

Open Source  
MANO

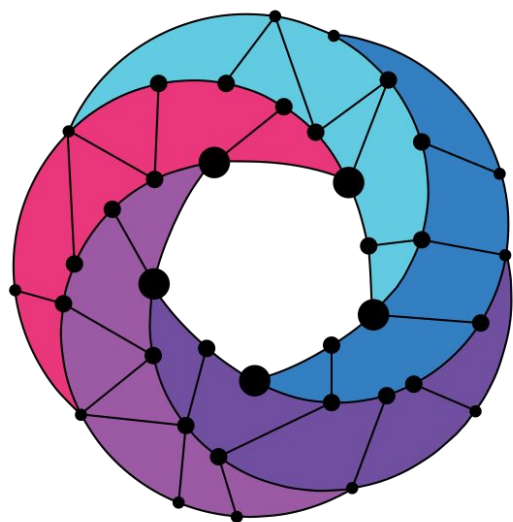
8<sup>th</sup> OSM Hackfest

# Hackfest Week Plan

OSM#8 & Hackfest Lucca	MONDAY 18 Nov		TUESDAY 19 Nov		WEDNESDAY 20 Nov		THURSDAY 21 Nov		FRIDAY 22 Nov	
09:00..10:30 (1h30)	LG		Opening Plenary	S2 Basic Descriptors	TECH	S6.1 Day1/2 Proxy Charms	TECH	S6.3 Nat Charms	Closing Plenary	S11 CI/CD Robot Frwrk
10:30..11:00 (0h30)	COFFEE BREAK									
11:00..13:00 (2h)	EUAG		Opening Plenary	S3 Multi-VDU VNFs	TECH	S6.1 Day1/2 Proxy Charms	TECH	S7 Fault & Perf Management	S12.2 VNF Onb to VCD	S10 How to contribute
S4.1 EPA				S6.2 Ansible		Wrap-up				
13:00..14:00 (1h)	REGISTRATION		LUNCH BREAK							
14:00..15:30 (1h30)	Welcome		TECH	S4.2 SR-IOV	5G Day	TECH	S4.3 PNFs & HNFs			
	TSC	S0 Intro		S5 Day 0			S12.1 VNF Onb Walkthrough			
		DEMO 0								
15:30..16:00 (0h30)	COFFEE BREAK									
16:00..18:00 (2h)	TSC & MDL	S1 OSM Install	TECH	S8 - Network Slicing	5G Day	TECH	S9 Service Function Chaining			
19:30 ...				Networking Dinner						
				Sessions		Room		Group		
				OSM Hackfest		OSM		All		
				5G Day		OSM		All		
				OSM Plenary / TECH		ETSI		OSM All		
				TSC / TSC&MDL				OSM TSC/TSC&MDL		
				LG				OSM LG		
				EUAG				OSM EUAG		



Slides



Open Source  
**MANO**

# 8<sup>th</sup> OSM Hackfest Introduction

José Miguel Guzmán (Whitestack)  
[jmguzman@whitestack.com](mailto:jmguzman@whitestack.com)

# Agenda

- Quick overview of NFV
- Introduction to the latest OSM Release





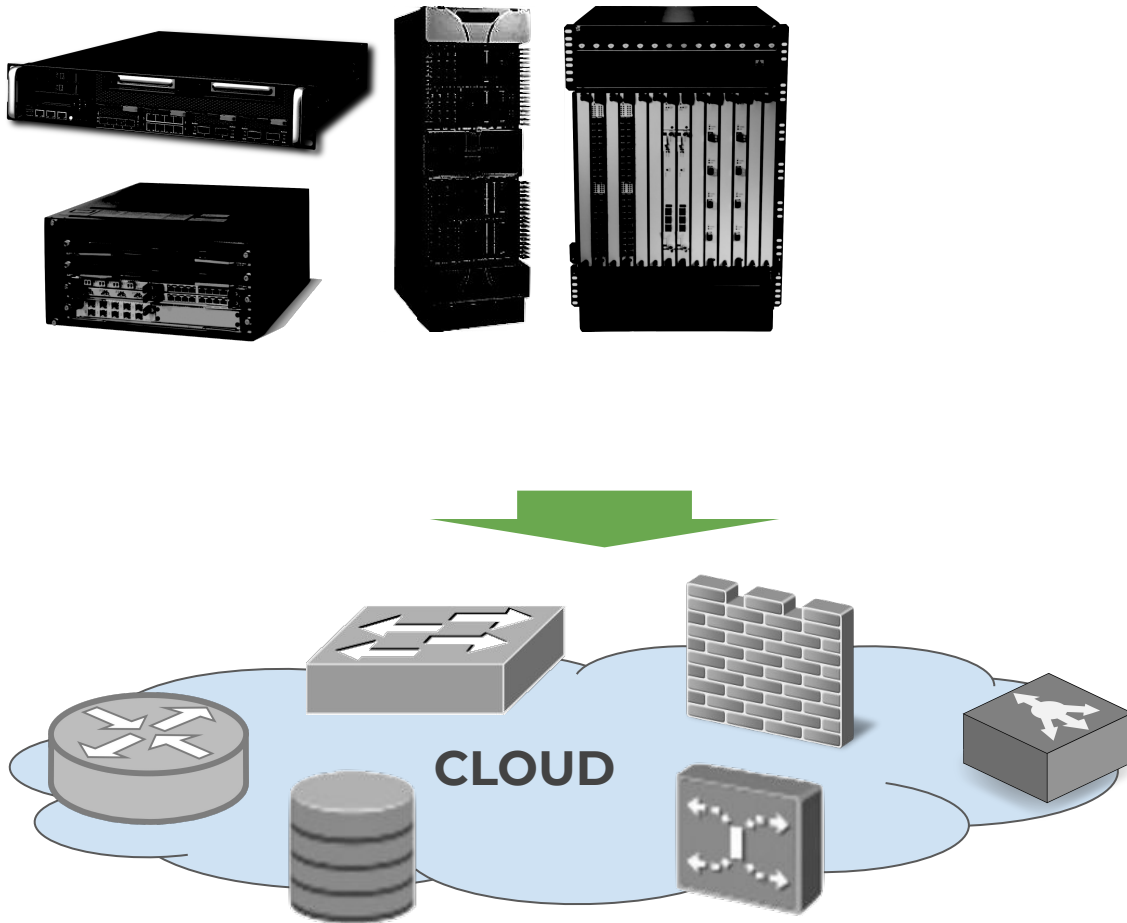
Open Source  
**MANO**

## Quick review of NFV



**Home of NFV**

# What is NFV trying to address?



- Network Function Virtualization (NFV) proposes to **virtualize network functions that typically run in dedicated appliances**
- The main goal is to **support virtualized functions over COTS servers.**
- Virtual Network Functions (VNFs) acquire **all the advantages of Cloud Applications!**



# The original idea.... triggered an industry movement

- Initial white paper was written in 2012 by the world's leading telecom network operators (Europe, América & Asia).
- This group evolved to the ETSI NFV ISG (Industry Specification Group), formed by 300+ companies.
- Their main motivation had to do with reducing TCO of building a network by using open solutions.

## Network Functions Virtualisation

*An Introduction, Benefits, Enablers, Challenges & Call for Action*

### OBJECTIVES

This is a non-proprietary white paper authored by network operators.

The key objective for this white paper is to outline the benefits, enablers and challenges for Network Functions Virtualisation (as distinct from Cloud/SDN) and the rationale for encouraging an international collaboration to accelerate development and deployment of interoperable solutions based on high volume industry standard servers.

### CONTRIBUTING ORGANISATIONS & AUTHORS

<b>AT&amp;T:</b>	Margaret Chiosi.
<b>BT:</b>	Don Clarke, Peter Willis, Andy Reid.
<b>CenturyLink:</b>	James Feger, Michael Bugenhagen, Waqar Khan, Michael Fargano.
<b>China Mobile:</b>	Dr. Chunfeng Cui, Dr. Hui Deng.
<b>Colt:</b>	Javier Benitez.
<b>Deutsche Telekom:</b>	Uwe Michel, Herbert Damker.
<b>KDDI:</b>	Kenichi Ogaki, Tetsuro Matsuzaki.
<b>NTT:</b>	Masaki Fukui, Katsuhiro Shimano.
<b>Orange:</b>	Dominique Delisle, Quentin Loudier, Christos Kolias.
<b>Telecom Italia:</b>	Ivano Guardini, Elena Demaria, Roberto Minerva, Antonio Manzalini.
<b>Telefonica:</b>	Diego López, Francisco Javier Ramón Salguero.
<b>Telstra:</b>	Frank Ruhl.
<b>Verizon:</b>	Prodip Sen.

### PUBLICATION DATE

October 22-24, 2012 at the "SDN and OpenFlow World Congress", Darmstadt-Germany.

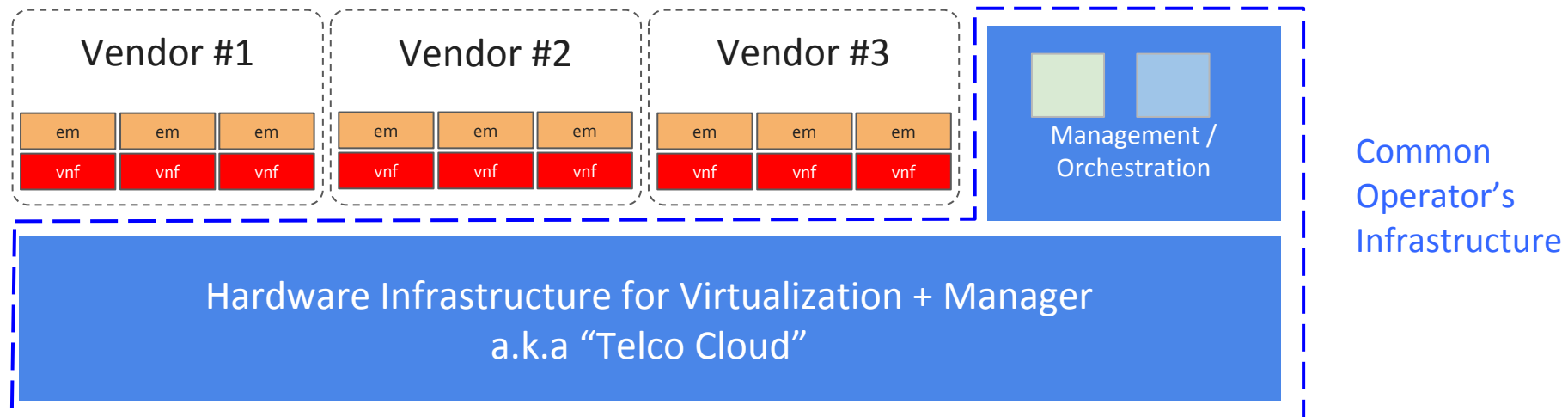
# ETSI Publications

- Based on member's feedback, field experiences and proof of concepts, standard documents have evolved.
- 60+ publications exist today, including the following three main documents:
  - NFV Architectural Framework  
[http://www.etsi.org/deliver/etsi\\_gs/NFV/001\\_099/002/01.02.01\\_60/gs\\_NFV002v010201p.pdf](http://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.02.01_60/gs_NFV002v010201p.pdf)
  - NFV Infrastructure Overview  
[http://www.etsi.org/deliver/etsi\\_gs/NFV-INF/001\\_099/001/01.01.01\\_60/gs\\_NFV-INF001v010101p.pdf](http://www.etsi.org/deliver/etsi_gs/NFV-INF/001_099/001/01.01.01_60/gs_NFV-INF001v010101p.pdf)
  - NFV Management and Orchestration  
[http://www.etsi.org/deliver/etsi\\_gs/NFV/001\\_099/002/01.02.01\\_60/gs\\_NFV002v010201p.pdf](http://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.02.01_60/gs_NFV002v010201p.pdf)



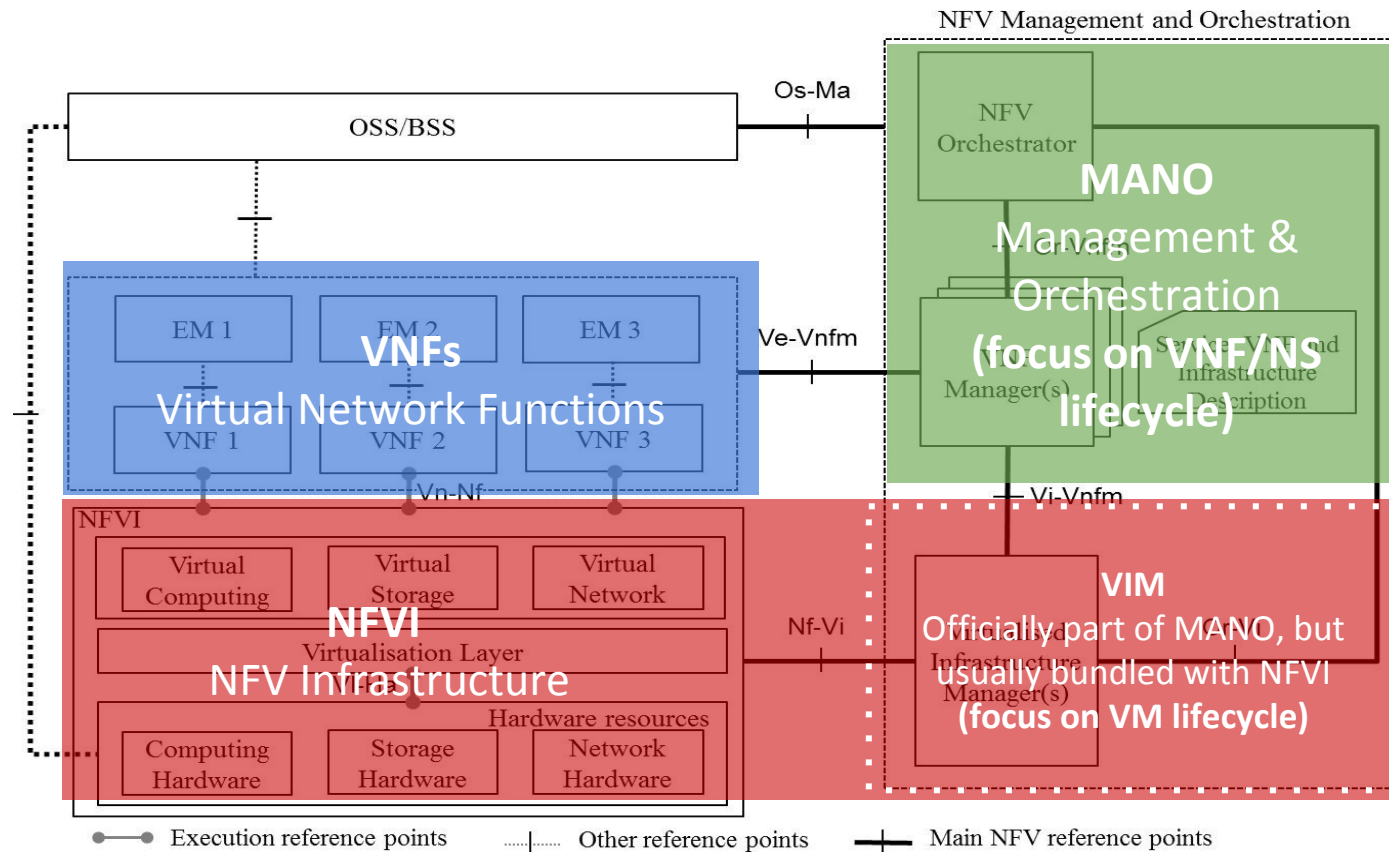
# Benefits of a standard NFV architecture

We are looking for a **unified and generic virtualization infrastructure**, compatible with any vendor's Virtual Networking Function (VNF), so **standardization is a must**.



