A Pattern for NFV Management and Orchestration

Ahmed M. Alwakeel , Florida Atlantic University
Abdulrahman K. Alnaim, Florida Atlantic University
Eduardo B. Fernandez, Florida Atlantic University

Network Functions (NF) normally require a tightly coupled infrastructure making the process of enhancing and expanding the network difficult. Network Function Virtualization (NFV) produces the functions of the network through virtualization. In NFV, different networking components such as firewalls, load balancers, and IDS are provided as a service and rely purely on the cloud, which makes expanding and upgrading the network an easy and fast process. In NFV, the Network Functions Virtualization Management and Orchestration (MANO) unit works as a backbone for the services, taking care of managing and orchestrating the different resources needed by the virtual network function to be implemented. We present here a pattern for an NFV MANO.

Keywords
virtualization, architecture patterns, network functions, security patterns, cloud computing, telecommunications, Management and orchestration, NFV MANO.

1.INTRODUCTION

The European Telecommunications Standards Institute (ETSI) introduced NFV as a new concept to provide network functions (NFs) as a service to the consumers in a virtual manner as an alternative to the traditional legacy way of implementing NFs. In legacy network systems Telecommunication Service Providers (TSPs) have to deploy the necessary network functions as proprietary hardware at every consumer's premises making deployment and management of these component a difficult process, requiring a lot of space and power. Moreover, the life cycle of such systems is getting shorter due to the acceleration of hardware development. This leads to increases in the Capital Expenditure (CapEx) and Operating Expenditure (OpEx) for TSPs [Haw14]. However, these challenges can be overtaken with the deployment of Network Function Virtualization (NFV) and finally provide users with Network Services (NSs) network services contains several VNF packages that provided to the user in a whole as one package to be consumed as a service.

NFV uses the power of the cloud to virtualize different NFs, such as firewalls, Domain Name Systems (DNS), and load balancers. Instead of providing these services with hardware, NFV builds network systems with software that virtualizes the services provided by the hardware components. This approach leads to reduction in the complexity of network design as well as allowing better and faster scalability and deployment of the network to the users. *However, such system is complex and require management and orchestration due to the interaction of several components that could be distributed; not only that but it also could be provided by different vendors. All the combined resources that create the service have to be managed to make the integration of old and new components possible.*

The general architecture of NFV modeled by ETSI (Figure1) consists of three main components [Ets14]:

· Network Function Virtualization Infrastructure (NFVI)

NFVI can be considered as a cloud data center that works as a foundation platform for NFV; it contains both hardware resources as well as the virtual resources which together build up the infrastructure on which VNFs are executed, deployed and managed. NFV contains three main components which are virtualized resources, virtualization layer and hardware resources.

· Virtualized Network Functions (VNFs)
VNFS are software packages that represent the actual implementation of the network functions. A VNF may contain one or more functions, even an entire service could be encapsulated in one single VNF to be provided to consumers to reduce complexity of implementation [Ets14].

· NFV Management and Orchestration (MANO)
The MANO unit could be considered as the heart of the NFV system. This unit takes care of the management and orchestration aspects of the entire set of VNFs. VNF management functions Include traditional fault management as well as security management. Moreover, the MANO takes care of creating VNFs, scaling VNFs by increasing or decreasing the capacity of the VNF, update or upgrade VNFs, and terminate VNFs. Also, it takes care of fault and performance management aspects of the system. On the other hand, Orchestration tasks differ than management tasks in that some of the orchestration tasks are registering network services into the catalog, monitoring the capacity of network services that the TSP provides to the consumer and validating and verifying resource requests.

A MANO includes three main components which are: NFV orchestrator (NFVO), VNF manager (VNFM), and virtualized infrastructure manager (VIM). These three components as well other components in the NFV architecture interact with each other through reference points.

In this paper we present a pattern for the NFV MANO unit. Our audience includes system designers, system architects and TSPs.

