INTRODUCING
ETSI OPEN SOURCE MANO (OSM)

Francisco-Javier Ramón Salguero (Telefónica)
ETSI OSM Chair

© ETSI 2019
NFV promises to go from traditional network management...

**Day 0**
- PNF installation
- Initial configuration to make PNF reachable (user, pwd, network, etc.)

**Day 1**
- License activation
- Injection of configuration
- Neighbor configuration
- Network configuration

**Day 2**
- Service provisioning
- Business provisioning

EMS

OSS

BSS
... to native NFV management, with highly efficient automation and operation.
Leveraging replaceable components that can be safely & automatically assembled…
Understanding the ACTUAL journey
Operating a real Virtualised Network looks more like this, with multiple sites and technologies...

(*) Topologies and combinations of technologies are provided as examples
... and unless we are ready to respect the layering, management gets really complicated.
... and unless we are ready to respect the layering, management gets really complicated
... and unless we are ready to respect the layering, management gets really complicated
Modelling is not helping either…

INADEQUATE VNF MODELLING

HARD ONBOARDING

UNEVEN VNF CATALOGUE

Basic NSD

Ad hoc integration often needed

BASIC AND HAND-MADE NETWORK SERVICE
while many technologies & IFs evolve independently from SDOs

NFVI, VIM, and App evolution are highly independent from specs...
... while many technologies & IFs evolve independently from SDOs

... impacting also some of the interfaces...

App & VNF technologies

- Ansible
- YANG
- Expect
- VNF A
- VNF B
- VNF C
- Paravirt
- SR-I0V
- Pass Through

Hypervisor, VIM & SDN technologies

- OpenStack
- Cloud Director
- AWS
- KVM
- ESX
- XEN
- Haswell
- Broadwell
- Skylake

© ETSI 2019
... while many technologies & IFs evolve independently from SDOs

And changes in both of them usually require changes in the IM for effective use...

How can we keep up evolving the IM at the pace of cloud evolution?
On the other hand, there is no such a thing as IM convergence out there.
On the other hand, there is no such a thing as IM convergence out there.
On the other hand, there is no such a thing as IM convergence out there

THE SERVICE PROVIDER NIGHTMARE:

- Different IMs ➞ **NO INTEROPERABILITY**  
  (NO COMPLIANCE EITHER)
- Unclear how to progress IM
  - Fix gaps
  - Model new technologies

THE VENDOR NIRVANA:

- **Compliance** used to pretend interoperability
- Used as tag, **allows to fake compliance**
Some **common mistakes**... disguised as shortcuts

**VERTICAL INTEGRATION**
- Same vendor: Infrastructure + MANO + VNFs
- Not an option for most realistic scenarios!
  - E.g. Slicing: RAN+MME+S/P-GW+HSS+PCRF...

**INCONSISTENT LIFECYCLE**
- Disconnection of resource and SW workflows
- Per VNF management (instead of NS or Slice management)

**FAILURE TO RESPECT THE LAYERING**
- Leads to **many hidden manual operations!**
“In theory, there is no difference between theory and practice. But, in practice, there is”

(Benjamin Brewster)
Why Open Source MANO for IM progression?
Many technologies & IFs evolve independently from NFV specs

OSM is really well suited for non-ambiguous evolution & testing...
... as IM changes are immediately percolated E2E and are tied with functionality
In this period, OSM Community has grown to 118 members!

• 14 Global Service Providers
• Leading IT/Cloud players
• VNF providers

(*) Names & brands may be claimed as the property of others
... and open to more fellow travellers, with REALLY LOW BARRIERS FOR PARTICIPATION

ETSI MEMBERS
- Sign Member Agreement & CCLA
- Free participation

NON-ETSI MEMBERS
- Sign Participant Agreement & CCLA
- Fees per F2F meeting (same as in ETSI NFV)

Individual developers and end users
- Just create an individual account

MORE INFO AT: osm.etsi.org
Key is INTEROPERABILITY, allowing architectural alternatives and competition.
SO, WHAT DOES OSM PROVIDE IN PRACTICE?
OSM provides a production-quality MANO stack...