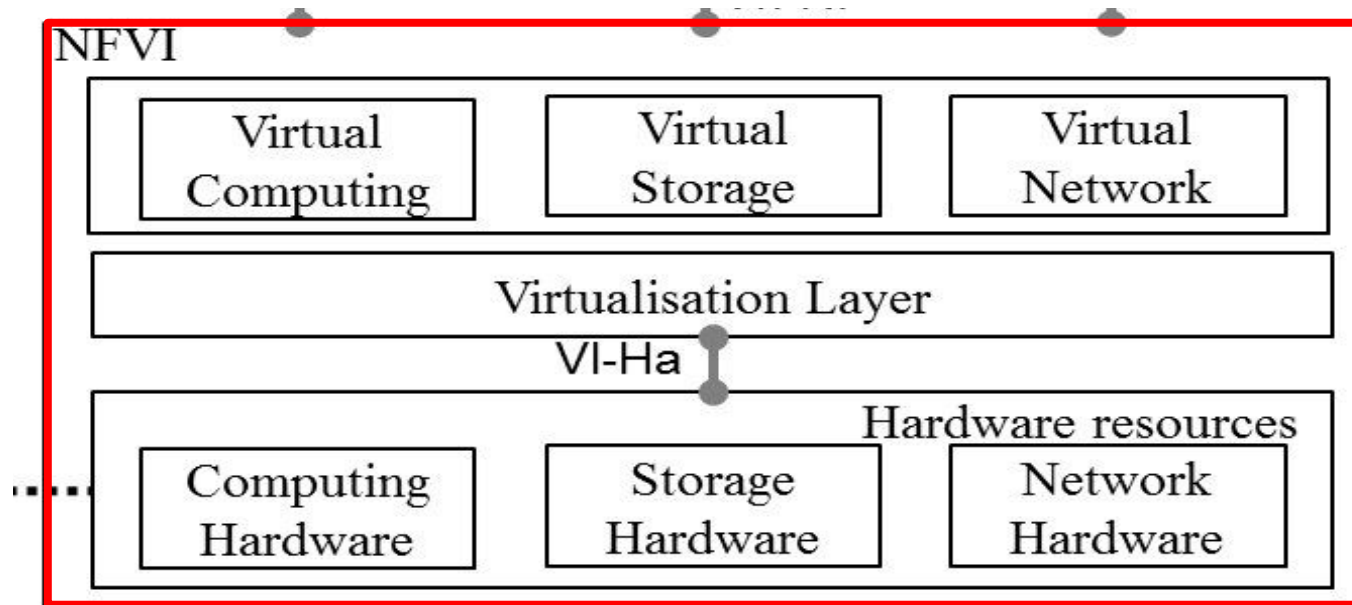
# The ETSI NFV Architectural Framework

The standard architecture can be better understood in three blocks:



# NFVI: NFV Infrastructure

NFVI goal is to provide a virtualization environment for VNFs, including virtual compute, storage and networking resources.



- But! Networking applications may have more strict performance requirements, we will discuss that later.



# NFVI: NFV Infrastructure

## VNF Special Requirements

VNFs, especially data-plane ones, usually have additional requirements than common cloud applications, including:

- **Minor latency** (disk I/O & network)
  - Faster hardware (More cores, SSD disks, faster buses)
  - Dataplane acceleration
- **Higher throughput or PPS**
  - Dataplane acceleration
  - EPA: Enhanced Platform Awareness
- **Geographical distribution**
  - multi-site cloud
- **Horizontal auto-scaling**
  - automated operations (orchestration)



# NFVI: NFV Infrastructure

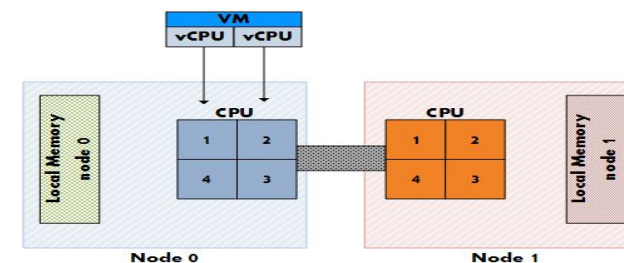
## VNF Special Requirements

EPA covers the different approaches that can be taken at the NFVI layer to increase performance while maintaining a generic (COTS) infrastructure. VIM and MANO should be able to request them.

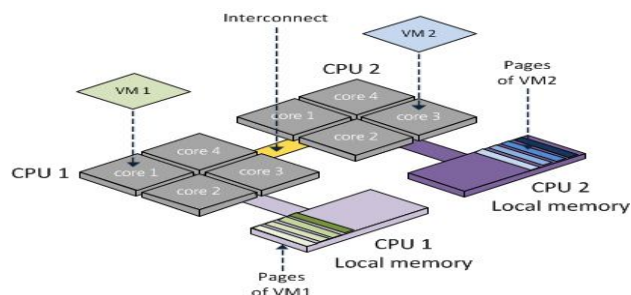
### Huge Pages



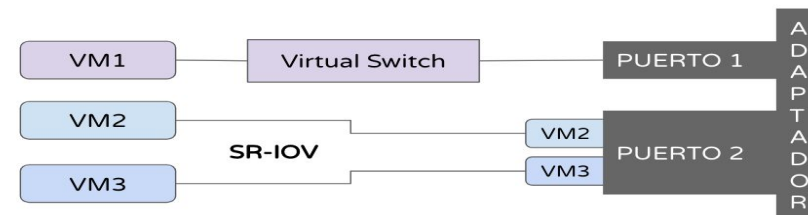
### NUMA Topology Awareness



### CPU Pinning



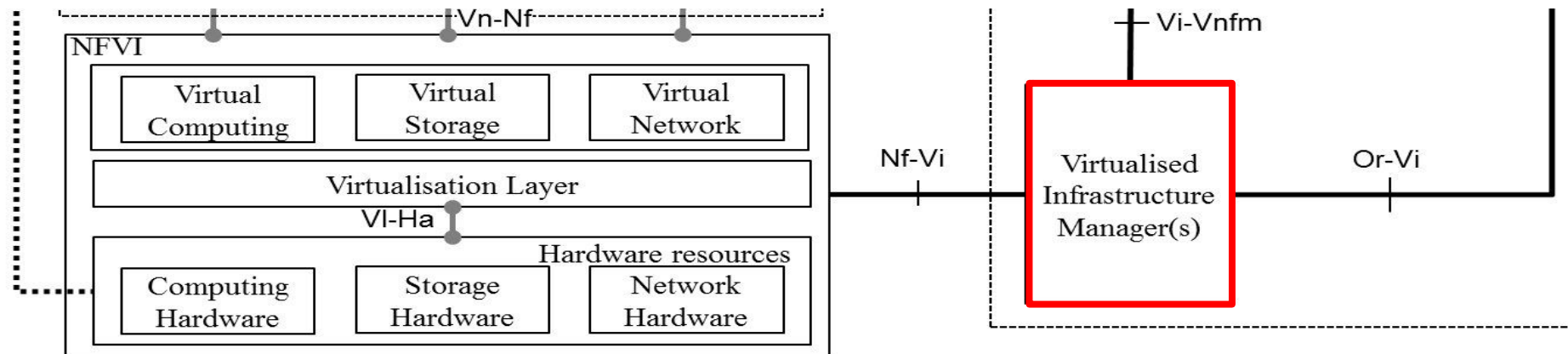
### Data Plane assignment



# VIM:

## Virtualised Infrastructure Manager

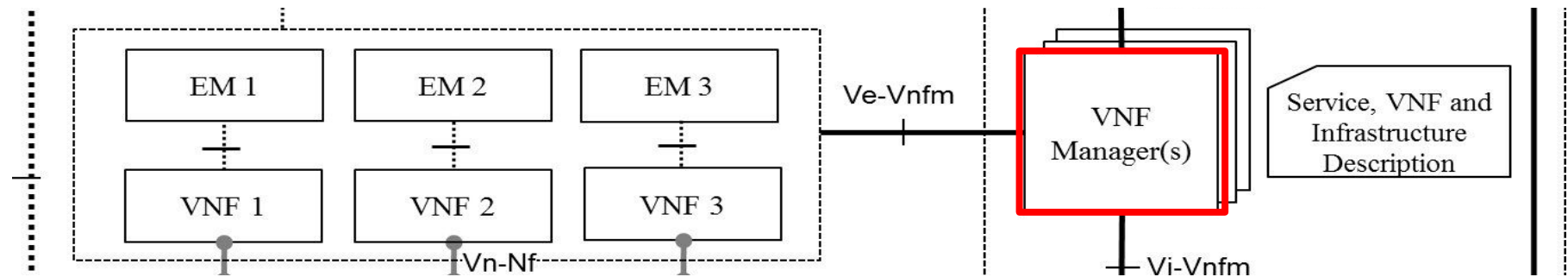
- The Virtualised Infrastructure Manager is part of the 'MANO Stack' and provides lifecycle management for virtualized resources (VMs, volumes, networking paths and connectivity, etc.)



# VNFM:

## VNF Manager

- The VNF Manager, also part of the 'MANO Stack', covers **lifecycle management for Virtual Network Functions (VNFs)**, either directly or through their own Element Management System (EMS).



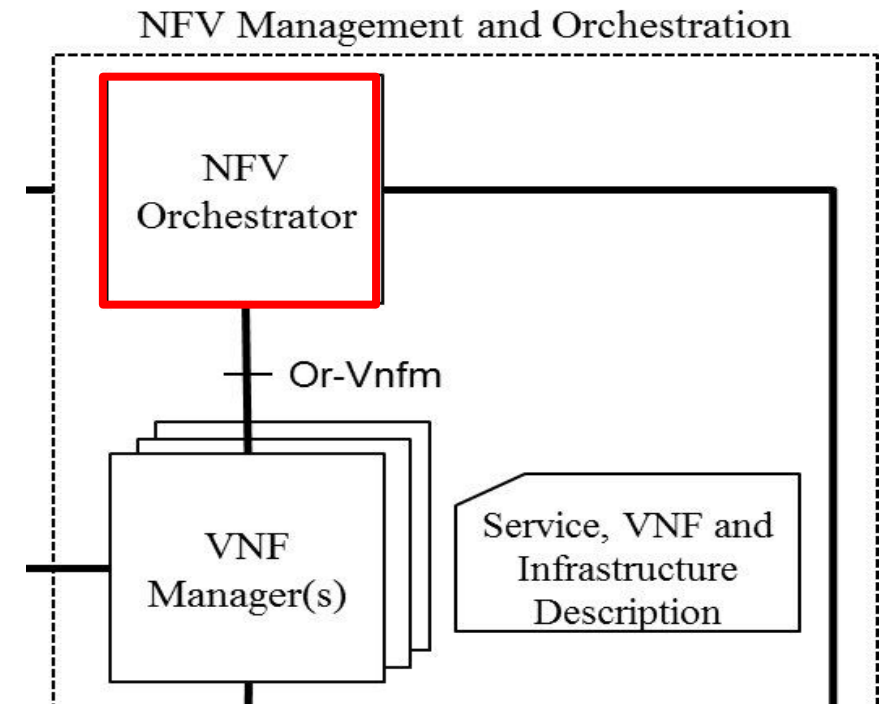
- VNF Managers can be generic (current trend), or vendor-specific ones.