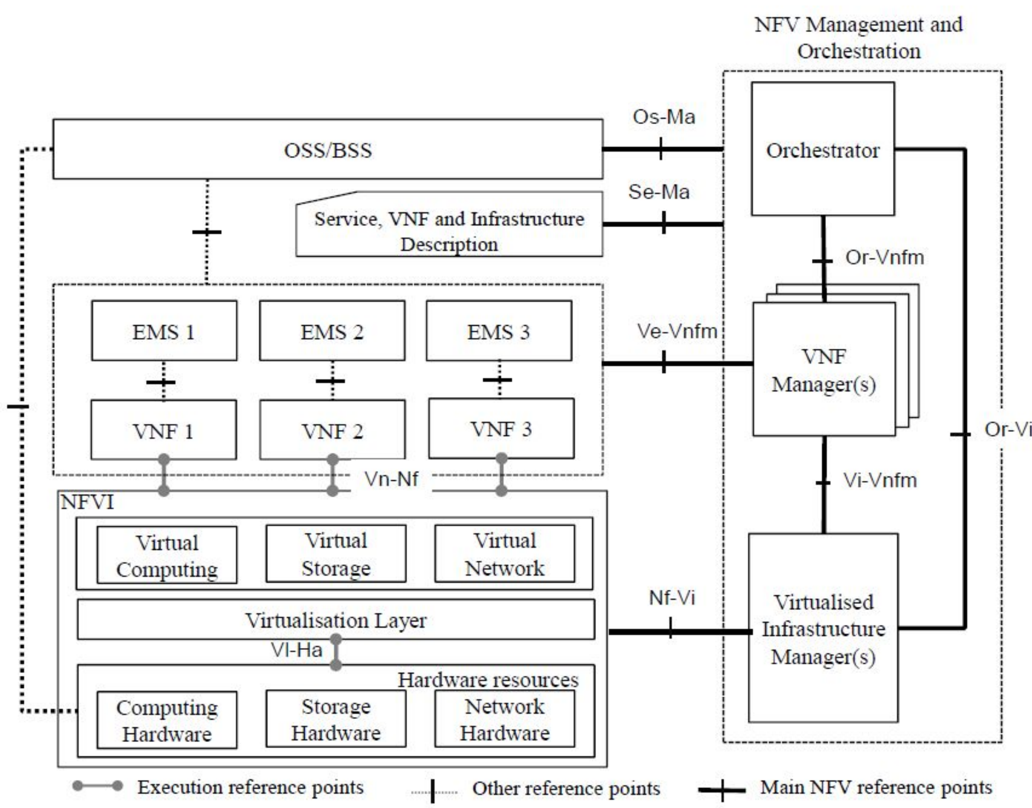

Figure 1 ETSI NFV architectural framework

2. MANO PATTERN

2.1Intent

MANO unit is an architectural block for NFV that addresses management and orchestration of the entire NFV service. This includes computing, networking, and virtual resource management, and network life cycle monitoring.

2.2 Example

Dave is subscribed to an NFVs service that provides him with twenty ports, four load balancers, one QoE measurement tool and finally a firewall to ensure the security of the network. However, Dave's business grows and he has decided to extend his office so he needs to scale up his network and add another twenty ports which will also lead to adding four more load balancers. He requested them from the TSP. How can the TSP provide them and scale up the current service?

2.3 Context

Different components in the NFV architecture such as physical resources and virtual services (VNFs) have to be managed and assigned to network users in a proper way. Together they provide Network Service (NS) to the user. We can define Network Service (NS) as a package of network functions virtualized to the user to provide network functionality similar to the one the user consume using legacy network systems. In NFV these network packages are called network graph. Moreover, different services have to be visible to users in a network services catalog that gets updated regularly according to the available resources in the pool in order for the user to choose from it.

2.4 Problem

NFV contains different functions from different vendors that have to be managed. This includes resource allocation in the NFVI which could be very complex since different requirements and constraints have to be met at the same time. Also, the allocation and release of resources is a dynamic process that needs to be fast and effective in response to their consumption. The solution to the management of NFV functions is affected by the following forces:

Flexibility: we would like to run different types of network services because the consumer has different needs.

Isolation: users shouldn't be able to have access to system functions or have access to functions provided to other users whether by error or intentionally.

Security: The created network graph should be secured from unauthorized access.

Scalability: increasing the number of units in NFV may introduce integration problems due to the variety of interfaces and protocols. We need to assure their smooth integration.

Modularity: NSs should have a well-defined interface in order for the user or another subsystem to interact with it and to improve interoperability.

Extensibility: We would like to be able to add new functions and features because of new needs of new consumers.

Management: We need to be able to handle the lifetime of a large number of networks with different requirements.

Performance: We need to provide to the consumer a satisfactory level of performance.

