OSM Hackfest
Introduction to OSM
Gerardo García (Telefónica)
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
... to native NFV management, with highly efficient automation and operation

**Day 0**
- VNF deployment
- NS deployment (complex topology)

**Day 1**
- VNF configuration
- Neighbor configuration
- License activation

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

© ETSI 2017
However, operating a real Virtualised Network looks more like this, with multiple sites and technologies…

(*) Topologies and combinations of technologies are provided as examples
... that require the configuration and coordination of an even larger set of elements...

**VNF setup**
- Ansible
- YANG
- Expect

**NFVI + VIM domain**
- OpenStack
- vCloud Director
- AWS
- KVM
- ESXi
- XEN
- Haswell
- Broadwell
- Skylake
- Niantic
- Fortville
- Other NIC

**Switch + SDN domain**
- ODL
- ONOS
- Flood Light
- Switch A
- Switch B

**NFV SITE**
- Paravirt
- SR-IOV
- Passthrough
- VNF A
- VNF B
- VNF C

© ETSI 2017
... and unless we are ready to respect the layering, management gets really complicated.

![Diagram of NFV Orchestration and NFVI/VIM layers]

- **NFVI**:
  - Hypervisor
  - Hardware

- **VIM**:
  - Cloud Management System
  - SDN Controller

- **NS#1**:
  - VNF 1
  - VNF 2
  - VNF 3

- **NFV Orchestration**
... and unless we are ready to respect the layering, management gets really complicated.

```
Hardware

Hypervisor

VNF 1

VNF 2

VNF 3

NFVI

NFV Orchestration

VIM

Config. vSwitch

Config. NIC

Config. Acceleration

Cloud Management System

SDN Controller

Console access to fix VNF boot

Pre-create ad hoc flavors

Customize for the use case

Create ad hoc connections

NS#1

Config.

Config.

Config.

Deploy VIM-dependent template

Deploy
```

© ETSI 2017
Proper modelling is required to assemble VNFs automatically... but it’s not a trivial task.

FROM...

INADEQUATE VNF MODELLING

UNEVEN VNF CATALOGUE

BASIC NSD

BASIC AND HAND-MADE NETWORK SERVICE

HARD ONBOARDING

Ad hoc integration

...TO...

VNFs AS REPLACEABLE COMPONENTS

ONBOARDING

VNF CATALOGUE

NSD

MANO

NETWORK SERVICE

© ETSI 2017
Strategy for a smart convergence

FOCUS ON WHAT WE HAVE IN COMMON

SERVICE PROVIDER 1
- BSS 1
- OSS
- MANO
- VIM A

SERVICE PROVIDER 2
- BSS 2
- OSS
- MANO
- VM B

SERVICE PROVIDER 3
- BSS 1
- BSS 2
- EMS
- OSS
- MANO
- SDN a
- VIM A
- SDN b
- VIM B
- VIM C

Key is INTEROPERABILITY, not full architecture

MULTIPLE VIMs & SDNs ARE HERE TO STAY (public clouds too!)

LEVERAGE ON ETSI NFV WORK

READY FOR GREENFIELD AND BROWNFIELD

PERFORMANCE MATTERS FOR THE BUSINESS CASE

OPEN SOURCE AS TOOL TO FACILITATE CONVERGENCE
3 reasons to go to open source

1. To accelerate the availability of a reference standard

2. To build a wide and competitive market of producers and consumers

3. To answer the BUY vs. MAKE question
Some requirements to make Open Source MANO fit for purpose

- **OPEN TO PLAYERS OF ALL SIZES**
  - Need of a diverse community
  - Adding expertise & demand

- **SUSTAINABLE AND RELIABLE**
  - The leaner, the better

- **Focus on the core, leave the rest to INTEROP**

- **OPEN TO NEW TECHNOLOGIES**
  - Key for future-proof

- **AND READY TO DELIVER!**
OSM has already delivered 4 releases

- MWC demo (Feb)
- Kick-off (Apr)
- Release ONE (Oct)
- Release TWO (Apr)
- 1st ETSI NFV Plugtest (Jan)
- Release THREE (Oct)

© ETSI 2017
OSM has already delivered 4 releases

- MWC demo (Feb)
- Kick-off (Apr)
- Release ONE (Oct)
- Release TWO (Apr)

2016

- Release ZERO (May)

Production Readiness

- 10K+ installs & upgrades
- 6000+ downloads
- 70+ countries
Release TWO brought already a really comprehensive set of capabilities...