CANONICAL



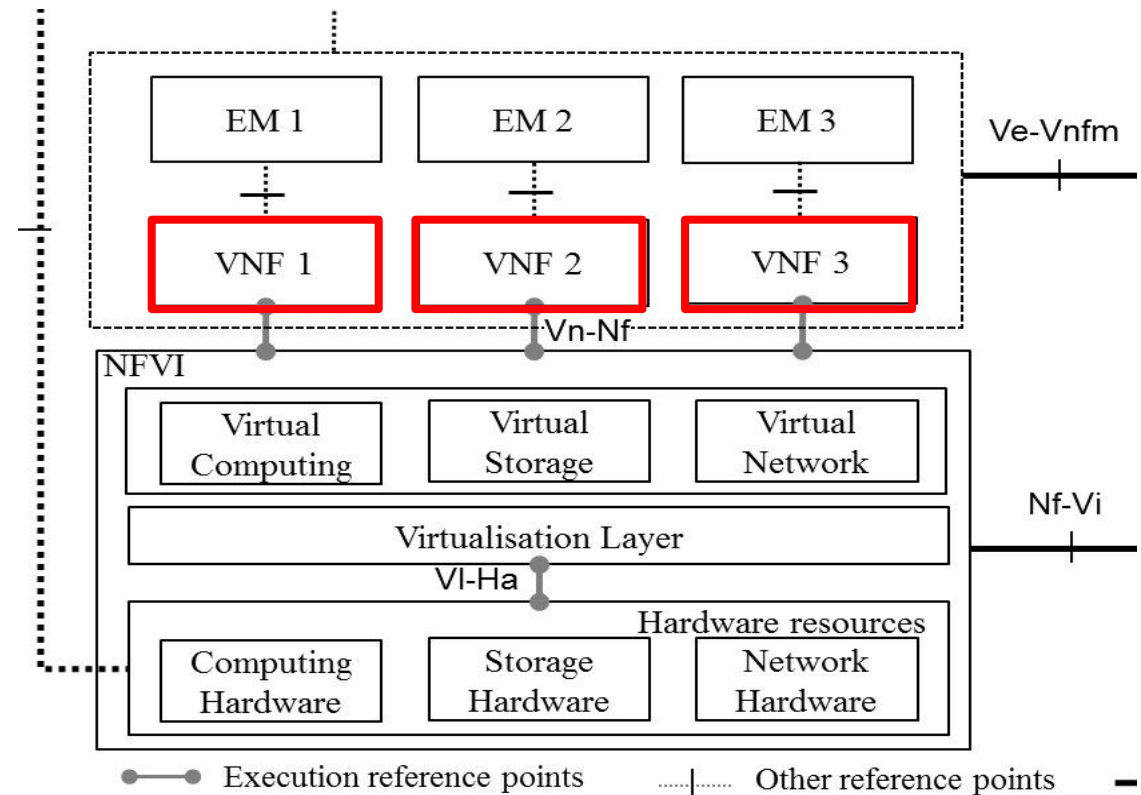
# MANO: NFV Orchestrator (NFVO)

- The NFV Orchestrator, the higher entity in the 'MANO Stack', covers general resource orchestration and services lifecycle, which comprise multiple VNFs and define their roles (traffic paths, scaling decisions, and other service-related requirements)
- It can interact with a generic VNF Manager, or vendor-specific ones.



# Virtual Network Functions (VNF)

- Finally, the VNFs, which are supported by the underlying NFVI, and managed by their own EM (internal, element manager) and the VNF Manager (external, 'context-aware' manager)
- They should be able to provide any networking function and interact with other VNFs.



# VNF Descriptor files (VNFD)

One of the most important aspects of achieving a unified VNF catalogue, is having a standard way of describing VNFs.

- MANO solutions should give the possibility to describe VNFs through 'descriptor files'
- The industry's goal is a unified and standard descriptor file format across different platforms.
- Both NS (comprised of VNFs) and VNFs should be described in a simple way.

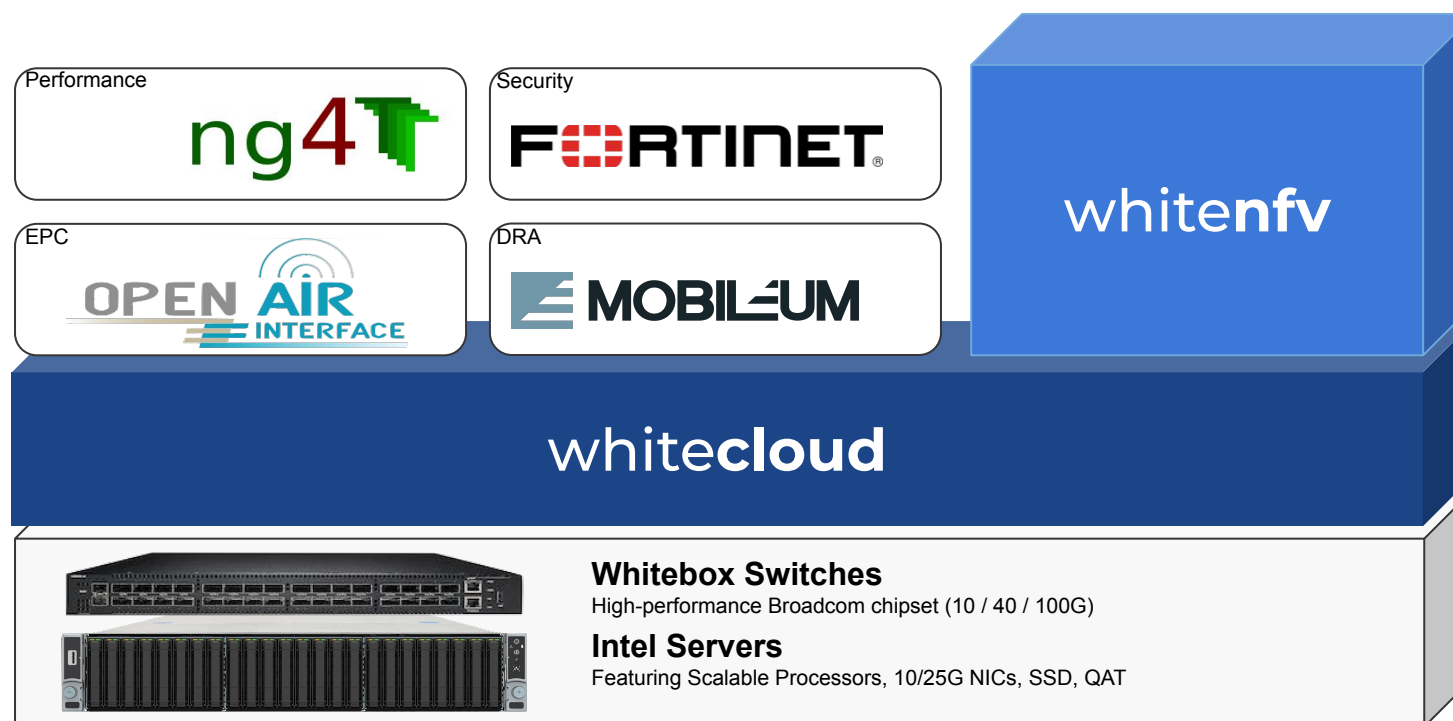
```
vnfd:vnfd-catalog:
  vnfd:vnfd:
    - vnfd:connection-point:
      - vnfd:name: eth0
        vnfd:type: VPORT
      vnfd:description: Generated by OSM pacakage generator
      vnfd:id: ubuntuvmf_vnfd
      vnfd:mgmt-interface:
        vnfd:cp: eth0
      vnfd:name: ubuntuvmf_vnfd
      vnfd:service-function-chain: UNAWARE
      vnfd:short-name: ubuntuvmf_vnfd
      vnfd:vdu:
        - vnfd:cloud-init-file: cloud_init
          vnfd:count: '1'
          vnfd:description: ubuntuvmf_vnfd-VM
          vnfd:guest-epa:
            vnfd:cpu-pinning-policy: ANY
          vnfd:id: ubuntuvmf_vnfd-VM
          vnfd:image: ubuntu_admin
          vnfd:interface:
            - rw-vnfd:floating-ip-needed: 'false'
              vnfd:external-connection-point-ref: eth0
```



# Multi-vendor NFV Showcase

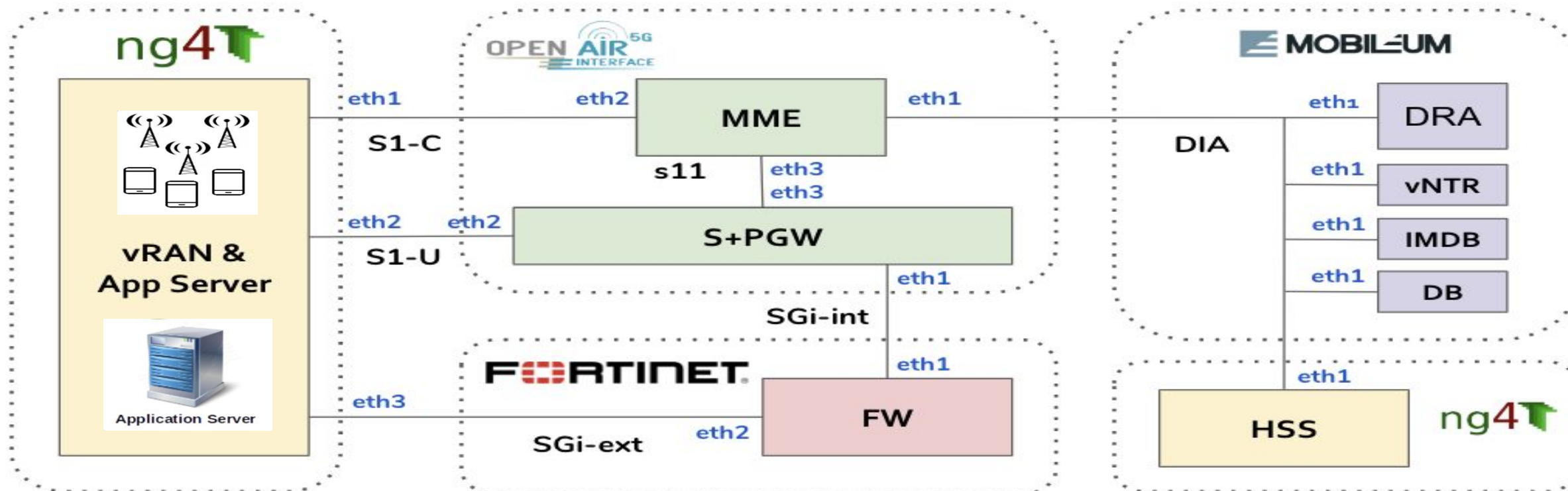
With the support of **leading NFV-enablers**, putting together a number of leading VNF vendors, on top of commoditized x86 infrastructure, managed by OpenStack and Open Source MANO.

**Goal:** to demonstrate publicly that multi-vendor networks are possible



# Multi-vendor NFV Showcase

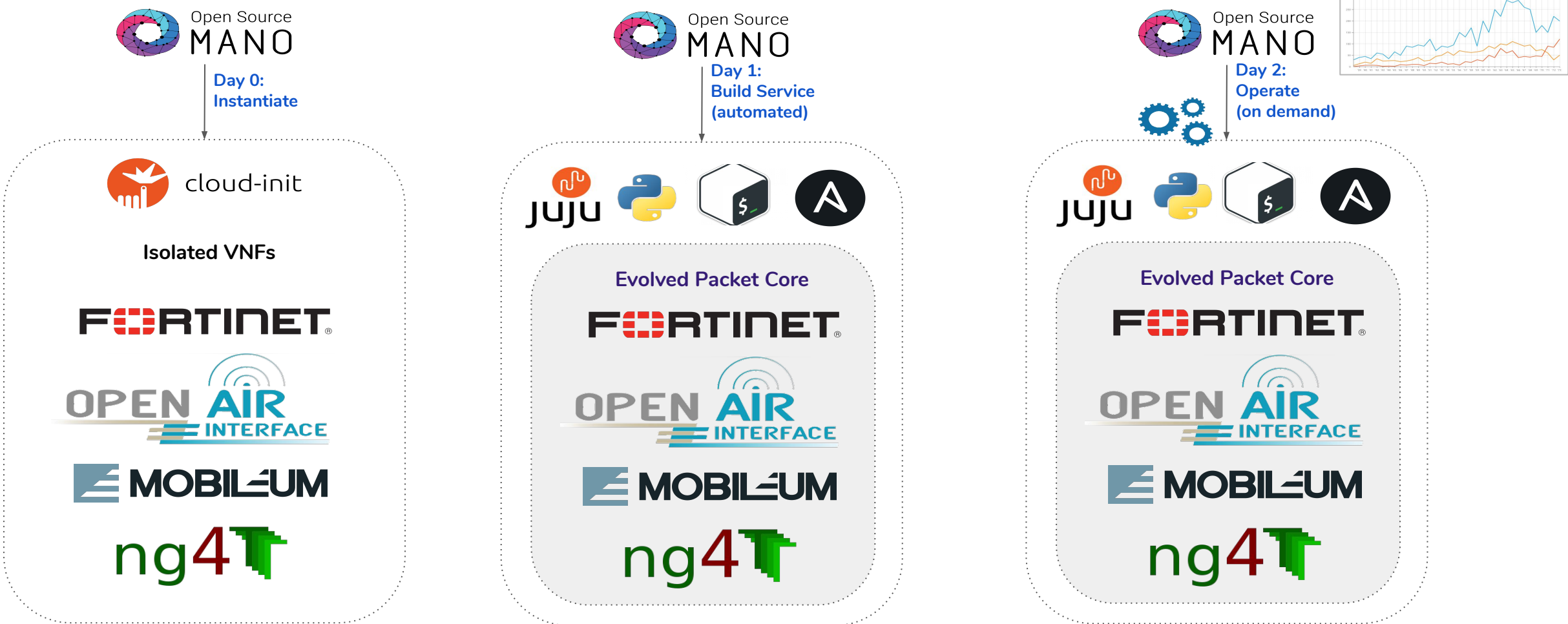
- **ng4t** VRAN: Emulates the vRAN
- **OpenAir Interface**: Implement the vEPC (MME, SGW, PGW)
- **Fortinet**: implement security
- **Mobileum**: implement DRA and NTR (Roaming Steering)





# Multi-vendor NFV Showcase

Day-0, Day-1 and Day-2



# Multi-vendor NFV Showcase

The final results, including configurations used for deploying this vEPC, are published, following the guidelines from **ETSI Plugtests Programme**.