2.5 Solution

MANO contains components for managements of the system allowing the NFV components to interact with each other in the NFV system. The MANO manages virtualized resources as well as hardware resources including updating, scaling and operating the resources. Moreover, it takes care of managing the virtual functions and the life cycle of VNFs. The MANO also handles the emulation of hardware resources and their presentation as a software entities. Also, The MANO contains different catalogs for the resources that together create NSs.

2.5.1 Structure

Figure 2 shows the class diagram for the MANO pattern in dotted lines as well as its interaction with other components in the system.

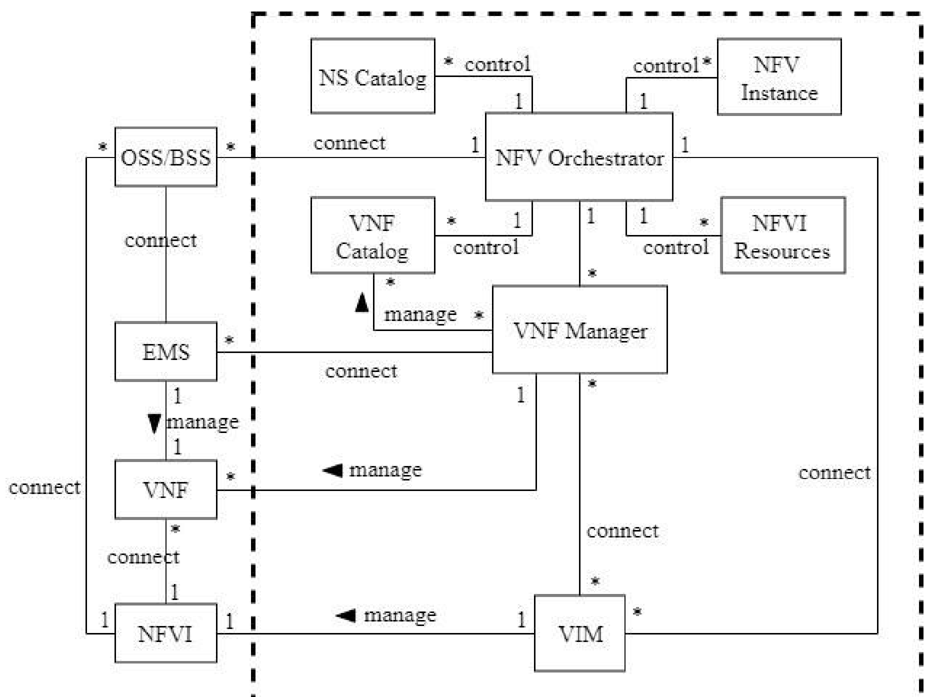

Figure 2. Class diagram for the structure of MANO unit

- VNF Manager takes control of Each VNF in the system as well as managing the VNF catalog.
- VIM class takes care of everything related to network function virtualization infrastructure including hardware resource management and monitoring.
- The NS catalog, NFV instance, VNF catalog and NFVI resources are all repositories that contain metadata about different components that could be used to build an NFV service.

- NFV orchestrator class takes care of the coordination and management of different repositories that provide resources to create network services.
- Four different repositories connected to NFV orchestrator which are NS catalog which contain a list of usable network services as well as a template for possible services to be provided to user ,NFV instance which holds all details about network services instances and how its related to VNF instances,VNF catalog which contain description of VNF deployment and operational processes and NFVI resources which contain all the resources available to the system to establish NFV service.
- VNF Manager is class to manage VNF through its entire life cycle.
- VIM is connected to Network function virtualisation infrastructure (NFVI) which take care of the following
    - Fault and performance management of hardware, software, and virtual resources.
    - Manages the life cycle of virtual resources in an NFVI domain.
    - Monitor the inventory of virtual machines and which physical resources related to it.
    - Manage the emulation of physical resources into virtual resources.

2.5.2 Dynamics

MANO use cases include “Requesting NFV network”, “Modify NFV network”, “Migrate NFV service” as well as other use cases. In this section we cover three use cases which are “Requesting an NFV network” “Terminating an NFV networks” and “Modifying an NFV service”.

