ETSI GR NFV-EVE 018 V5.1.1 (2024-05)

GROUP REPORT

Network Functions Virtualisation (NFV) Release 5; Evolution and Ecosystem; Report on Multi-tenancy in NFV

Disclaimer

The present document has been produced and approved by the Network Functions Virtualisation (NFV) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference

DGR/NFV-EVE018

Keywords

functional, isolation, management, MANO, NFV,
orchestration, tenancy, virtualisation

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00   Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from:
https://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

If you find a security vulnerability in the present document, please report it through our
Coordinated Vulnerability Disclosure Program:
https://www.etsi.org/standards/coordinated-vulnerability-disclosure

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.
No recommendation as to products and services or vendors is made or should be implied.
No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.
In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Group Report (GR) has been produced by ETSI Industry Specification Group (ISG) Network Functions Virtualisation (NFV).

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document studies multi-tenancy related Use Cases and concepts for NFV, with the goal to remove the gap between the existing general functional requirements on multi-tenancy as described in ETSI GS NFV-IFA 010 [i.10] and the missing requirement details regarding NFV elements consumed by different tenants. Key issues on multi-tenancy in NFV (e.g. tenant-dependent LCM, tenant-dependent resource management, traffic separation, management isolation, etc.) are identified and analysed as well as recommendations for further work are provided.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference/.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[i.1] ETSI GR NFV 001: "Network Functions Virtualisation (NFV); Use Cases".

[i.2] ETSI GS NFV 002: "Network Functions Virtualisation (NFV); Architectural Framework".

[i.3] ETSI GS NFV 003: "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

[i.4] ETSI GS NFV 004: "Network Functions Virtualisation (NFV); Virtualisation Requirements".

[i.5] ETSI GS NFV-IFA 005: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Or-Vi reference point - Interface and Information Model Specification".

[i.6] ETSI GS NFV-IFA 006: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Vi-Vnfm reference point - Interface and Information Model Specification".

[i.7] ETSI GS NFV-IFA 007: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Or-Vnfm reference point - Interface and Information Model Specification".

[i.8] ETSI GS NFV-IFA 008: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Ve-Vnfm reference point - Interface and Information Model Specification".

[i.9] ETSI GS NFV-SOL 001: "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; NFV descriptors based on TOSCA specification".

[i.10] ETSI GS NFV-IFA 010: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Functional requirements specification".

[i.11] ETSI GS NFV-IFA 011: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; VNF Descriptor and Packaging Specification".

[i.12] ETSI GR NFV-EVE 012: "Network Functions Virtualisation (NFV) Release 3; Evolution and Ecosystem; Report on Network Slicing Support with ETSI NFV Architecture Framework".

[i.13] ETSI GS NFV-IFA 013: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Os-Ma-nfvo reference point - Interface and Information Model Specification".

[i.14] ETSI GS NFV-IFA 014: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Network Service Templates Specification".

[i.15] ETSI GR NFV-IFA 028: "Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Report on architecture options to support multiple administrative domains".

[i.16] ETSI GS NFV-IFA 030: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Multiple Administrative Domain Aspect Interfaces Specification".

[i.17] ETSI GR NFV-IFA 034: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Report on Architectural enhancement for VNF License Management support and use of VNF licenses".

[i.18] ETSI GS NFV-IFA 036: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Requirements for service interfaces and object model for container cluster management and orchestration specification".

[i.19] ETSI GS NFV-IFA 040: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Requirements for service interfaces and object model for OS container management and orchestration specification".

[i.20] ETSI GS NFV-SOL 014: "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; YAML data model specification for descriptor-based virtualised resource management".

[i.21] ETSI GS NFV-SEC 023 (V0.0.7): "Network Functions Virtualisation (NFV) Release 5; Security; Container Security Specification".

[i.22] ETSI GR ZSM 010: "Zero Touch Network and Service Management (ZSM); General Security Aspects".

[i.23] 3GPP TR 28.804: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management; Study on tenancy concept in 5G networks and network slicing management (Release 16)".

[i.24] OpenStack® documentation: "OpenStack® Operations Guide".

[i.25] ETSI GS NFV-SEC 025 (V0.0.15): "Network Functions Virtualisation (NFV); Security; Secure End-to-End VNF and NS management specification".

[i.26] ETSI GS NFV-SEC 026 (V0.0.10): "Network Functions Virtualisation (NFV); Security; Isolation and trust domain specification".

[i.27] ENISA NFV 5G security report: "NFV Security in 5G - Challenges and Best Practices".

[i.28] Kata Containers project page.