<https://www.whitestack.com/posts/results-multivendor-nfv-showcase/>





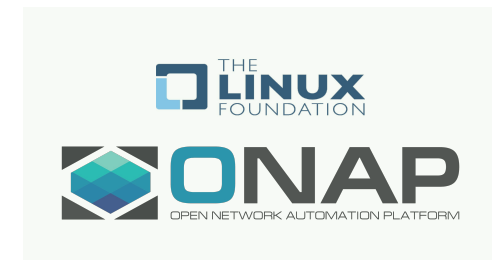
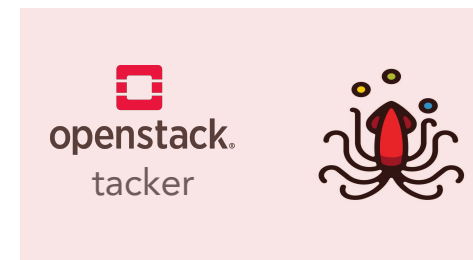
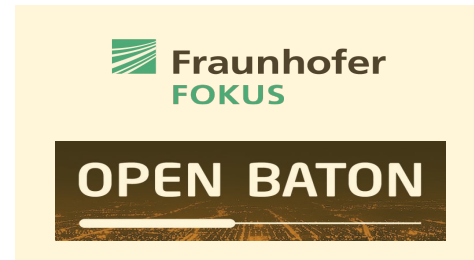
# Introduction to OSM Release SIX



# The NFV MANO Landscape

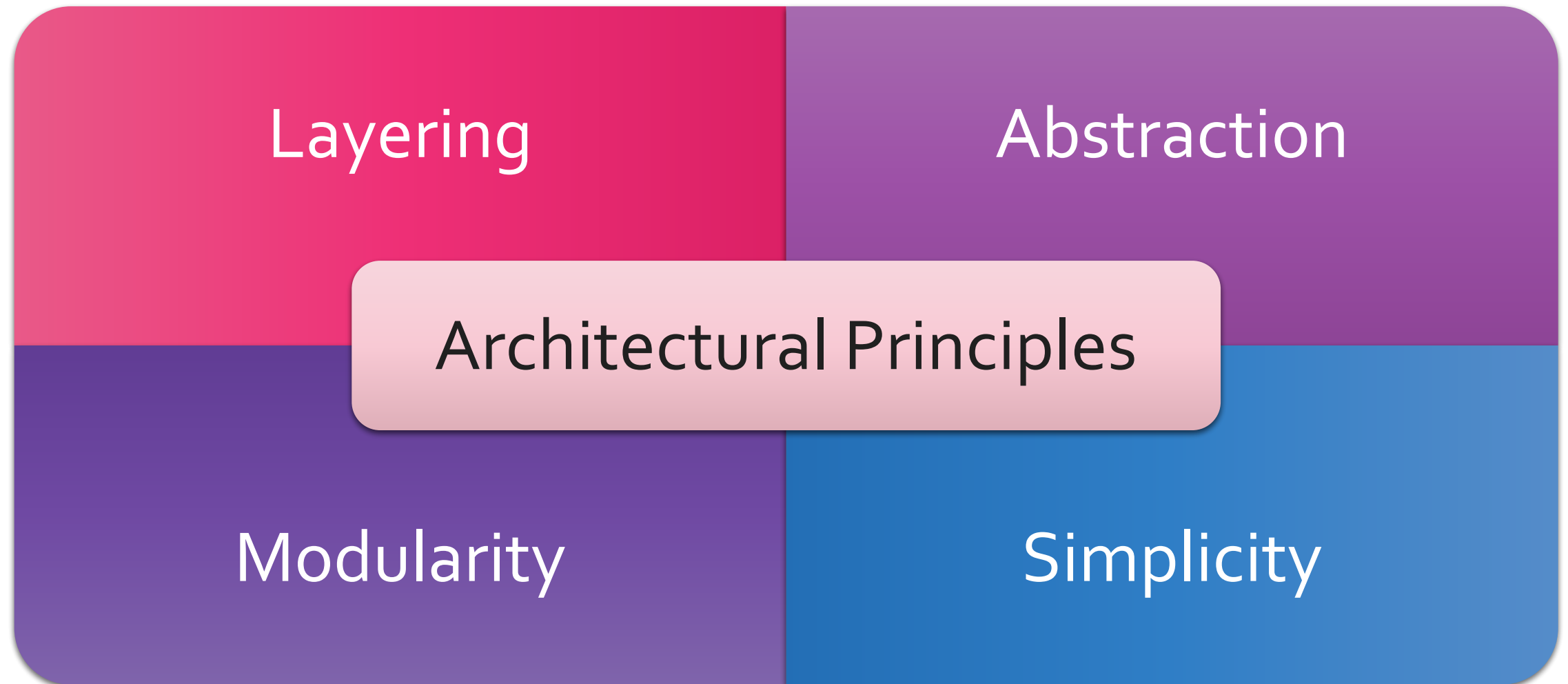


- Given that the VIM is already well covered by OpenStack distributions and proprietary solutions (e.g. vCD), in practice, **the “NFV MANO” part focuses on the VNF Manager and NFV Orchestrator.**
- Among the most popular open source platforms for NFV MANO, we have:



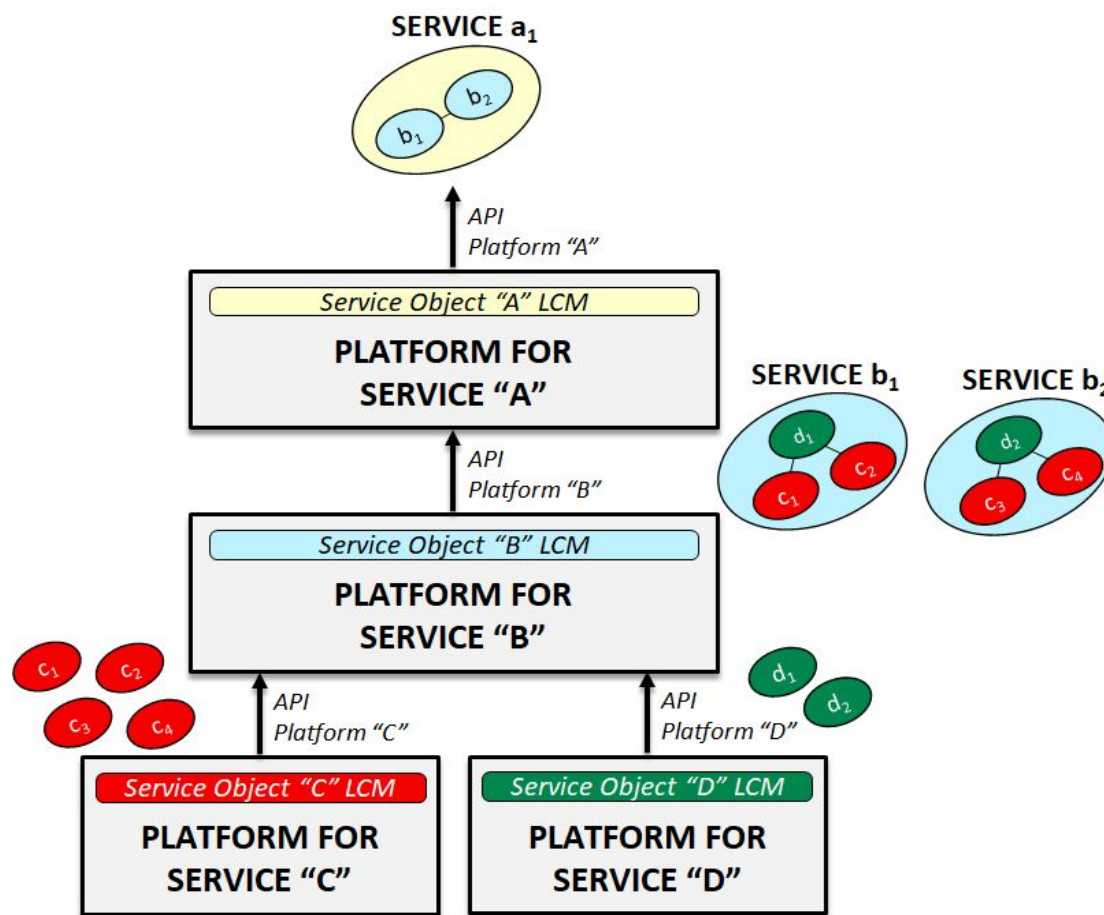


# OSM Architectural Principles



# Layered Model

A multilayered model, where each layer provides a “service object”, composed by service objects provided by lower layers.

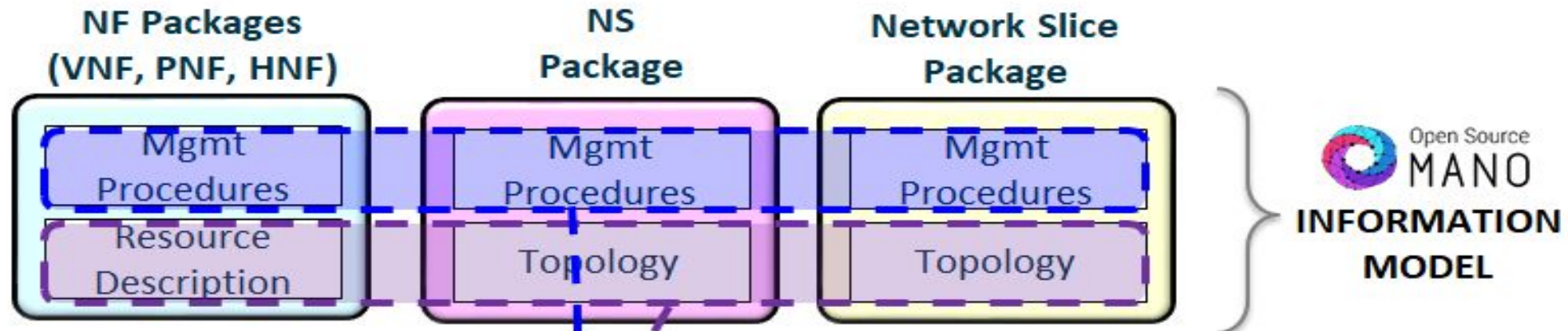




# OSM's approach aims to minimize integration efforts

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

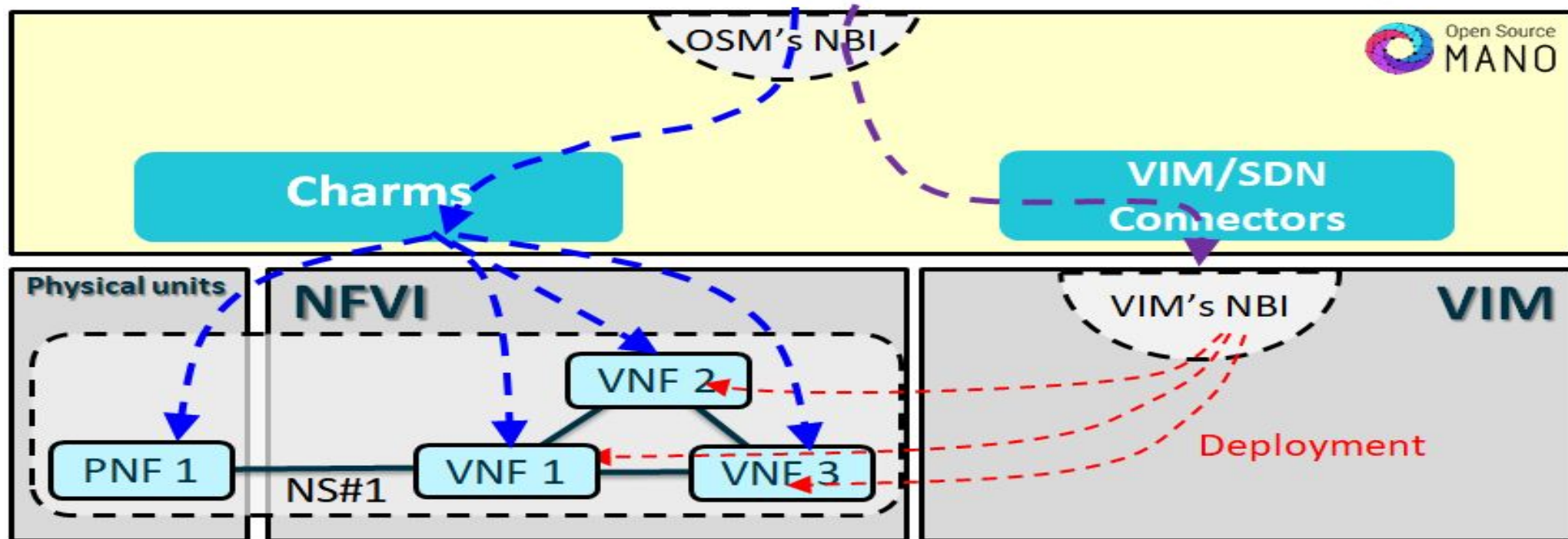
- VNFD (VNF Descriptor),  
VNFR (VNF Record),
- NSD (Network Service Descriptor),  
NSR (Network Service Record),
- NST (Network Slice Template),  
NSI (Network Slice Instance)