UC1: Request NFV network
Summary: the NFV consumer requests a network, and in turn the NFV orchestrator handles the request.
Actor: NFV orchestrator.
Precondition: the NFV consumer has a valid account with the NFV provider.
Description:

1. The NFV orchestrator receives an NFV network request from a consumer.
2. NFV orchestrator checks available resources in the NFVI resources repositories
3. NFV Orchestrator forwards request to the VIM.
4. VIM forwards the request to NFVI. .
5. The NFVI sets up the service and create a VMs to host the service.
6. NFVI confirms network creation to the VIM.
7. VIM confirms network creation to NFV orchestrator.
8. NFV Orchestrator assigns service ID to the user ID which requested it.

Postcondition: a new NS will be created and assigned to the user requested it.
Exception the requested network cannot be provided due to lack of resources. After the NFVI receive the request and before creating the network graph it has to check for the resources and inform the customer with the result.

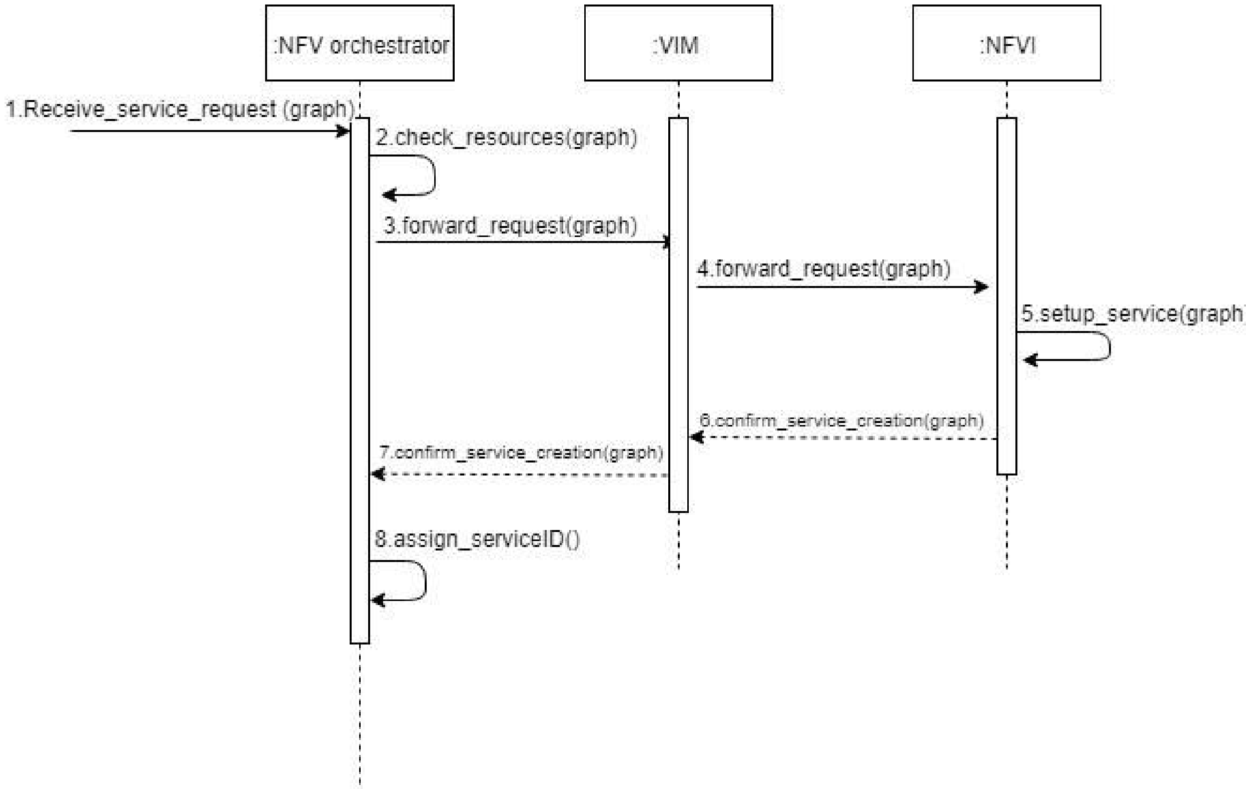

Figure 3. Sequence diagram for requesting a service.