[i.29] ETSI GS NFV-SOL 003: "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; RESTful protocols specification for the Or-Vnfm Reference Point".

[i.30] ETSI GS NFV-IFA 048: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Policy Information Model Specification".

[i.31] ETSI GS NFV-SOL 013: "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; Specification of common aspects for RESTful NFV MANO APIs".

[i.32] ETSI GS NFV-SOL 005: "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; RESTful protocols specification for the Os-Ma-nfvo Reference Point".

[i.33] ETSI GS NFV-SOL 002: "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; RESTful protocols specification for the Ve-Vnfm Reference Point".

[i.34] ETSI GS NFV-SEC 022: "Network Functions Virtualisation (NFV) Release 4; Security; Access Token Specification for API Access".

[i.35] ETSI GS NFV-SOL 004: "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; VNF Package and PNFD Archive specification".

[i.36] ETSI GS NFV-SEC 021: "Network Functions Virtualisation (NFV) Release 4; Security; VNF Package Security Specification".

[i.37] Anuket project web site.

[i.38] Anuket: "Reference Model for Cloud Infrastructure (RM)".

[i.39] Anuket: "Reference Architecture for OpenStack based cloud infrastructure (RA1)".

[i.40] Anuket: "Reference Architecture for Kubernetes based cloud infrastructure (RA2)".

[i.41] Kubernetes® reference documentation: "Multi-tenancy".

[i.42] Official Kubernetes® reference documentation.

[i.43] Kubernetes® Cluster API.

[i.44] vCluster Documentation (provided by Loft Labs, Inc.).

[i.45] The Kubernetes® API documentation.

[i.46] ETSI GR NFV-IFA 037: "Network Functions Virtualisation (NFV) Release 4; Architectural Framework; Report on further NFV support for 5G".

[i.47] ETSI GS NFV-SOL 016: "Network Functions Virtualisation (NFV) Release 3; Protocols and Data Models; NFV-MANO procedures specification".

[i.48] ETSI GR NFV-EVE 022 (V5.1.1): "Network Functions Virtualisation (NFV) Release 5; Architectural Framework; Report on VNF configuration".

[i.49] ETSI GS NFV-IFA 049 (V5.1.1): "Network Functions Virtualisation (NFV) Release 5; Architectural Framework; VNF generic OAM functions specification".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GS NFV 003 [i.3] apply.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GS NFV 003 [i.3] and the following apply.

NOTE: An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in ETSI GS NFV 003 [i.3].

MANO-P	Provider MANO
MANO-T	Tenant MANO
MLA	Management Level Agreement
NFVO-P	Provider NFVO
NFVO-T	Tenant NFVO
NMT	NFV-MANO Tenant
VIM-P	Provider VIM

VIM-T	Tenant VIM
VNFM-P	Provider VNFM
VNFM-T	Tenant VNFM

4 Overview

According to ETSI GR ZSM 010 [i.22], "*multi-tenancy refers to an architecture in which a single instance of software runs on a server and serves multiple tenants*". In NFV deployments this refers not only to VNF instances but also to NFV-MANO components (NFVO, VNFM, VIM as specified in ETSI GS NFV 002 [i.2]) and to entities like CISM (see ETSI GS NFV-IFA 040 [i.19]) and CCM (see ETSI GS NFV-IFA 036 [i.18]).

The present document focuses on Use Cases where multiple users consume services from the same NFV-MANO environment that are offered via the Os-Ma-nfvo reference point (see ETSI GS NFV-IFA 013 [i.13]). It is analysed e.g. how the NFV-MANO can be enabled to protect a consumer (NFV-MANO Tenant, NMT) against another consumer.

The Use Cases in clause 5 show example configurations where such isolation or protection is provided.

The ways of protection and isolation can be grouped in two groups:

- The first group is related to management actions. This type of protection can use some access control and ownership of entities. This is sometimes called management isolation:
    - Resources created by NMT-A cannot be managed by NMT-B (see note 1).
    - Resources created by NMT-A cannot be used by NMT-B (e.g. bandwidth on a VL of NMT-A cannot be consumed by NMT-B).
    - NMT-B is not able to access information about resources created by NMT-A.
    - NMT-B is not able to monitor usage of resources created by NMT-A.
    - NMT-B is not able to access traffic on resources created by NMT-A.
- The second group is related to normal operation of the workloads. This is sometimes called resource isolation and can be achieved in a strict way or by more light-weight but less powerful means:
    - Traffic on resources serving NMT-A is not able to access resources serving NMT-B.
    - Failures of resources serving NMT-A are not affecting resources serving NMT-B.
    - Load situations of resources serving NMT-A are not affecting resources serving NMT-B.
    - Vulnerabilities of resources serving NMT-A are not affecting resources serving NMT-B.