- Capable of consuming openly published IM/DM
- Available for everyone, to minimize uncertainties
- Suitable for all VNFs, capturing real production complexity
- Operationally significant: including Service Orchestration too!
- VIM-independent, WAN-independent

ALIGNED TO NFV ISG INFORMATION MODELS
- ... but capable of providing prompt and constructive feedback whenever needed

ENABLING AN ECO-SYSTEM OF IM-COMPLIANT VNF VENDORS
- Ready to be offered to cloud and service providers
- No need of integration per- customer & MANO vendor basis
OSM is Model-centric. Modelling is key to hide internal complexity and simplify/automate daily operations.

OSM modelling is complete, self-contained and agnostic to technology infrastructure.

OSM allows modelling three different types of entities:

- **NF Packages (VNF, PNF, HNF)**
  - Mgmt Procedures
  - Resource Description
- **NS Package**
  - Mgmt Procedures
  - Topology
- **Network Slice Package**
  - Mgmt Procedures
  - Topology

All the life-cycle management aspects can be modeled: creation, initial configuration and further operations (reconfiguration, scaling, maintenance, etc.)
OSM is Model-centric. Modelling is key to hide internal complexity and simplify/automate daily operations.

OSM modelling is complete, self-contained and agnostic to technology infrastructure.

OSM allows modelling three different types of entities:

All the life-cycle management aspects can be modeled: creation, initial configuration and further operations (reconfiguration, scaling, maintenance, etc.).

NF Packages are preserved across all the chain and all the customers.
The IM feeds OSM behaviour, independently of the specific technologies southbound...
... so that a single OSM NS/Slice can span across different types of VIM+NFVI, Transports, and Physical Functions...

OSS/BSS

SOL005 + NS LCM calls

E2E Service Orchestration

IM integrated

T-SDN (TAPI, MEF-Legato...)

Or-Vi

PNFs & HNFs

TRANSPORT DOMAIN

VIRTUAL DOMAIN

© ETSI 2019
OSM provides Network as a Service (NaaS) in the form of NS and Slices that can span across different domains and technologies...
... while the northbound view for **Day-2 operations for the NS/Slice remains as high level actions**

![Diagram](image)

- **NS/NSI actions** are available northbound as **high level primitives** during runtime.
OK, BUT HOW CLOSE IS OSM TO ACHIEVE THESE GOALS?
OSM has demonstrated to be an agile community, with regular release delivery since its foundation...
... with a large community of users and testers behind each release...
Releases TWO & THREE brought a really comprehensive set of capabilities...

- Multi-VIM
- Multi-SDN
- Network Service scaling
- Monitoring
  - Plugin Model, NFVI to VDU correlation, App metrics, normalization
- Full Day 0 & Day 1 operations
- Multi-site Network Services
- VIM emulator (OpenStack-like)
- SDN assist for underlay chaining with EPA
  - Enables EPA deployments E2E for VIMs with no underlay support
- Explicit port ordering & Device Role Tagging
- Anti-affinity rules for VNF resiliency
- Role-Based Access Control
- One-click installer (multiple formats)
- Tenants/Projects in orchestration

... and many improvements in interoperability, stability, security, etc.
... Release FOUR extended OSM in almost all areas...

MODEL-DRIVEN NORTHBOUND INTERFACE
- SOL005 aligned
- OpenAPI model

MONITORING IMPROVEMENTS
- On-demand config of alarms & metrics
- Push notifications (via bus)
- Policy support

IMPROVED MODELLING & NETWORKING
- Full support of IP profiles
- Consistency checking of addressing
- MAC address setting
- Support of alternative images

USER EXPERIENCE & OPTIMIZATION
(CLOUD-NATIVE BUILD)
- Reduced footprint
- Faster startup and responsiveness
- Improved stability
- Better event and log visualization
- New Lightweight GUI
- Docker install (LW build)
- Batteries included: OSM client, emulator, event visualization, etc.