UC2: Terminate an NFV service
Summary: the NFV consumer requests a termination of service, and in turn the NFV orchestrator handles the request.
Actor: NFV orchestrator.
Precondition: the NFV consumer has a working network.
Description:

1. NFV orchestrator receives a service termination request from a consumer.
2. NFV orchestrator forwards the request to VNF manager.
3. VNF Manager terminates VNFs related to the network; this step is repeated until all VNFs services are terminated.
4. VNF manager notifies VIM of NFV service termination.
5. VIM notifies NFVI of NFV service termination.
6. NFVI releases resources of NFV service.
7. NFVI notifies the VIM of the release of resources.
8. VIM notifies the NFV orchestrator of the release of resources.
9. NFV orchestrator updates repositories.

Postcondition: The network will be terminated and the resources of the network will return to the resources pool.

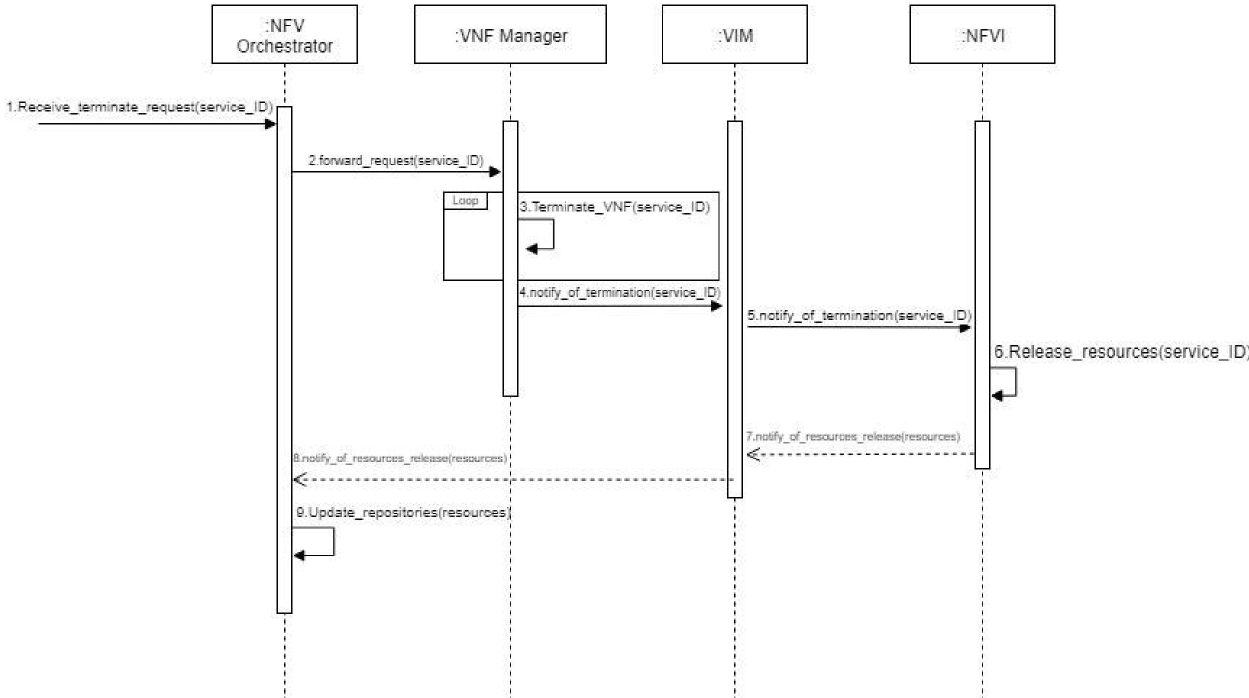

Figure 4. Sequence diagram for service termination.

UC3: Modify an NFV service

Summary: the NFV consumer requests a modification of network service that he/she already have, and in turn the VNF manager handles the request.

Actor: VNF Manager.

Precondition: the NFV consumer has a working network service.

Description:

1. VNF manager receives network service modification request from the user.
2. VNF manager forwards user request to NFV orchestrator.
3. NFV Orchestrator checks available resources in the resources repository.

This use case has two alternative scenarios:

A-[Resources Available]

4. NFV Orchestrator forwards the request to VIM.
5. VIM forwards the request to NFVI.
6. NFVI modifies the network service according to the user needs.
7. NFVI confirms network service modification to VIM.
8. VIM confirms network service modification to NFV orchestrator.
9. NFV Orchestrator confirms network service modifications to NFV manager.