NOTE 1: In case of shared resources NMT-B could be provided with limited management permissions.

Nevertheless, there can be cases where NMTs A and B share resources.

Different types of resources can be protected in different ways. Therefore in some cases, compute resources, storage resource or network resources are discussed separately.

In many cases resources to isolate are not directly visible on the Os-Ma-nfvo reference point. Therefore the information about isolation expectations is communicated between the entities of the NFV-MANO.

The present document analyses the interworking within NFV-MANO to understand potential enhancements of NFV-MANO that can be used to allow protection and isolation of physical or virtual resources of the NFVI.

This is done by analysing several key issues which are derived from the Use Cases. For each key issue, solutions are provided and evaluated.

Clause 6 describes several key issues and related solutions. From these solutions, recommendations for the further work are derived.

NOTE 2: The present document does not consider multi-tenancy aspects related to security management, certificate management, analytics, automation and intent management as defined in ETSI GS NFV-IFA 010 [i.10].

The present document provides input to the security analysis in ETSI GS NFV-SEC 025 [i.25] the Secure End-to-End VNF and NS management specification and ETSI GS NFV-SEC 026 [i.26] the Isolation and trust domain specification.

5 Use Cases and Scenarios

5.1 Introduction

The first five Use Cases (Use Cases #1 to #5) illustrate in a set of scenarios how an NFV environment can be shared between multiple tenants. Then, in Use Case #6, it is shown how a service provider can offer NFV to tenants. All Use Cases can only be realized if there is established isolation. The next two Use Cases (Use Cases #7 and #8) then demonstrate how different levels of isolation can be achieved, and the last Use Case specifically discusses how NFV entities can be shared between tenants. Finally, Use Case #9 showcases how multiple NMTs can use the same entity.

ETSI GS NFV-SEC 026 [i.26] analyses the Use Cases from a security point of view. Security isolation deals with the protection of entities (e.g. VNFs) in situations when another entity is attacked. Without proper isolation, vulnerabilities of one entity could lead to vulnerability of other entities. Therefore, it is important to understand the different levels of isolation that can be defined in an NFV system.

ETSI GS NFV-SEC 025 [i.25] analyses vulnerabilities and attacks per LCM operation and therefore provides some more details on the use of isolation for protection.

The security considerations will lead to additional aspects in isolation, e.g. more fine-grained affinity definitions or the use of hardware enclaves. These aspects are not discussed in the present document.