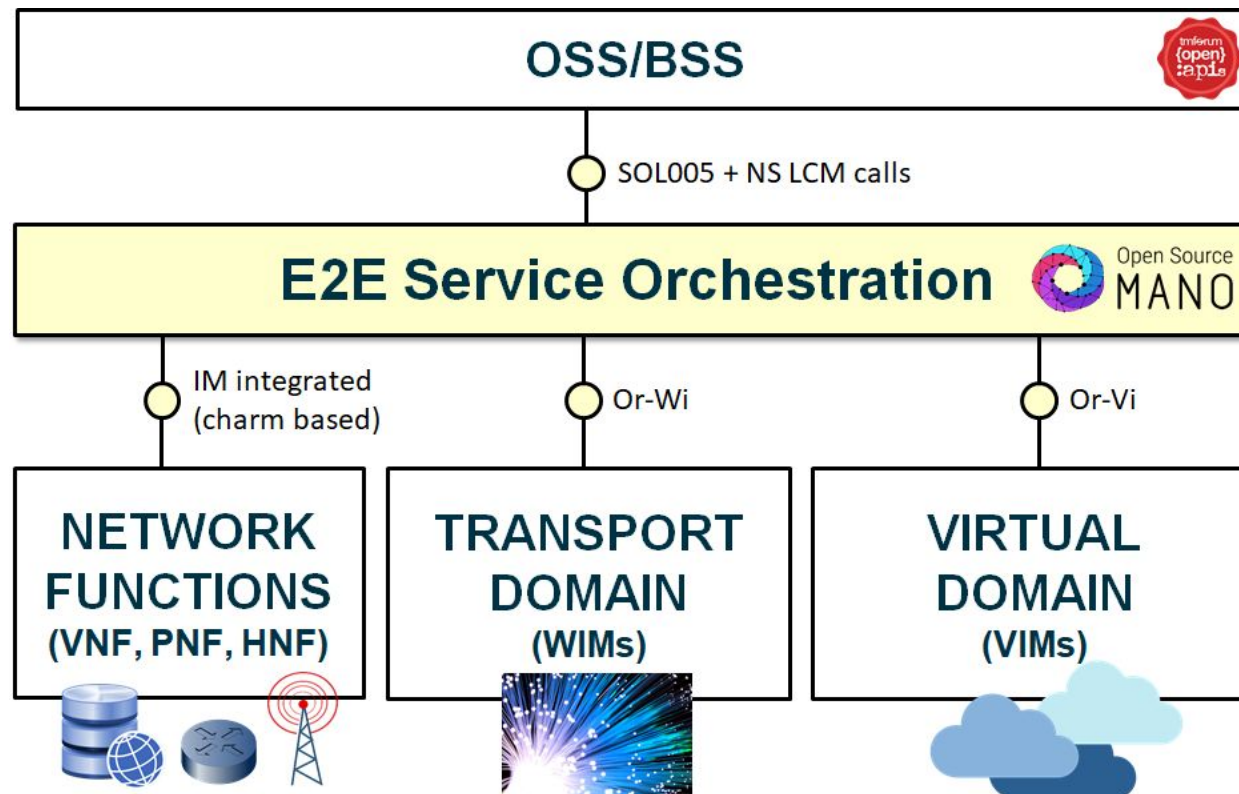
OSM's approach aims to minimize integration efforts

## 2. A unified northbound interface (NBI), based on NFV SOL005

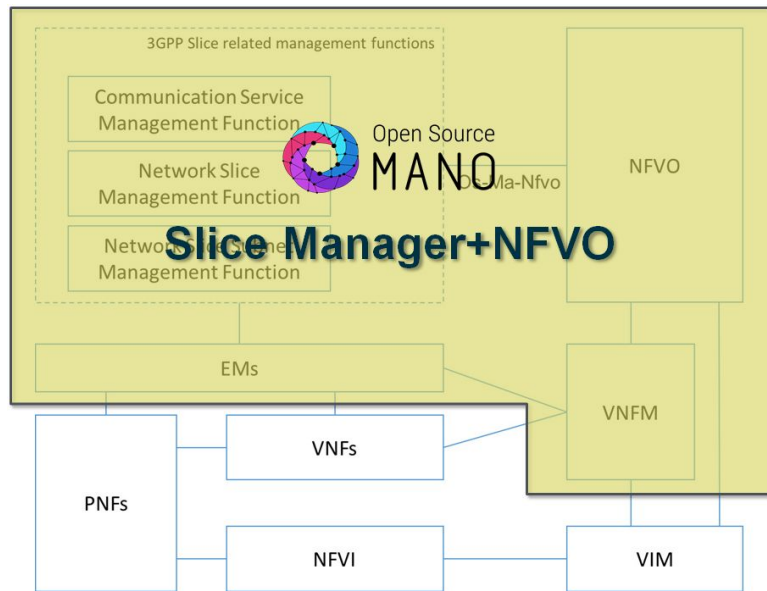


# OSM's approach aims to minimize integration efforts

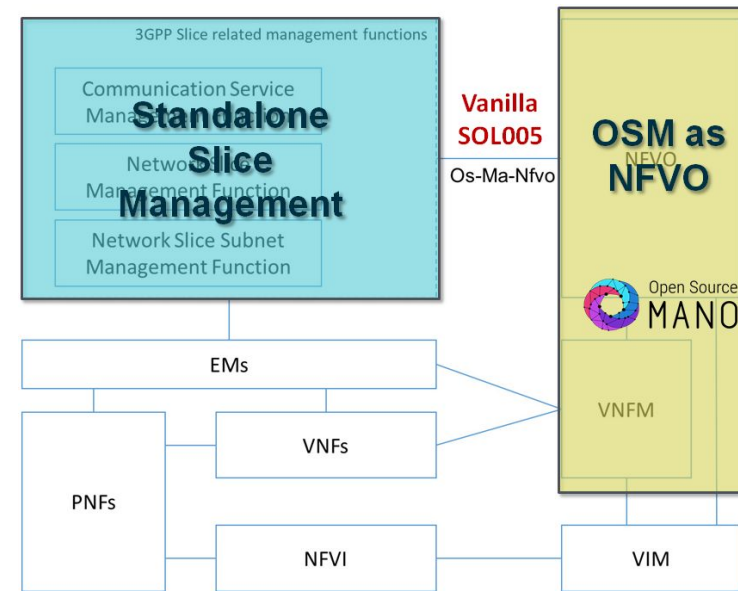
3. The extended concept of “**Network Service**” in OSM, so that an NS can span across the different domains identified and therefore control the full lifecycle of an NS interacting with VNFs, PNFs and HNFs.



4. In addition, OSM can also manage the lifecycle of **Network Slices**, assuming if required the role of Slice Manager, or integrating with an external Slice Manager



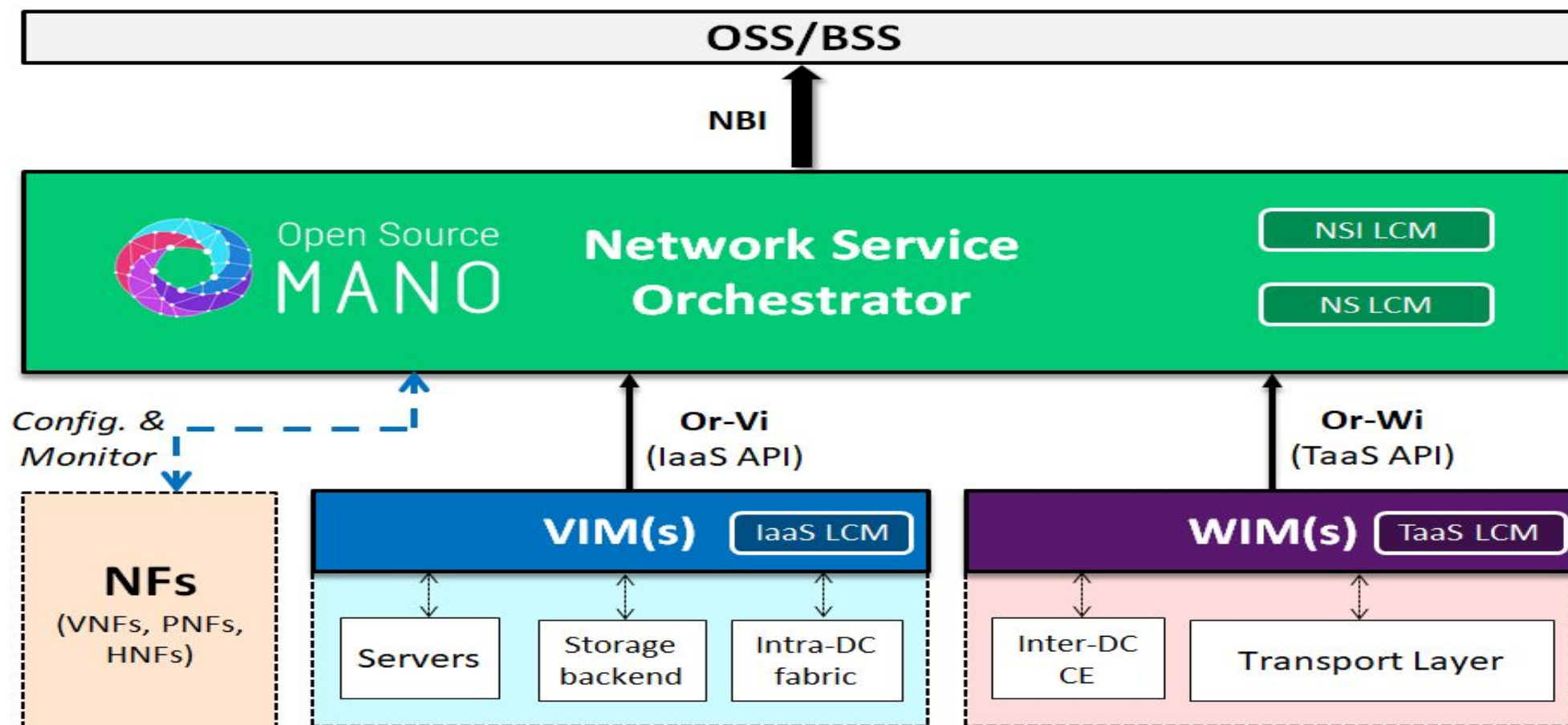
## Full E2E Management (Integrated Modelling)



## Standalone Management (*Vanilla NFV/3GPP*)

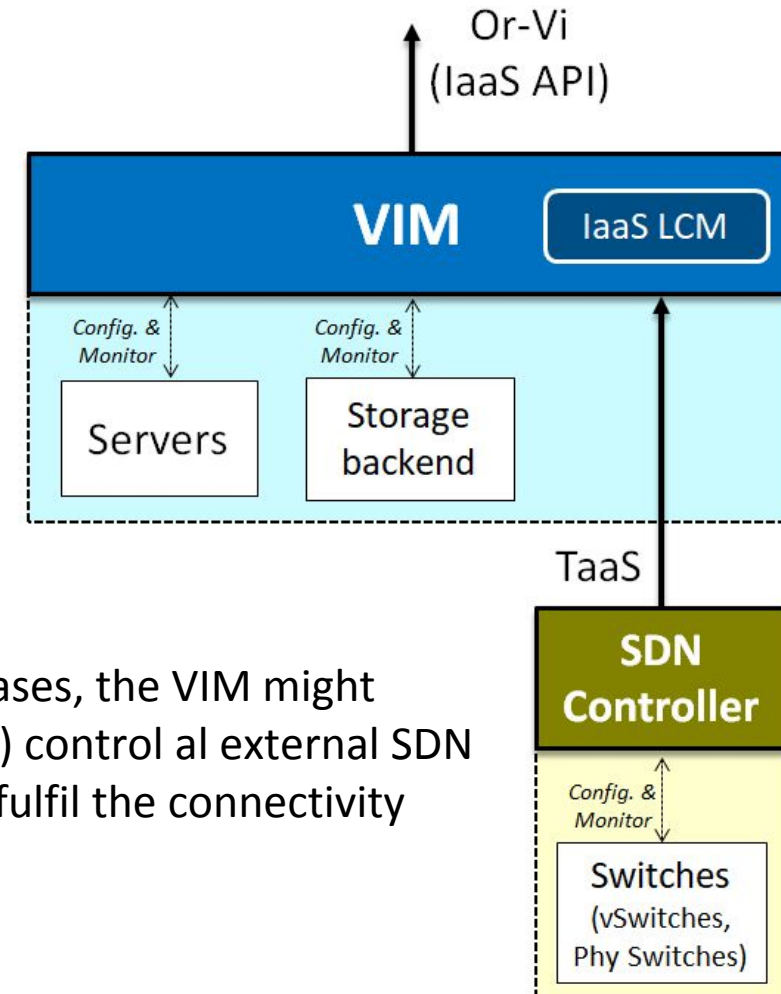
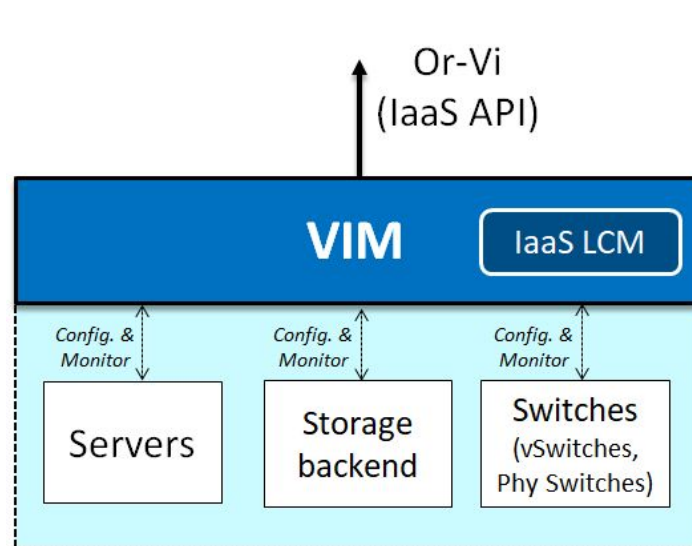
# Service Platform view

