency — <e.g. polling frequency of 10 seconds>
  - Existing Metrics — OSM_VM_STATUS, OSM_VIM_STATUS
- Recovery Actions -
  - REBOOT_THEN_REDEPLOY — first attempt to reboot the affected VDU; if this fails, then it attempts to redeploy the affected VNFD (on the same host)
  - REBOOT_ONLY — only attempt to reboot the VM
  - REDEPLOY_ONLY — only attempt to redeploy the VM
What are Next Steps?

**MON**

- Advance monitoring using **streaming telemetry** for e.g. gRPC and gNMI
- Replace threshold based alerts with **ML based advanced anomaly detections.**
- ML based **predictive thresholds**

**POL**

- **Alert correlation and alarm consolidation**
- **Recommendation engine**

**Dashboard**

- **Visualize anomaly points in Grafana charts**
- **Visualize timeseries predictions and thresholds**

**Placement**

- Intelligent placement as recommended by POL
Thanks!