5.2 Use Case #1: Two users with own NFVO on shared NFVI

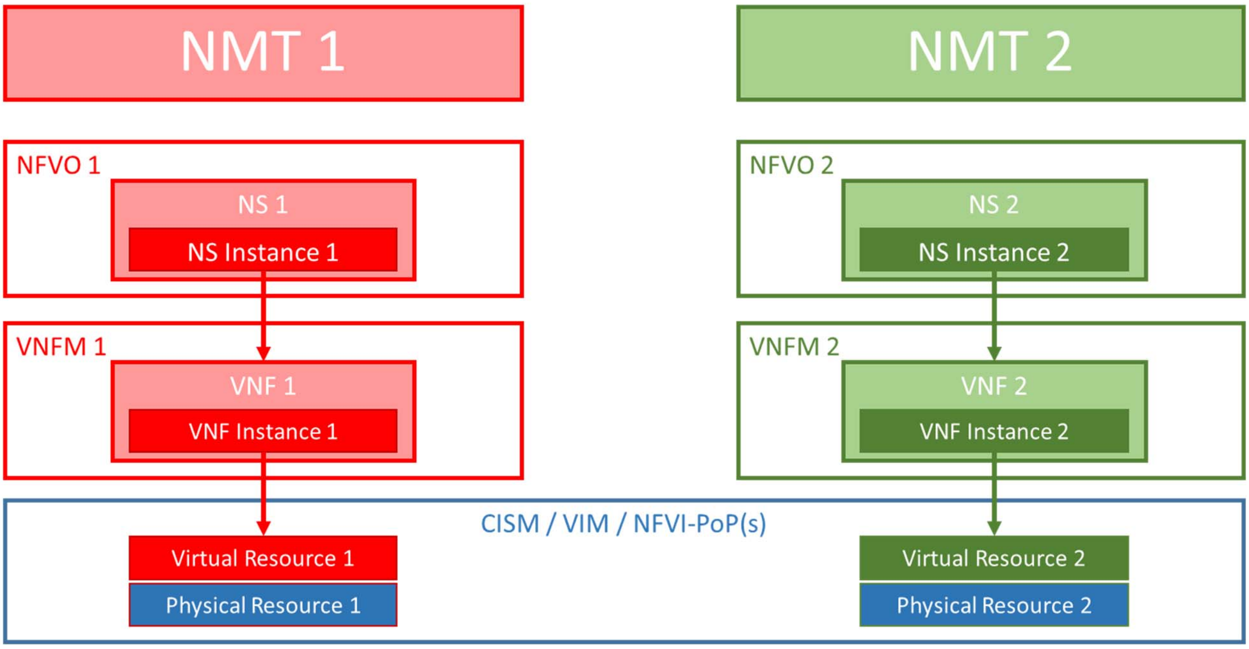
5.2.1 Motivation

This Use Case describes two users using their own NFVO and VNFM but allocate resources from the same NFVI-PoP(s) to build their NSs. It is assumed that an NFVI-PoP is managed by a single VIM

NOTE: The NFV-MANO implementations of the two users can be from different vendors.

In this Use Case the term NFV-MANO tenant (NMT) is used for a user managing NSs via the Os-Ma-nfvo reference point. This reduces ambiguity between NFVO users, VNFM users, VIM users, etc.

Figure 5.2.1-1 illustrates the relation of the NMTs and NFV-MANO FB instances, including the NSs and resources.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.2.1-1: Two users with own NFVO on shared NFVI

In this Use Case it is expected that NFV-MANO FBs are aware that they are expected to provide the appropriate isolation between the resources allocated to the different users (NMTs) as required in ETSI GS NFV 004 [i.4].

In this Use Case it is assumed that the NSs of the different NMTs can be deployed on different resources within the NFVI. For sharing physical resources see other Use Cases.

This Use Case is derived from Use Case #1 in ETSI GR NFV 001 [i.1], Network Function Virtualisation Infrastructure as a Service (NFVIaaS).

5.2.2 Detailed User Story

5.2.2.1 Summary

In this Use Case, each NMT has its own NFVO and VNFM(s) to instantiate their NSs via the Os-Ma-nfvo reference point, see ETSI GS NFV-IFA 013 [i.13]. Therefore NFVO and VNFM(s) do not know about different users/consumers. Only the VIM is aware whether resources are allocated to the same or a different user/consumer.

As specified in ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6], the VIM uses resource groups to identify the expected isolation.

5.2.2.2 Actor(s)

Table 5.2.2.2-1 describes the Use Case actors and roles. It is assumed that NMT1 and NMT2 have no business relationship and their network services are expected to be isolated.

Table 5.2.2.2-1: Use case #1, actors and roles

#	Actor	Description
1	NMT1	OSS or other management system of service provider 1. NMT1 expects isolation from NMT2.
2	NMT2	OSS or other management system of service provider 2. NMT2 expects isolation from NMT1.
3	NFVO1	NFV Orchestrator used by NMT1
4	NFVO2	NFV Orchestrator used by NMT2
5	VNFM1	VNF Manager used by NMT1
6	VNFM2	VNF Manager used by NMT2
7	VIM	VIM managing the NFVI hosting all resources involved
8	VNFs	VNFs of the NS

5.2.2.3 Pre-Conditions

Table 5.2.2.3-1 describes the pre-conditions.

Table 5.2.2.3-1: Use case #1, pre-conditions

#	Pre-condition	Description
1	NFV-MANO of both NMTs (VIM, NFVOs and VNFMs) is running.	
2	NMT1 and NMT2 have established their business relationship with the provider(s) of the NFV-MANO environments and NFVI allowing them to deploy their services.	
3	NMT1 and NMT2 have prepared their NFV packages and templates (e.g. NSD, VNFD) for the onboarding	

5.2.2.4 Description

Table 5.2.2.4-1 describes the flow for onboarding an NS for NMT1. There is no difference to the standard flow. In this Use Case the NFVOs do not consider different tenants during the onboarding, since each NFVO works for a single NMT only.

Table 5.2.2.4-1: Use case #1, base flow for onboarding an NS for NMT1

#	Flow	Description
1	NMT1 -> NFVO1	NMT1 requests NFVO1 to onboard the NS, providing the NSD.
2	NFVO1	NFVO1 executes the onboarding
3	NFVO1 -> NMT1	NFVO acknowledges the onboarding
4	NMT1 -> NFVO1	NMT1 subscribes for notifications. See note.
5	NMT1 -> NFVO1	NMT1 requests NFVO1 to onboard the VNFs, providing the VNF packages and VNFDs.
6	NFVO1	NFVO1 executes the onboarding.
7	NFVO1 -> NMT1	NFVO1 acknowledges the onboarding.
NOTE: Subscription can also be done earlier.		