B-[Resources unavailable]

4. NFV Orchestrator Notifies VNF manager of unavailability of resources

Postcondition:
A- The user network service will be modified to meet his needs.
B-The user will be notified that there isn’t available resources to fulfill his request.

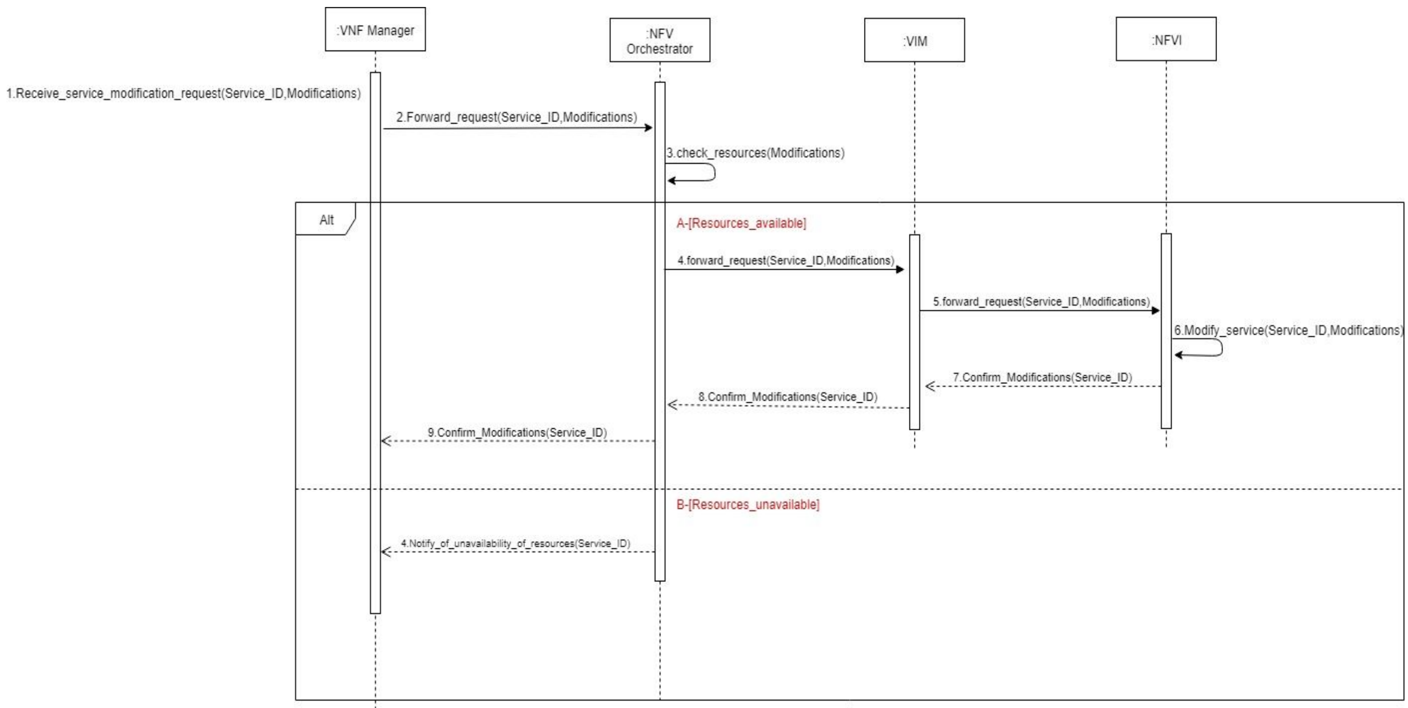

Figure 5. Sequence diagram for *Modify an NFV service*

.

2.6 Implementation

A TSP that implements VNFs can either create their own MANO unit or it can be rented from other providers to be composed with the NFV architecture that TSP provides. The MANO should provide management for the different entities and components of the NFV. VNF management unit which is a part of MANO controls and access all the VNFs in the system at all times to provide managements tasks including fault and performance management. A Virtualized Infrastructure Manager is also created to have access to the hardware resources and takes care of decoupling and transforming these hardware resources into virtual resources in order to be used to build VNFs. The internal components of MANO are interconnected with each other through different reference points, reference points also connect external NFV components to the MANO [Ets13,Ets13-2].