Available at: osm.etsi.org
... and Release FIVE has extended OSM in scope and functionality for 5G use cases

**NETWORK SLICING FOR 5G**
- Integrated Slice Manager
- IM extended to support NST and NSI

**MULTI-SITE EXTENSIONS**
- Dynamic inter-DC connections
- WIM plugin model
- Multi-VIM Enhancements

**MONITORING IMPROVEMENTS**
- Extended interop capabilities
- Policy support
- VNF + VIM Metrics Collection

**IMPROVED MODELLING & NETWORKING**
- Service Function Chaining (SFC)
- Physical Deployment Units
- Multi-VDU relations in VNF

**USER EXPERIENCE & OPTIMIZATION**
- GUI based NS composer
- Faster startup and responsiveness
- Better event and log visualization
- Docker, Vagrant and VM image install
COMMERCIAL AVAILABILITY
(a.k.a. Ecosystem pages)
What is the “OSM Ecosystem”?  

**OSM Ecosystem:** Companies listing together their products and offers related to OSM (like “OSM Yellow pages”)

- Searchable by potential customers looking for OSM-related products
- Only with demonstrable OSM-related products/offers
- Opt-in process, continuously open

OSM has already become the orchestrator of choice for the vast majority of 5G Research Projects.

https://osm.etsi.org/wikipub/index.php/Research
OSM has already organized 5 Hackfests to enable the ecosystem... and planning a new one!

1st OSM Hackfest (Sophia Antipolis, France)
• Co-located with 2nd Plugtests @ ETSI premises

2nd OSM Hackfest (Madrid, Spain)
• Co-located with Zero Touch CA Congress @ Intel

3rd OSM Hackfest (Oslo, Norway)
• Co-located with OSM-MR5 @ Telenor

4th OSM Hackfest (Palo Alto, USA)
• Co-located with OSM#6 @ VMware

5th OSM Hackfest (Barcelona, Spain)
• Co-located with OSM-MR6 @ CTTC

6th OSM Hackfest (Santa Clara, USA)
• Co-located with OSM#7 @ Intel
Quantifying benefits from ecosystem enablement: E.g. VNF onboarding

- VNF info request (INFO)
- VNF Topology confirmed (TOPO)
- Agreed Day-0 vs. Day-1 split (DAY0&1-SPLT)
- Day-1 vs. Day-2 configuration strategy (DAY1&2-STR)
- Design of Day-1 config. & parameters (DAY1-DES)
- Design of Day-2 primitives and metrics (DAY2-DES)
- VNFD available (VNFD)
- Dry run of instantiation (DRYRUN)
- Instantiation with Day-0 (DAY0-TST)
- Instantiation with Day-0 & Day-1 (DAY1-TST)
- Instantiation with Day-0 & Day-1 & Day-2 (DAY2-TST)

Pre-requisite
Advisable as pre-requisite
VNF internals
Static instantiation
Day-1 & Day-2
Quantifying benefits from ecosystem enablement: E.g. VNF onboarding

After ecosystem enablement, these activities are no longer needed.
Key takeaways

- **OSM provides Networks as a Service** (based on NS/NSI)
  - Coordinates behaviours of the different components
  - No fundamental differences between a VNF, a PNF or an Hybrid NF

- **OSM does not mandate specific protocols** to interact with the Network Functions

- The VNF provider does not need to know upfront the target NFVI+VIM at their customers (OSM plugin models handle it!)