Table 5.2.2.4-2 describes the flow for instantiating an NS for NMT1. There is no difference to the standard flow. The description only highlights some aspects related to tenants, which is mainly during steps 9 and 10.

Table 5.2.2.4-2: Use case #1, base flow for instantiating an NS for NMT1

#	Flow	Description
1	NMT1 -> NFVO1	NMT1 requests instantiation of the NS.
2	NFVO1	NFVO1 validates the requests. NFVO1 checks that all VNFs are onboarded. NFVO1 checks resource availability for the VNF instantiation. See note 1.
3	NFVO1 -> NMT1	NFVO1 acknowledges the NS instantiation request.
4	NMT1 -> NFVO1	NMT1 subscribes for the relevant notifications. See note 2.
5	NFVO1	NFVO1 validates the requests.
6	NFVO1 -> NMT1	NFVO1 acknowledges the subscriptions.
7	NFVO1 -> VNFM1	NFVO1 requests instantiation of the VNFs for NMT1 as defined for the NS according to the deployment flavour. See note 1 and note 3.
8	VNFM1	VNFM1 validates the request. This includes package validation. See note 1.
9	VNFM1 -> VIM	VNFM1 requests the resources specifying proper resource groups appropriate for NMT1. See note 4.
10	VIM	VIM allocates the resources keeping track of the resource groups.
11	VIM -> VNFM1	VIM provides VNFM1 with the resources for the VNFs for NMT1.
12	VNFM1 -> VNF	VNFM1 finalizes the instantiation which can include configuration and VNF specific operations.
13	VNFM1 -> NFVO1	VNFM1 acknowledges the VNF instantiation.
14	NFVO1 -> NMT1	NFVO1 acknowledges the NS instantiation.
NOTE 1: This step is simplified to avoid two flows for direct or indirect mode. Also allocation of other NS resources and the granting dialogue are not shown. NOTE 2: Subscription can also be done earlier. NOTE 3: This Use Case does not cover nested NSs which will be covered in separate Use Case. NOTE 4: VIM distinguishes NMT1 and NMT2 by the used resource groups, so the NFVI is able to provide resource isolation, e.g. network isolation.		

Onboarding and instantiation for NMT2 are similar.

5.2.2.5 Post-Conditions

Table 5.2.2.5-1 describes the post-conditions.

Table 5.2.2.5-1: Use case #1, post-conditions

#	Post-condition	Description
1	VNFs and NSs are correctly instantiated.	
2	Resources are allocated as per NMT information.	
3	NFV-MANO FBs have all NMT information to provide management isolation.	This includes subscription to notifications.
4	NFVI has all information to isolate resources between NMTs.	This includes the appropriate information for network isolation.

5.2.3 Variants

In ETSI GR NFV-EVE 012 [i.12], isolation of network slices in a multi-domain environment is explained using the diagram in Figure 5.2.3-1. Here also the tenants use their own NFVO and VNFM(s) as illustrated above. In addition, the use of multiple NFVI-PoPs and the SDN environment are shown.