## OSM as a Network Service Orchestrator (NSO)



# Service Platform view

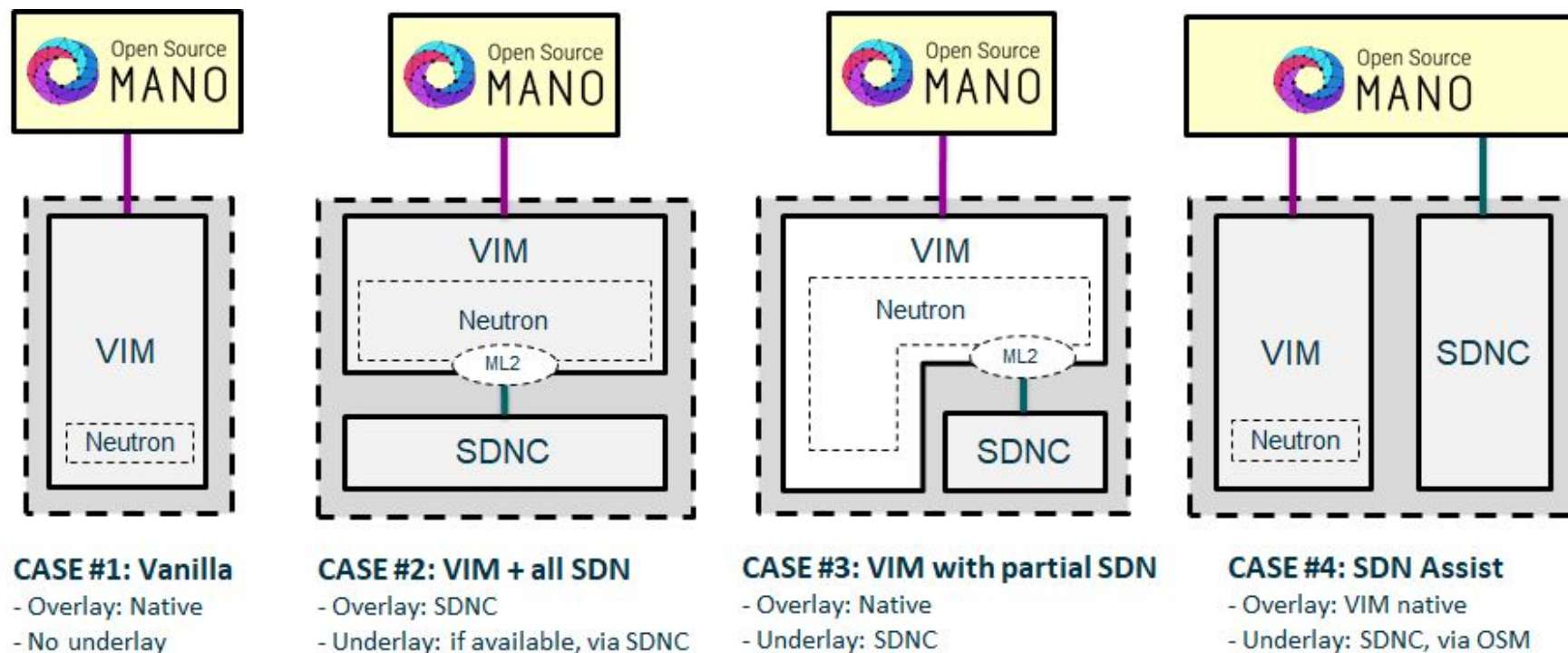
VIM manages the virtual network to support VNF's connectivity



In advanced cases, the VIM might (transparently) control an external SDN Controller, to fulfil the connectivity requirements



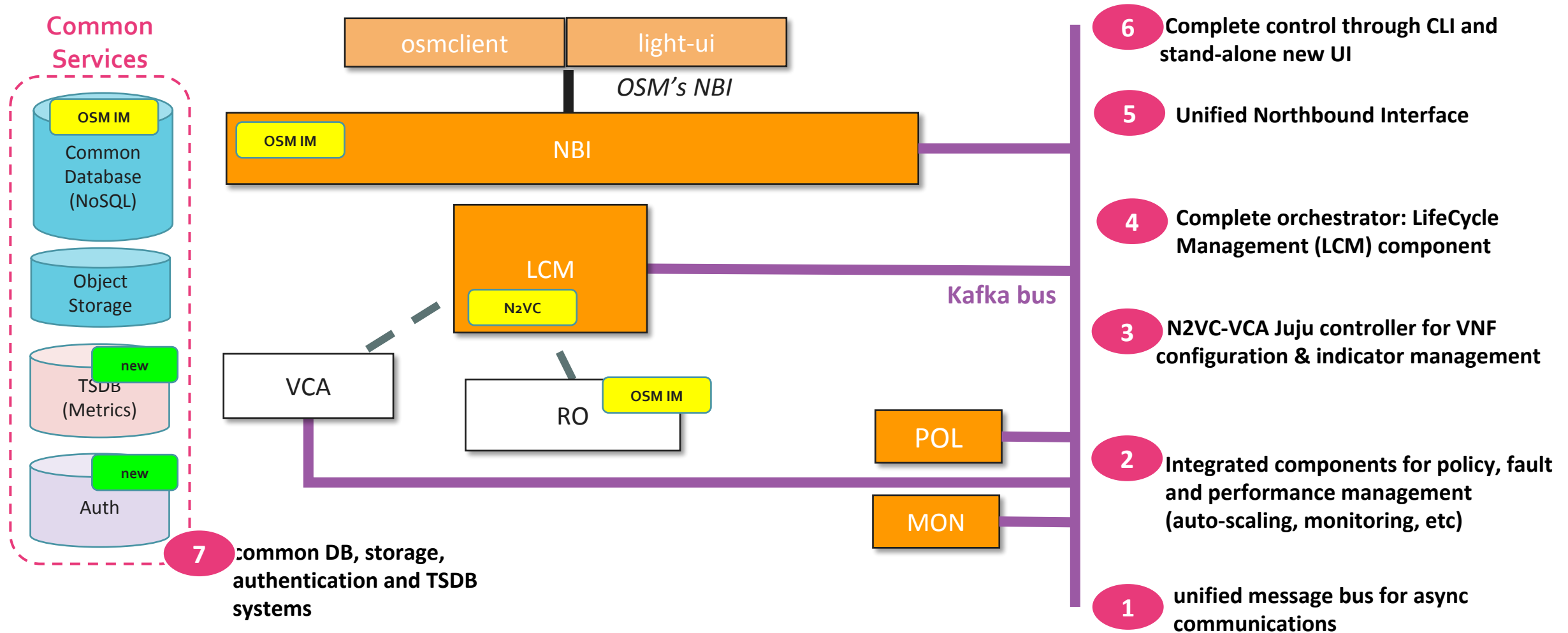
# Integration with SDN Controller



## SDN Assist

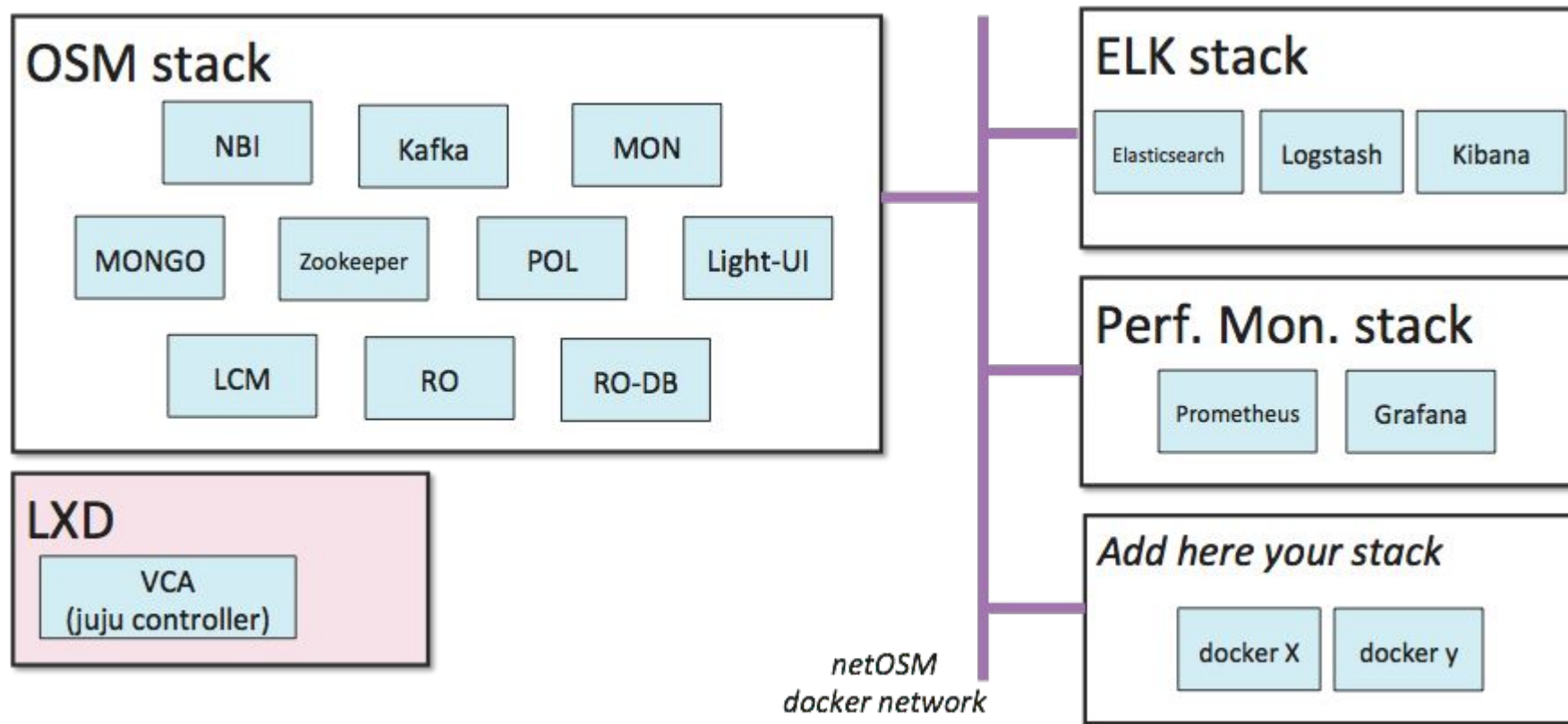
Allows OSM to control SDN connectivity, even when not possible by the VIM (eg: PCI Passthrough, SR-IOV)

# Release FOUR+ architectural view



# Release FOUR+ architecture

Microservice architecture to enable extensibility





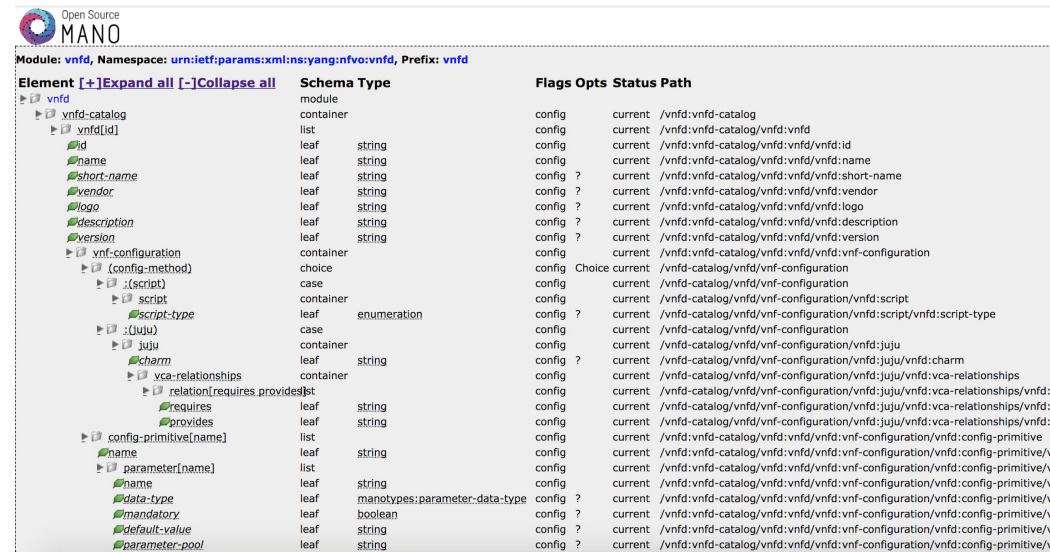
# Why is OSM Awesome?