A good implementation of MANO is Open source MANO which is created by ETSI with the help of different contributor in the industry. Open MANO allows the entire infrastructure of NFV to be deployed and configured in minutes to avoids the need for expensive and time consuming customization [Int16]. Open MANO have the following advantages [Osm18]:

- Automated end to end service orchestration.
- A plugin model that allows integration of multiple NFVIs with different hardwares.
- Fault and performance management of NS.
- Support for network service scaling.
- support simplifying VNF package generation with user friendly interface.
- Support for NS instances with VNFs running in multiple different datacenters.
- catalog and repositories search functions.

Several main leaders in the networking industry use OSM such as Atos, Cablelabs and Verizon. Figure 6 shows a simplified view of OSM architecture [Int16].

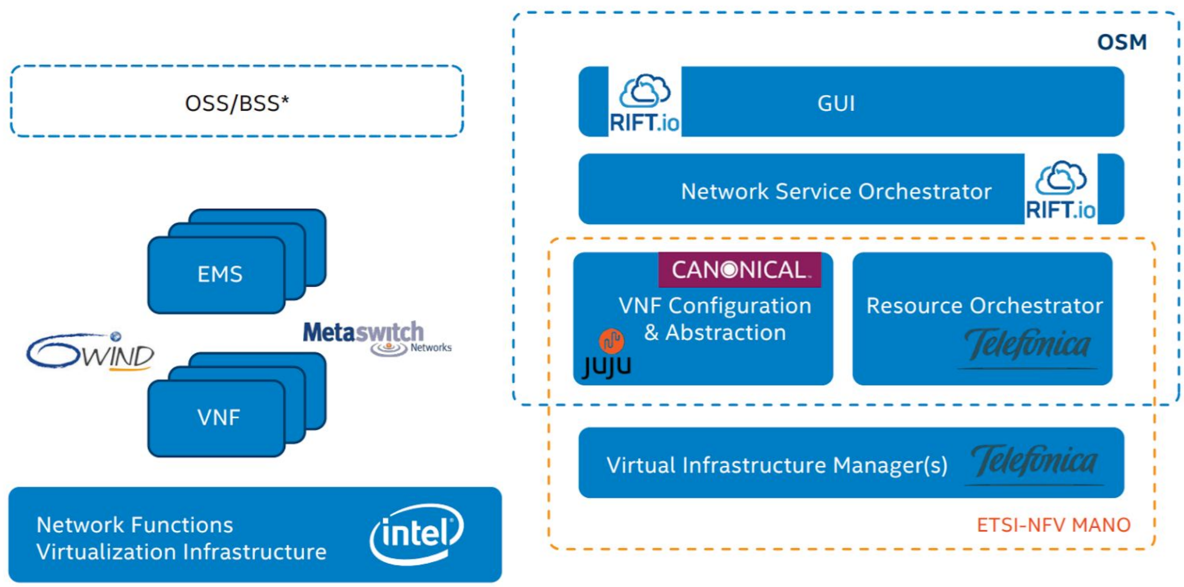

Figure 6 OSM architecture simplified view [Int16]

2.7 Known uses

- Ericsson uses Anuta Networks Solution in order to provide Management and orchestration for their NFV architecture (Amu16).
- The OPEN-Orchestrator Project has an open source orchestration that provide orchestration and management for software-defined networking (SDN) and network function virtualization (NFV) operations provided by the Linux foundation [Far17].
- Hewlett Packard Enterprise (HPE) uses a MANO provided by Ciena called Blue Planet NFV orchestrator [BLU18].
- Huawei created their own MANO which is called CloudOpera MANO [Hua14]
- Open source MANO (OSM) is an open source project provided by ETSI, which implements the ETSI MANO framework [Lop15].

2.8 Consequences

The MANO pattern presents the following advantages:

Flexibility: we can run different types of network services.

Isolation: The VIM ensures that the underlying infrastructure is protected.

Security: VNF manager should ensure that network functions and network graphs including virtual links, virtual networks, sub-nets etc are only accessed by authorized users.

Scalability: MANO can handle the increase of NFV services.

Modularity: Orchestration part of the MANO provides NS execution with a standard interface that is easy to adopt to other subsystems.

Extensibility: It is possible to dynamically provide additional services by adding more resources to the repositories.

Management: using MANO we can handle and manage the life cycle of any number of networks with different requirements in the same time

Performance: We can provide the consumer with a satisfactory level of performance.