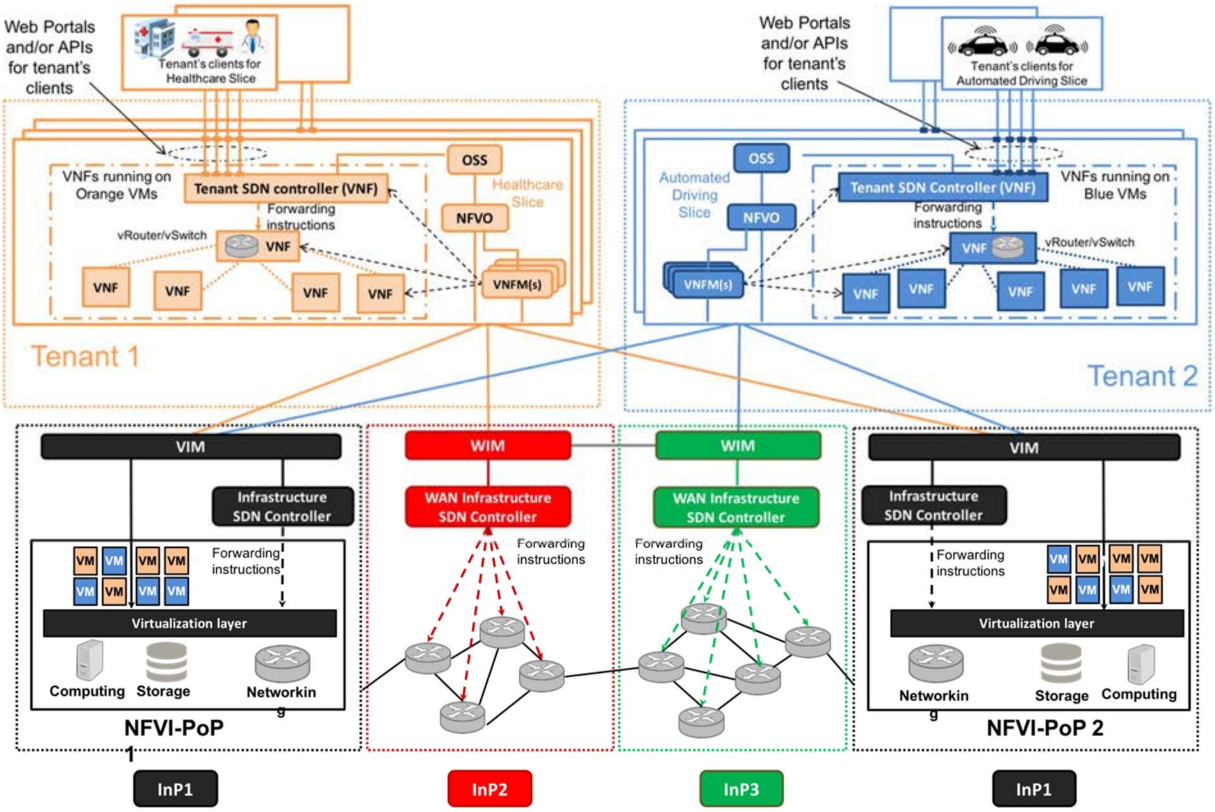
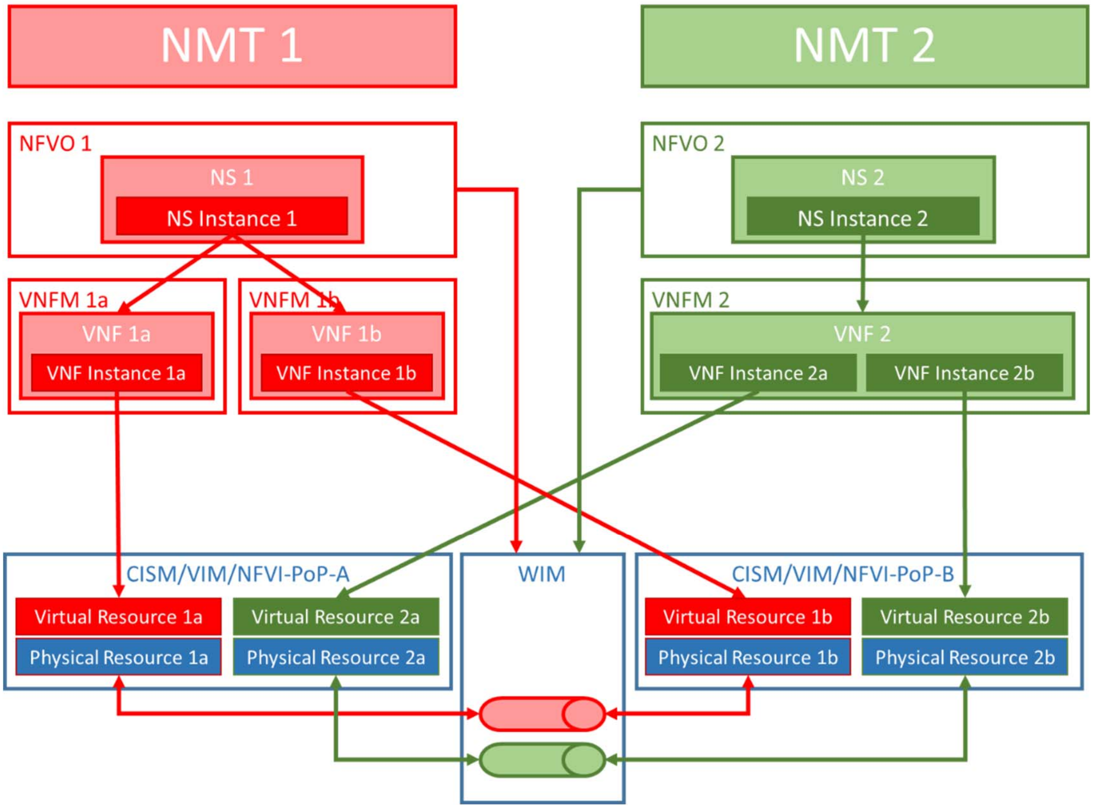