## It has a rich and open information model

- Agnostic to VIM, SDN platform, VNF and OSS connectors/specifics.
- It allows for a uniform NFV orchestration, abstracted from the environment
- Aligned with ETSI-NFV Information Model

Visit:

[https://osm.etsi.org/wikipub/index.php/OSM\\_Information\\_Model](https://osm.etsi.org/wikipub/index.php/OSM_Information_Model)



Module: **vnfd**, Namespace: **urn:ietf:params:xml:ns:yang:vnfd**, Prefix: **vnfd**

Element	Schema Type	Flags	Opts	Status	Path
<b>vnfd</b>	module				
<b>vnfd-catalog</b>	container				/vnfd:vnfd-catalog
<b>vnfd[id]</b>	list			current	/vnfd:vnfd-catalog/vnfd:vnfd
<b>id</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:id
<b>name</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:name
<b>short-name</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:short-name
<b>vendor</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vendor
<b>logo</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:logo
<b>description</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:description
<b>version</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:version
<b>vnfd-configuration</b>	container			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vnfd-configuration
<b>(script:method)</b>	choice		Choice	current	/vnfd-catalog/vnfd:vnfd-configuration
<b>script</b>	case			current	/vnfd-catalog/vnfd:vnfd-configuration/vnfd:script
<b>script-type</b>	leaf			current	/vnfd-catalog/vnfd:vnfd-configuration/vnfd:script:vnfd:script-type
<b>juju</b>	case			current	/vnfd-catalog/vnfd:vnfd-configuration/vnfd:juju
<b>charm</b>	leaf			current	/vnfd-catalog/vnfd:vnfd-configuration/vnfd:juju:vnfd:charm
<b>vca-relationships</b>	container			current	/vnfd-catalog/vnfd:vnfd-configuration/vnfd:juju:vnfd:vca-relationships
<b>requires</b>	leaf			current	/vnfd-catalog/vnfd:vnfd-configuration/vnfd:juju:vnfd:vca-relationships/vnfd:requires
<b>provides</b>	leaf			current	/vnfd-catalog/vnfd:vnfd-configuration/vnfd:juju:vnfd:vca-relationships/vnfd:provides
<b>config-primitive[name]</b>	list			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vnfd-configuration/vnfd:config-primitive
<b>name</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vnfd-configuration/vnfd:config-primitive/vnfd:config-primitive/name
<b>parameter[name]</b>	list			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vnfd-configuration/vnfd:config-primitive/vnfd:config-primitive/parameter
<b>name</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vnfd-configuration/vnfd:config-primitive/vnfd:config-primitive/parameter/vnfd:config-primitive/name
<b>data-type</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vnfd-configuration/vnfd:config-primitive/vnfd:config-primitive/parameter/vnfd:config-primitive/data-type
<b>mandatory</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vnfd-configuration/vnfd:config-primitive/vnfd:config-primitive/parameter/vnfd:config-primitive/mandatory
<b>default-value</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vnfd-configuration/vnfd:config-primitive/vnfd:config-primitive/parameter/vnfd:config-primitive/default-value
<b>parameter-pool</b>	leaf			current	/vnfd:vnfd-catalog/vnfd:vnfd:vnfd:vnfd-configuration/vnfd:config-primitive/vnfd:config-primitive/parameter-pool

# Why is OSM Awesome?

It has a large and diverse community! More than **130** members!

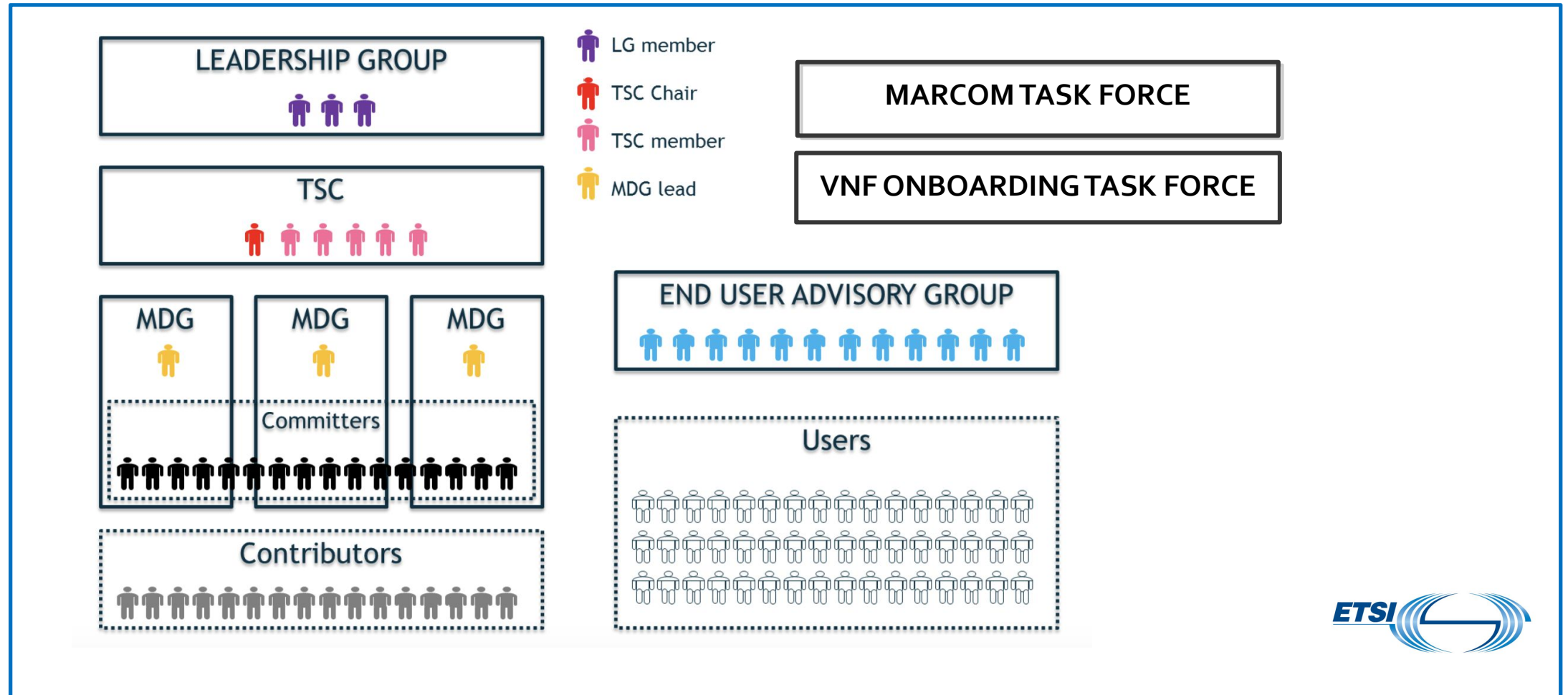


- 12 Global Service Providers
- Leading IT/Cloud players
- VNF providers



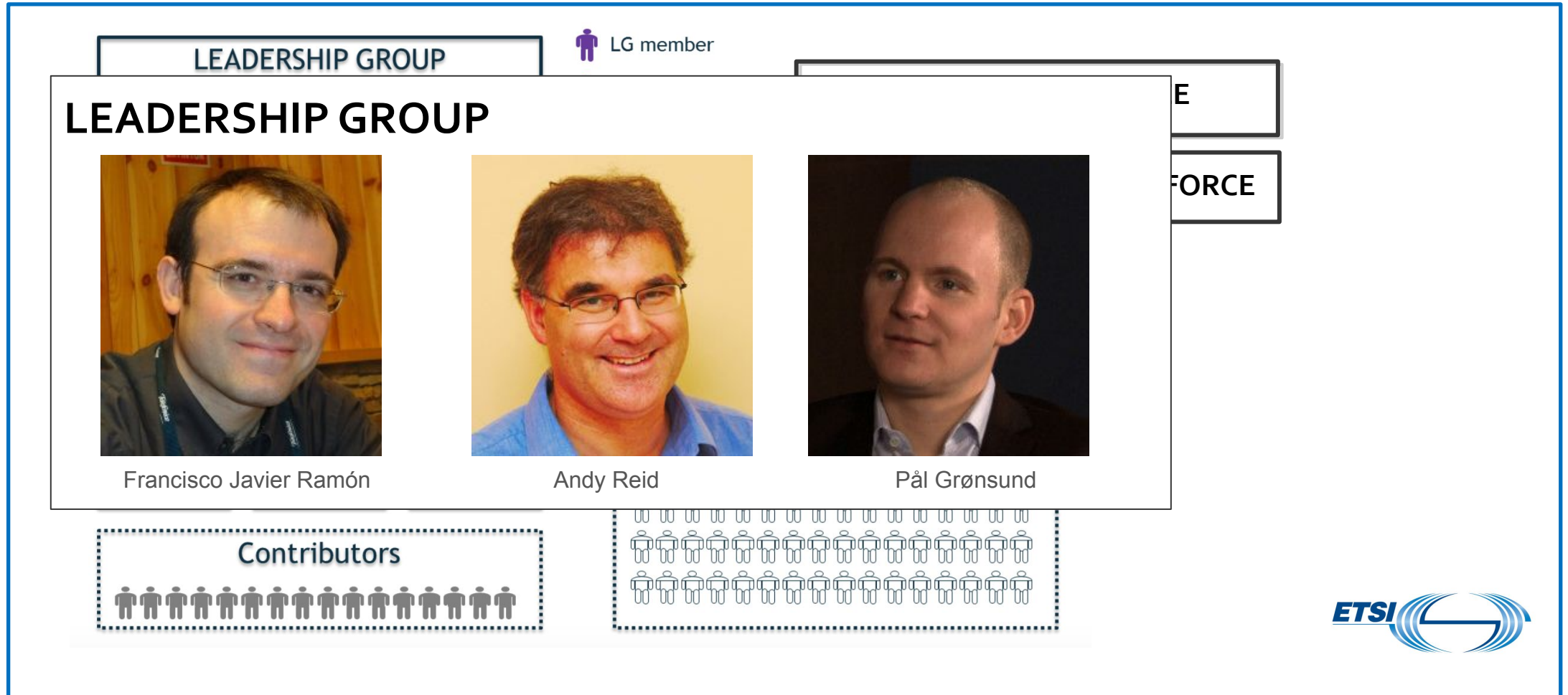
# Why is OSM Awesome?

It is well organized for producing production-ready upstream code



# Why is OSM Awesome?

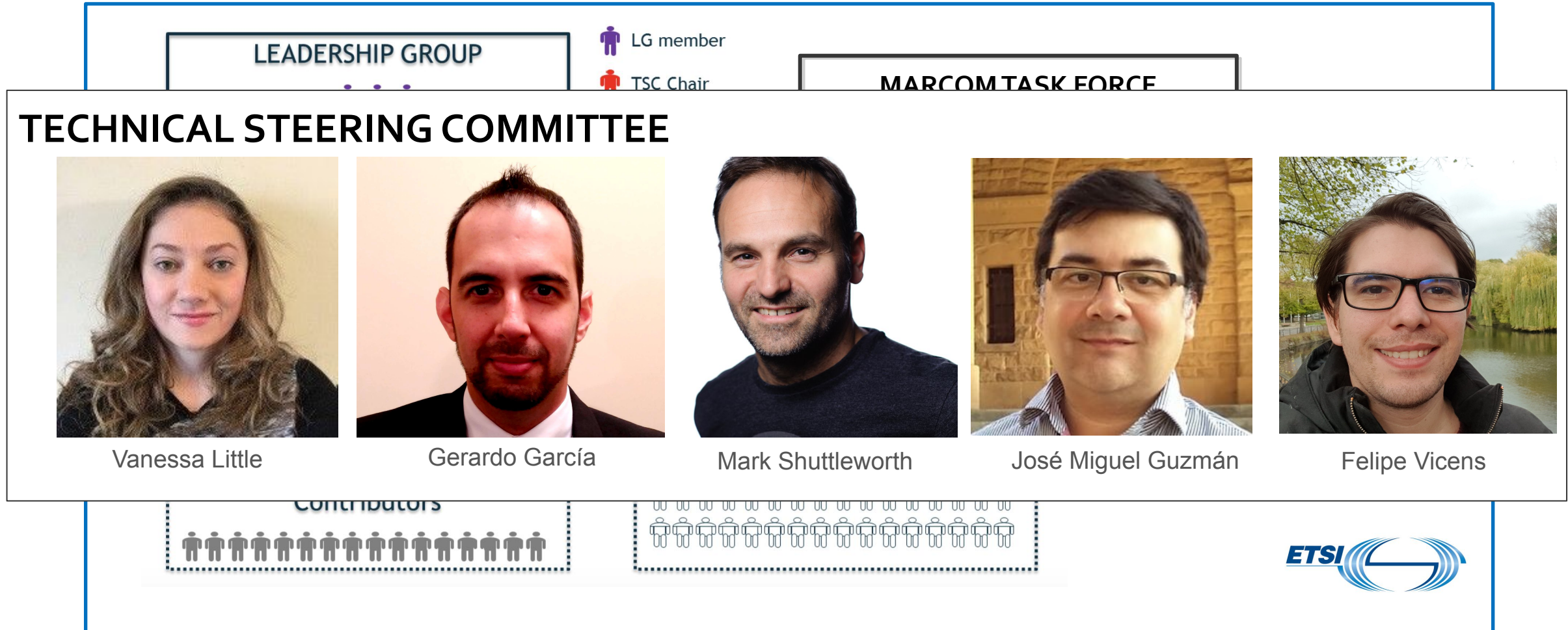
It is well organized for producing production-ready upstream code