2.9 Example Resolved

When Dave wants to expand his network service he will log into the portal of his NFV provider and request network service modification. The request will be sent directly to the MANO in order to be handled. The MANO will check in the repositories for available resources and then forward the request to the VNF manager to perform the necessary changes to the service. In Dave's case the VNF manager will add twenty more ports, four new load balancers, and make sure that they are connected to the network and expand the coverage of the firewall to include the new ports. finally the orchestrator will notify Dave with the result of the request.

2.10 Related patterns

- A pattern for Network Functions Virtualization [Fer15] presented the NFV architecture that shows how to create network services using cloud Software-as-a-Service (SaaS).
- Virtual Machine Operating System architecture [Fer13] describe the structure of the hypervisor and its VMs.
- Cloud ecosystem [Fer16]; shows how the NFV pattern interacts with the different parts of the ecosystem.
- A pattern for an NFV Virtual Machine Environment [Aln18] shows this environment and how it is related to NFV, as well as describing how the NFV architecture interacts with the virtual environment.
- A pattern for SaaS is given in [Has12].

Acknowledgments

We thank our shepherd, Y C Cheng, for his valuable comments that contributed to improve our paper.

References

[Aln18] A. K. Alnaim, A. M. Alwakeel., and E. B. Fernandez. "A Pattern for an NFV Virtual Machine Environment" (accepted in the 13th annual IEEE international systems conference 2019).

[Anu16] Anuta Networks, Network Service Orchestration for Multi-Vendor NFVI. 2016 https://anutanetworks.com/wp-content/uploads/2016/06/Case-Study-Network-Service-Orchestration-for-Multi-Vendor-NFVI.pdf

[Blu18] Blue Planet. Blue Planet streamlines the definition and creation of NFV-based services. Retrieved from https://www.blueplanet.com/products/nfv-orchestration.html

[Ets13] ETSI, "GS NFV 002 v. 1.1.1 - Network functions virtualization (NFV); Use Cases," October 2013. https://www.etsi.org/deliver/etsi_gs/nfv/001_099/001/01.01.01_60/gs_nfv001v010101p.pdf

[Ets13-2] ETSI, "Network functions virtualization (NFV), Network Operator Perspectives on Industry Progress," October 2013. https://portal.etsi.org/Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf

[Ets14] ETSI, "GS NFV 002 - V1.2.1 - Network Functions Virtualisation (NFV); Architectural Framework," 2014, https://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.02.01_60/gs_NFV002v010201p.pdf
[FAR17] Slim, Farah, et al. "Towards a dynamic adaptive placement of virtual network functions under onap." Network Function Virtualization and Software Defined Networks (NFV-SDN), 2017 IEEE Conference on. IEEE, 2017.

[Fer13] E. B. Fernandez, "Security patterns in practice: Building secure architectures using software patterns", Wiley Series on Software Design Patterns, 2013.

[Fer15] E. B. Fernandez and B. Hamid, "A pattern for network functions virtualization," in Proceedings of the 20th European Conference on Pattern Languages of Programs, 2015, p. 47.

[Fer16] E. B. Fernandez, N. Yoshioka, H. Washizaki, and M. H. Syed, "Modeling and security in cloud ecosystems," Future Internet, 2016, 8(2), 13; doi:10.3390/fi8020013.

[Has12] K.Hashizume,E.B.Fernandez,and M.M.Larrondo-Petrie,"A pattern for software-as-a-service in clouds," in Workshop on Redefining and integrating Security Engineering (RISE12), 2012.

[Haw14] H. Hawilo, et al. "NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC)." IEEE Network 28.6 (2014): 18-26.

[Hua14]White Paper - Huawei Observation to NFV - white paper http://www.huawei.com/ilink/en/download/HW_399662

[Int16] Intel End-to-End Service Instantiation Using Open-Source Management and Orchestration Components white paper, Network Functions Virtualization Mobile World Congress 2016

[Lop15] D. R. Lopez, “OpenMANO: The Dataplane Ready Open Source NFV MANO Stack,” Proc. IETF 92 Meeting Proc., Dallas, TX, Mar. 2015.

[Osm18] OSM Release FOUR Technical Overview white paper , May 2018 by ETSI

[Pri12] P. Price and S. Rivera. "Opnfv: An open platform to accelerate nfv." White Paper. A Linux Foundation Collaborative Project (2012).