Figure 5.2.3-1: Network slicing deployment applying NFV concepts to achieve isolation

The orange and blue tenants use VMs in two NFVI-PoPs, also marked with orange and blue colour. The constraints for isolation between resources of tenant 1 and tenant 2 include also the connections between the NFVI-PoPs. Therefore also the WIMs are made aware of the isolation constraints. For more details see clause 4.3 of ETSI GR NFV-EVE 012 [i.12].

Figure 5.2.3-2 provides a simplified diagram showing multiple NFVI-PoPs connected by a WIM and using a tunnel for isolation of the traffic.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.2.3-2: Two users with own NFVO in multi-site deployment

5.2.4 Analysis

As shown in the flow in clause 5.2.2.4, the information about expected isolation is provided to the VIM via Vi-Vnfm and Or-Vi reference points. This Use Case can be implemented without providing direct information about tenancy on these reference points; it is sufficient to provide information about groups of resources, within which sharing is possible, whereas the groups itself are to be isolated against each other.

ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6], already introduce resource groups to identify the expected isolation on resource level. However, management of resource groups is not yet specified and not all operations and information elements include the appropriate information (e.g. in the interfaces for compute host reservation, start and end time can be specified by a given tenant, without providing appropriate parameters to bind the reservation to a tenant or resource group. Also the use of resource groups between ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6] is not completely consistent. See solution proposal #1.3 in clause 6.1.4 and recommendations in Table 6.1.6-2.

Also requirements for management isolation of the VIM will be recommended, which will protect resources assigned for a tenant against management from a different tenant. See solution proposal #2.2 in clause 6.2.3 and related recommendations in Table 6.2.7-1.

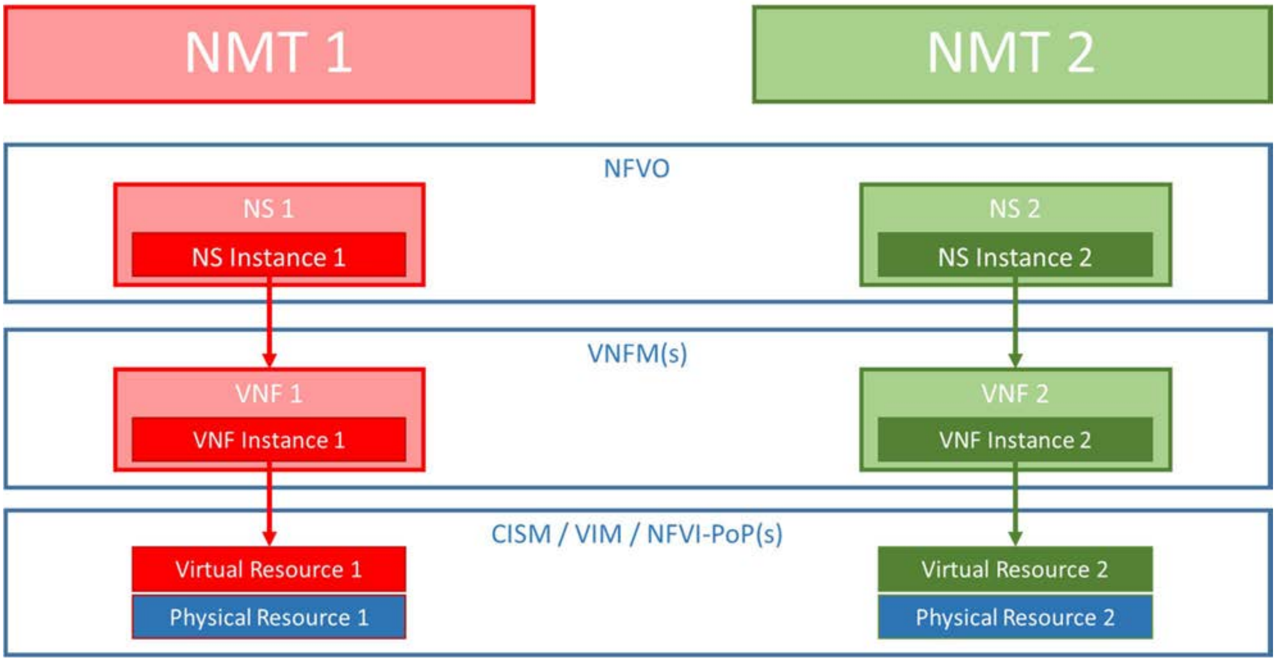
5.3 Use Case #2: Two users share the same NFV environment

5.3.1 Motivation

This Use Case describes two service providers creating NSs on the same NFV environment (i.e. the same NFVO and other FBs); they are thereby being built with resources of the same NFVI-PoP(s). The service providers use the same NFV-MANO service to deploy these NS instances. Figure 5.3.1-1 illustrates the relation of the service providers and NFV-MANO, including the NSs and resources.