# Why is OSM Awesome?

It is well organized for producing production-ready upstream code









# Why is OSM Awesome?

## END USER ADVISORY GROUP



Andy Reid

production-r

-  LG member
-  TSC Chair
-  TSC member
-  MDG lead

## MARCOM TASKFORCE



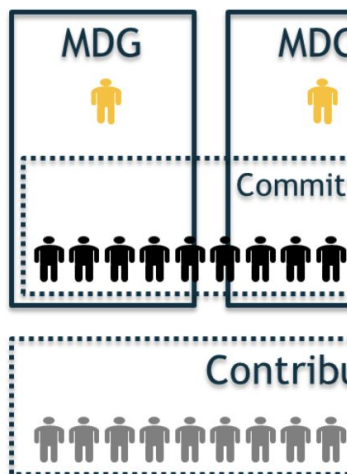
Mona Hrapkowicz

## END USER ADVISORY GROUP

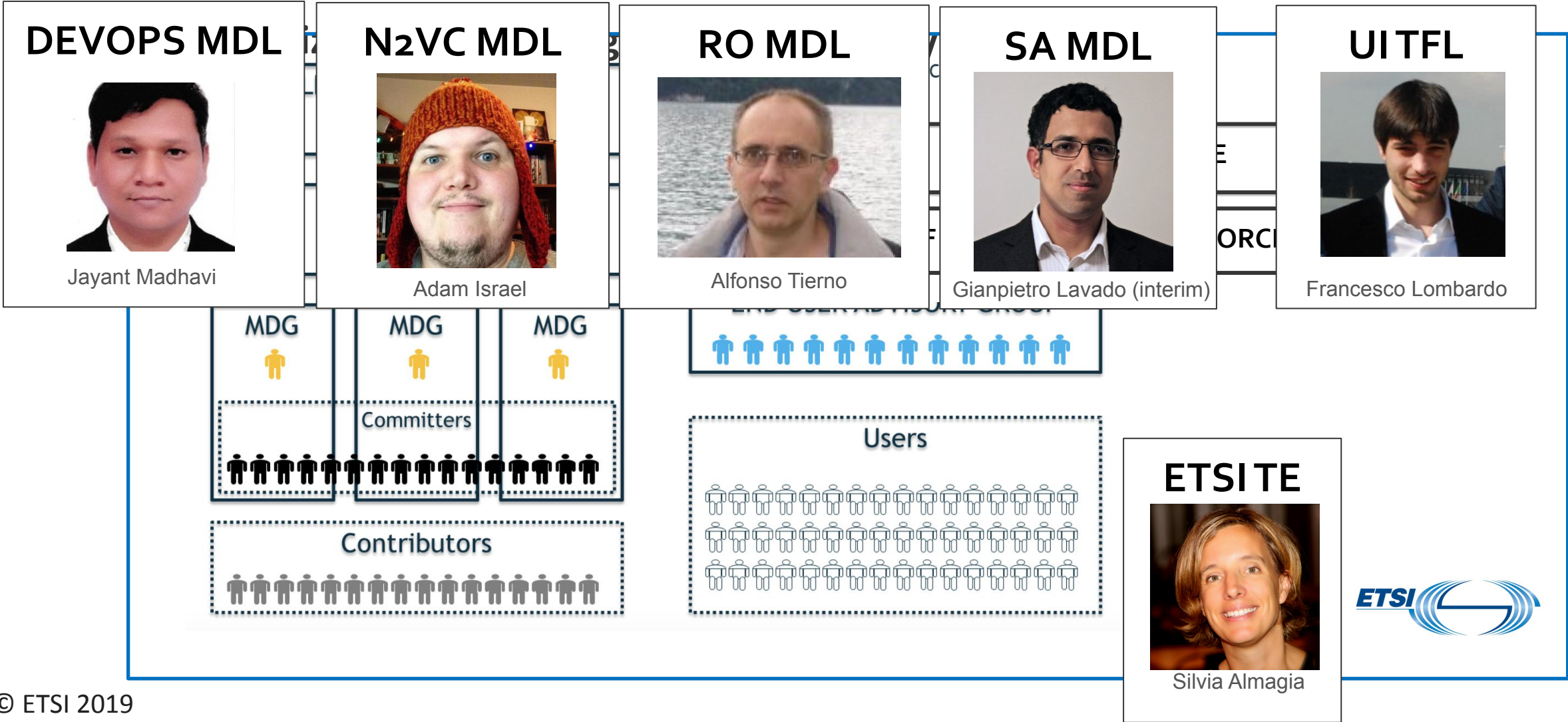
## VNF ONBOARDING TASKFORCE



Gianpietro Lavado

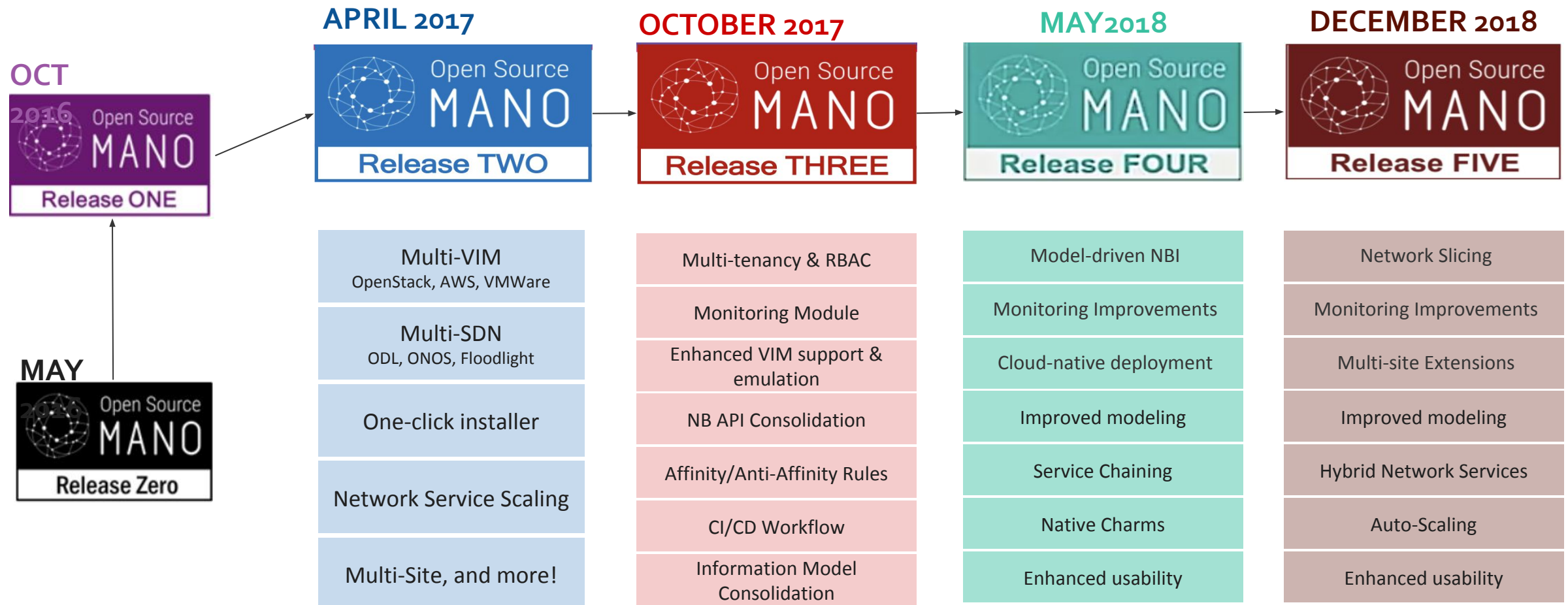


# Why is OSM Awesome?



# Why is OSM Awesome?

It prioritizes features for production readiness...



# Why is OSM Awesome?



**...and launched Release SIX in June 2019, with more key features for 5G and production environments!**

- Role-based authentication control (RBAC)
- Support for full/native charms for enhanced VNF management
- Network Slicing extensions
- Network-Service-level primitives
- Improved monitoring of infrastructure components and VNFs
- Enhanced support for instantiation parameters



# Why is OSM Awesome?

## And because other people say that OSM Rocks!

Table III: OSM vs ONAP resource footprint comparison.

Resource	OSM-4	ONAP-B
vCPU	2	88
Memory(GB)	8	176
Storage(GB)	40	1760
IP Addresses	1 static	20 Floating 3 static



Scan &  
Download

This work has been submitted to the IEEE for possible publication.

### On the Challenges and KPIs for Benchmarking Open-Source NFV MANO Systems: OSM vs ONAP

Girma M. Yilma, Faqir Zarrar Yousaf, Vincenzo Sciancalepore, Xavier Costa-Perez  
Email: {girma.yilma|zarrar.yousaf|vincenzo.sciancalepore|xavier.costa}@neclab.eu

**Abstract**—NFV management and orchestration (MANO) systems are being developed to meet the agile and flexible management requirements of virtualized network services in the 5G era and beyond. In this regard, ETSI ISG NFV has specified a standard NFV MANO system that is being used as a reference by MANO system vendors as well as open-source MANO projects. However, in the absence of MANO specific KPIs, it is difficult for users to make an informed decision on the choice of the MANO system better suited to meet their needs. Given the absence of any formal MANO specific KPIs on the basis of which a performance of a MANO system can be quantified, benchmarked and compared, users are left with simply comparing the claimed challenges of testing and validating MANO systems in general, and propose MANO specific KPIs. Based on the most popular open-source MANO projects, namely ONAP and OSM, using a complex open-source vCPE VNF and identify the features/performance gaps. In addition, we also provide a sketch of a test-jig that has been designed for benchmarking MANO systems.

and has specified interfaces and operations on its various reference points to support different functional features in its various specification documents. Fig. 1 provides a high level overview of the ETSI NFV MANO system functional blocks and the various interfaces defined on the reference points. The ETSI NFV MANO framework is also serving as a reference to other independent MANO projects that are being undertaken either by vendors or by open source communities. The latter is gaining a lot of prominence and attention from operators due to the diverse efforts that are being expended towards developing open source MANO platforms.

#### A. Problem Statement

Open source MANO projects such as ONAP [3], OSM [4], Open Baton [5], Cloudify [6], OPNFV [7], are under different stages of steady development. All are competing to make their mark in the operators' infrastructure but, owing to the complex nature of the NFV MANO system itself, no project to date can claim to support the entire LCM spectrum of the NFV assets or be ready for field operations. More prominent among these projects are Open Network Automation Platform (ONAP) and Open Source MANO (OSM), which have gained a lot of attention from the operators' community, especially because of the patronage of some big operators behind the development of ONAP and OSM. For instance ONAP, which is being developed under the umbrella of the Linux Foundation, is mainly supported by AT&T, whereas OSM is driven by Telefonica and is being developed under the mandate of the newly formed ETSI Open Source Group (OSG).

Both ONAP and OSM are under different stages of their releases but they are far from being complete or stable. Both are aiming to provide an integrated NFV MANO framework, but they are following very different directions in terms of architecture and implementation. There are still gaps between what is being claimed and what features and functionalities are actually supported. There are ambiguities in terms of their deployment footprint as well as operational efficiency for providing carrier-grade management to NFV services. Owing to the fact that these are relatively latest developments, there is very much less information and experience available in terms of the functional and operational capabilities of these platforms and technology readiness level (TRL).

Moreover, carrying out MANO systems

#### I. INTRODUCTION

**A**GILITY and flexibility for the management of the network resources and services represents one of the key innovations of 5G networks to support carrier-grade operations for different verticals with diverse service requirements at reduced CAPEX/OPEX costs. In this context, Network Function Virtualization (NFV) has been widely accepted as a technology enabler for addressing the challenging requirements of 5G networks [1]. The key concept of NFV is the decoupling of the network functions from the underlying hardware platforms, while the network functions are realized as a virtualized entity commonly referred to as Virtualized Network Functions (VNFs). VNFs can embody less complex network functions such as Firewall (vFW), load balancer (vLB) to more complex functions such as Evolved Packet Core (vEPC), Customer Premises Equipment (vCPE) to name a few. End-to-end Network Services (NS) are composed by chaining relevant VNFs over Virtual Links (VL).

The introduction of NFV technology has great implications on the network management systems where they need to be extended to provide Life Cycle Management (LCM) of VNFs, NSs and VLs beyond the traditional FCAPS (Fault, Configuration, Accounting, Performance, Security) management services. The LCM actions include operations such as on-boarding, instantiation, scaling in/out/up/down, migration, update/upgrade, etc of a VNF and its associated components. In this regard the ETSI ISG NFV has proposed a framework for NFV Management and Orchestration (MANO) systems.





# Contributing to the Community



# Joining the OSM Community

- Join [here](https://osm.etsi.org/about/how-to-join) as a company or individual contributor!  
<https://osm.etsi.org/about/how-to-join>



## HOW TO GET INVOLVED IN OSM

There are two paths to get involved in OSM as an organisation: as an ETSI Member, or as an OSM Participant.

Check first if your organization is already involved by consulting the list of [OSM Members and Participants](#).

### Get involved as an ETSI Member

To take part in the development of OSM and participate to the meetings, ETSI Members need to sign the [OSM Membership Agreement and CCLA](#). In doing this, they agree to the OSM operating rules which in some cases are different from those in ETSI's Technical Working Procedures. [Check if your company is an ETSI Member](#).

### Get involved as an OSM Participant

Organizations who are not members of ETSI may also participate in OSM, attend meetings and help to develop OSM by making technical contributions. They are not applicable for leadership (LG) positions and must pay a participation fee to attend OSM meetings. To get involved as a Participant, please sign the [OSM Participant Agreement and the CCLA](#).

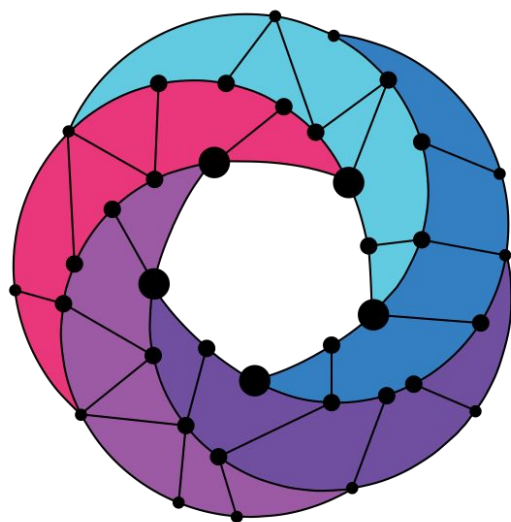
## Developers and Users

Individual developers and end users are welcome to contribute code and feedback to OSM, they just need to [create an individual contributor or user account](#).

- **Weekly Conference Calls**
  - Technical, MARCOM, DevOps, SA, and more!
- **Face to Face Meetings**
  - Plenaries and Mid-Release meetings (every 3 months)
  - Next location: Madrid (March 2020)
- **OSM Hackfests**
  - Co-located with OSM Face-to-Face meetings.

# Ways to contribute to OSM

- **Try OSM** and give feedback to the community.
- Join as a developer to **make contributions to the code**.
- Join the community to **contribute to design discussions**.
- **Start building your own distribution** of OSM as an integrator.
- **Host an OSM meeting** to contribute to the community's growth in your area.



# Open Source MANO

***Thanks !***

José Miguel Guzmán (Whitestack)  
[jmguzman@whitestack.com](mailto:jmguzman@whitestack.com)