- **Multi-VIM**
- **Multi-SDN**
  - OPEN DAYLIGHT
  - FloodLight
  - ONOS

**SDN assist for underlay chaining with EPA**
Enables EPA deployments E2E for VIMs with no underlay support

- **One-click installer**
  - (multiple formats)

- **Network Service scaling**
  - Multi-site Network Services

- **Full Day 0 & Day 1 operations**

... and many improvements in interoperability, stability, security, etc.
... that **Release THREE** has extended with a new set of advanced features

**Role-Based Access Control**

**Tenants/Projects in orchestration**

**Monitoring (experimental)**

Plugin Model, NFVI to VDU correlation, App metrics, normalization

**Explicit port ordering & Device Role Tagging**

**Full Interop with VIO 4**

**Anti-affinity rules for VNF resiliency**

**VIM emulator (OpenStack-like)**

... and many improvements in interoperability, stability, security, etc.
BUT... HOW DOES IT WORK IN PRACTICE?
REMEMBER?

Hardware

Hypervisor

NFVI

VNF 1

VNF 2

VNF 3

NFV Orchestration

NS#1

Config.

Config.

VIM

Cloud Management System

SDN Controller

Config. vSwitch

Config. NIC

Config. Acceleration

Config.

Deploy

VIM-dependent template

Console access to fix VNF boot

Pre-create ad hoc flavors

Create ad hoc connections

Customize for the use case

Deploy

DEPLOY

© ETSI 2017
Open Source MANO provides a unique entry point for management, even for Day 2 operation...

[Diagram of MANO architecture]
... so that all that matters from an operation perspective is the Network Service itself, with one single entry point.
Having an appropriate Information Model, agnostic to VIM and VNF config, is key for NFV Orchestration.

- **VNF Packages**
  - Mgmt Procedures
  - Resource Description

- **NS Package**
  - Mgmt Procedures
  - Topology

- **OSM’s NBI**

- **SO**

- **VCA**
  - Charms

- **RO**
  - VIM/SDN Connectors

- **NFVI**
  - NS#1
  - VNF 1
  - VNF 2
  - VNF 3

- **VIM**

© ETSI 2017
OSM Information Model is open and all the information is publicly available

• Wiki page: https://osm.etsi.org/wikipub/index.php/OSM_Information_Model

• YANG models:
  • Gitweb (master): https://osm.etsi.org/gitweb/?p=osm/IM.git;a=tree
  • Folder: models/yang

• IM tree representation of YANG models both in text and navigable format:
  • VNFD: http://osm-download.etsi.org/ftp/osm-doc/vnfd.html
  • NSD: http://osm-download.etsi.org/ftp/osm-doc/nsd.html
OSM enables a delivery chain with clear roles while minimizing the complexity of Day-2 operations.

- **VNF PACKAGE BUILDING**
  - VNF A Package
  - VNF A Images

- **NS PACKAGE BUILDING**
  - NS Package
    - VNF A Package
    - VNF A Images
    - VNF B Package
    - VNF B Images
    - VNF Z Package
    - VNF Z Images

- **NS INSTANTIATION**
  - NS Package
    - VNF A Package
    - VNF A Images
    - VNF B Package
    - VNF B Images
    - VNF Z Package
    - VNF Z Images

- **DAY 2 OPERATION**
  - Actions (Day 2)
  - PARAMETERS
  - PRODUCTION INFRASTRUCTURE
  - SERVICE PROVIDER

- **VNF VENDORS**
  - Infra Type #1
  - Infra Type #2
OSM ARCHITECTURE
Architectural Principles for OSM

- **LAYERING**
  - Require clear delineation between the layers and modules.
  - Should be broadly aligned with ETSI-NFV

- **ABSTRACTION**
  - Moving up/down the layers should offer clear differentiation in the levels of abstraction/detail presented.

- **MODULARITY**
  - Even within layers, clear modularity enabled with a plugin model preferred to facilitate module replacements as OSM community develops.

- **SIMPLICITY**
  - Solution must have the minimal complexity necessary to be successful and no more.
OSM scope

RUN-TIME SCOPE

- Automated E2E Service Orchestration
- Superset of ETSI NFV MANO
- Plugin model for integrating multiple SDN controllers
- Plugin model for integrating multiple VIMs
- Integrated Generic VNFM with support for integrating Specific VNFMs
- Support for Physical Network Function integration
- Greenfield and brownfield deployments

DESIGN-TIME SCOPE

- Network Service Definition
- Model-Driven Environment with Data Models aligned with ETSI NFV
- VNF Package Generation
- GUI

© ETSI 2017
OSM Architecture

DevOps
- One Click Installer
- Jenkins
- Host
- Common
- UI
- MON
- SO
- N2VC
- VCA
- RO

User Interface
- VNF Package Generator
- VNF/NS Catalog Composer
- Account Manager
- Launchpad
- OSM Client

Service Orchestrator
- API Service & Mgmt. Endpoint
- Service Orchestration Engine
- Configuration Data Store
- Network Service Composition Engine
- Catalog Manager
- Resource Orchestration Plugin

Network Service To VNF Communication (N2VC)
- VNF Configuration & Abstraction
  - VCA Engine (Juju Adapter)
  - VNF Monitoring