- There is already a **thriving market of OSM-based offers**

- **Reverse engineering tasks can be avoided** effectively thanks to OSM IM
If you want to explore further:

- OSM Release FIVE – GIVE IT A TRY!
  - Follow instructions at: https://osm.etsi.org

- OSM EUAG White Papers

- OSM Release FIVE White Paper

- Tutorials and examples from 5th OSM Hackfest
  - https://osm.etsi.org/wikipub/index.php/5th_OSM_Hackfest
**DAY-0**

- **Objective**: VNF reachable to receive config. commands, but with no scenario-specific or hardcoded information.
  - i.e. Minimal config. to let the VNF accept further remote config. actions
- **Characteristics**:
  - Static
  - General
  - No hardcoded values
- **Mechanism**:
  - Pre-configured VM images and/or static cloud-init files

**DAY-1**

- **Objective**: Minimal initial configuration to make the VNF fully functional upon instantiation, based on parametrized inputs fed by OSM.
- **Characteristics**: Parametrization (via instantiation parameters) so that it is adapted to the environment and the rest of the NS.
- **Mechanisms**: Config protocols/procedures available in the VNF.

**DAY-2**

- Reconfiguration actions once the NS is already functional
OSM enables a delivery chain with clear roles while minimizes the complexity of Day-2 operations.
If we followed a traditional approach, we might suffer a lot of integrations in large NFV environments...
If we followed a traditional approach, we might suffer a lot of integrations in large NFV environments...
OSM in a Global Architecture (see EUAG WP)

**OSS/BSS**

- IaaS API
- NS API
- NSlice API
- TaaS API

**E2E Service Orchestration**

**Common Services**

- Identity
- DNS
- PKI
- SSO

**Portal(s) & Tools**

- Dashboards
- Wizards

**O&M**

- VIM(s) O&M
- Pod Manager(s)

**Backends**

- Logs
- Alarms
- Metrics

**NFs** (VNFs, PNFs, HNFs)

- Server(s)
- Storage backend
- Intra-DC fabric

**VIM(s)**

- IaaS LCM

**WIM(s)**

- TaaS LCM

**Inter-DC CE**

**Transport Layer**

© ETSI 2019
Integration points of OSM

- User Database
- SSO Server
- PKI

Common Services

SERVICES CONSUMED SOUTHBOUND

- OSS/BSS
- Portals & Wizards
- Installer & mgmt.
- O&M Tools
- Alarms collector
- Metrics collector
- Logs collector

Backends

- SDN-Controller (for SDN Assist)
- VIMs (API and monitoring)
- WIMs
OSM Or-Vi combinations: VIM and intra-DC SDN (options available)

CASE #1: Vanilla
- Overlay: Native
- No underlay

CASE #2: VIM + all SDN
- Overlay: SDNC
- Underlay: if available, via SDNC

CASE #3: VIM with partial SDN
- Overlay: Native
- Underlay: SDNC

CASE #4: SDN Assist
- Overlay: VIM native
- Underlay: SDNC, via OSM
OSM is really committed to optimize code efficiency continuously...

<table>
<thead>
<tr>
<th>Release</th>
<th>Min. RAM (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seed code (2016-Q1)</td>
<td>24</td>
</tr>
<tr>
<td>Release ZERO (2016-Q2)</td>
<td>24</td>
</tr>
<tr>
<td>Release ONE (2016-Q4)</td>
<td>16</td>
</tr>
<tr>
<td>Release TWO (2017-Q2)</td>
<td>12</td>
</tr>
<tr>
<td>Release THREE (2017-Q4)</td>
<td>8</td>
</tr>
<tr>
<td>Release FOUR target (2018-Q2)</td>
<td>2</td>
</tr>
</tbody>
</table>
... bearing in mind that perspective sometimes helps

Min. RAM (GB)

- Star Trek's ENTERPRISE (2017-Q4)
- Seed code (2016-Q1)
- Release ZERO (2016-Q2)
- Release ONE (2016-Q4)
- Release TWO (2017-Q2)
- Release THREE (2017-Q4)
- Release FOUR target (2018-Q2)

Min. RAM: 336 GB

x168

Min. RAMs:
- Star Trek's ENTERPRISE (2017-Q4): 336 GB
- Seed code (2016-Q1): 24 GB
- Release ZERO (2016-Q2): 24 GB
- Release ONE (2016-Q4): 16 GB
- Release TWO (2017-Q2): 12 GB
- Release THREE (2017-Q4): 8 GB
- Release FOUR target (2018-Q2): 2 GB