Resource Orchestration
- Resource Orchestration Engine
  - VIM Plugin
    - OpenStack
    - Amazon Web Services EC2/VPC
    - OpenVIM
    - VMware vCD
  - SDN Plugin
    - OpenDaylight
  - ONOS
  - Floodlight

Monitoring
- Core
  - Monitoring Tool Plugin
    - OpenStack (Aodh/Gnocchi)
    - AWS CloudWatch
  - VMware vRealise Operations
OSM Architecture
Simplified view

UI: User interface
SO: Service Orchestrator
VCA: VNF Configuration & Abstraction
RO: Resource Orchestrator

© ETSI 2017
The **plugin model** facilitates interop with different clouds and controllers

- OpenStack (several flavours)
- OpenVIM
- VMware (vCD)
- AWS
- **Add your plugin here**

<table>
<thead>
<tr>
<th>CMS PLUGINS</th>
<th>SDN PLUGINS</th>
</tr>
</thead>
<tbody>
<tr>
<td>• OpenStack (several flavours)</td>
<td>• ODL</td>
</tr>
<tr>
<td>• OpenVIM</td>
<td>• ONOS</td>
</tr>
<tr>
<td>• VMware (vCD)</td>
<td>• FloodLight</td>
</tr>
<tr>
<td>• AWS</td>
<td>• <strong>Add your plugin here</strong></td>
</tr>
</tbody>
</table>
EPA support combined with SDN Assist enables chaining of high performance VNFs

1. Accurate assignment of resources at VM level
2. Proper assignment of I/O interfaces to the VM
3. SDN gives the ability to create underlay L2 connections
   - Interconnecting VMs
   - Attaching external traffic sources
Operations and lifecycle with OSM
Day-0, Day-1, and Day-2
VNF’s operational procedures are already embedded in their VNF Package...

VNF package

- VNFD
- VNF artifacts
- Additional metadata?

Capacity aspects
- VNF resource orchestration info (EPA resources and internal connectivity)

Functionality aspects
- Descriptive information
  - metadata.yaml
  - config.yaml
  - actions.yaml
- Executables
  - Hooks
  - Actions
- Additional info (icon, README)
... ready to be used in complex Network Services, which define their own E2E operational procedures...

HIGH-LEVEL PRIMITIVES

E.g.:
- Add subscriber
- Add service profile
- Update subscriber profile
- Add service access to subscriber
...

PARAMETRIZED NS

PARTICULARIZATION

IP pools = x1
QoS1 definition = x2
QoS2 definition = x3
...

© ETSI 2017
OSM enablers for NFV management

OSM NB API:
- Create new NS instance
- Get NS instance record
- Get VNF metrics
- Call NS primitive
- Scale in/out NS instance

VNFD:
- Basic configuration (SSH keys, hostname, user-data scripts, etc.)
- Initial config primitive
- Config primitive
- Charms → Actions
- Charms → Metrics

NSD:
- Initial config primitive
- Service primitive
- Scaling groups
- Pre and post scaling primitives
- Scripts
Day-0 / Day-1 configuration (at instantiation time) is descriptor-based

OSM NB API:
- Create new NS instance

VNFD:
- Basic configuration (SSH keys, hostname, user-data scripts, etc.)
- Initial-config-primitive
- Charms -> Actions

NSD:
- Initial config primitive
- Scripts
Day-2 configuration

OSM NB API:
- Get NS instance record
- Call NS or VNF primitive

VNFD:
- Config primitive
- Charms -> Actions

NSD:
- Service primitive
- Scripts
Day-2 elasticity (scaling)

OSM NB API:
- Get NS instance record
- Scale in/out NS instance

VNFD:
- Basic configuration (SSH keys, hostname, cloud-init scripts, etc.)
- Charms → Actions

NSD:
- Scaling groups
- Pre and post scaling primitives
OSM organization
OSM has an organization oriented to the production of upstream code...
... which favours efficient decision taking

**LEADERSHIP GROUP**
Sets the policies of the organization
Takes administrative decisions

Confirms TSC Chair
Supports TSC work

**TSC**
Sets the Information Model
Decide features per release

Reports progress to the LG

**END USER ADVISORY GROUP**

Reports progress on features

Produces use cases
Produces feature requests

Commits project releases

**MDG**
Creates/removes MDG
Appoints/revokes MDG leads

Commits module releases

**MDG**

Commits project releases
OSM CI/CD
OSM has a rich CI/CD Pipeline...

[Diagram showing stages of the CI/CD pipeline with labels such as 'Trigger per-module pipelines', 'Scan', 'U Test', 'Build', 'Archive', 'Commit', 'Artifact Storage', 'System Tests', 'Smoke Tests', 'System Install', 'Stage 1', 'Stage 2', 'Stage 3', 'Stage 4', 'OSM Release Server', 'VNF', 'NFVI & VIM']
... that enjoys the connection to OSM’s Network of Remote Labs to test interop with different VIMs and NFVIs