In this Use Case the term NFV-MANO Tenant (NMT) is used for a user managing NSs via the Os-Ma-nfvo reference point. This reduces ambiguity between NFVO users, VNFM users, VIM users, etc.

NOTE: This Use Case is applicable, for instance, if the NFV environment including the whole NFV-MANO functionality is deployed on a public cloud infrastructure.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.3.1-1: Two users share the same NFV environment

In this Use Case it is expected that all NFV-MANO FBs are aware that they are expected to provide the appropriate isolation between the resources allocated to the different users (NMTs) as required in ETSI GS NFV 004 [i.4].

In this Use Case it is assumed that the NSs of the different NMTs can be deployed on different resources within the NFVI. For sharing of NFVI resources see other Use Cases.

5.3.2 Detailed User Story

5.3.2.1 Summary

In this Use Case, both NMTs instantiate their NSs with the same NFVO via the Os-Ma-nfvo reference point, see ETSI GS NFV-IFA 013 [i.13].

There are multiple ways for the NFVO to know the origin of a request:

- The NMTs can use separate interfaces (e.g. IP-addresses) to the NFV-MANO.
- The NMTs can use some token to identify themselves.

It is out of scope for the Use Case which method is used. It is assumed that NFVO can identify the NMT of the NS as the originator of a request on the Os-Ma-nfvo reference point.

In this Use Case it is expected that NFV-MANO provides isolation of the NSs created by different NMTs (see note 1). This includes resource and traffic isolation similar to anti-affinity, and also management isolation, so an NMT can only manage NSs and resources under its responsibility.

NOTE 1: NMTs can waive isolation by agreement. In this case NFV-MANO is not expected to distinguish these users.

NOTE 2: NMTs can also share resources, as is illustrated and analysed in Use Case #9, see clause 5.10.

NOTE 3: A NMT here can also represent a group of users.

In this Use Case it is also assumed that the same NMT instantiates the NS and subsequently owns all VNF instances and related resources allocated during the VNF instantiation. The NMT can issue subsequent operations on the NSs he created as well as on their constituent resources. E.g. it is expected that an NMT will subscribe for notifications related to the NSs and its constituent resources.

NOTE 4: This Use Case is formulated with the assumption that the NFV-MANO can distinguish the users. Isolation could also be achieved by some anti-affinity definition on higher level (NSs). This solution is shown in key issue #1, solution #1.4, see clause 6.1.5.

The VIM is made aware whether resources are allocated to the same or a different user/consumer.

As specified in ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6], the VIM uses resource groups to identify the expected isolation.

As specified in some information elements in ETSI GS NFV-IFA 007 [i.7], the resource groups are also used on the Or-Vnfm reference point for tenant information.

5.3.2.2 Actor(s)

Table 5.3.2.2-1 describes the Use Case actors and roles. It is assumed that NMT1 and NMT2 have no business relationship and their network services are expected to be isolated.

Table 5.3.2.2-1: Use case #2, actors and roles

#	Actor	Description
1	NMT1	OSS or other management system of service provider 1. NMT1 expects isolation from NMT2.
2	NMT2	OSS or other management system of service provider 2. NMT2 expects isolation from NMT1.
3	NFVO	NFV Orchestrator for the NS instances involved.
4	VNFM	VNF Manager for the VNFs involved.
5	VIM	VIM managing the NFVI hosting all resources involved.
6	VNFs	VNFs of the NS.

5.3.2.3 Pre-Conditions

Table 5.3.2.3-1 describes the pre-conditions.

Table 5.3.2.3-1: Use case #2, pre-conditions

#	Pre-condition	Description
1	NFV-MANO (VIM, NFVO and VNFM) is running.	
2	NMT1 and NMT2 have established their business relationship with the provider(s) of the NFV environment allowing them to deploy their services.	This includes that NFV-MANO is aware of the expected isolation of NSs and their constituents.
3	NMT1 and NMT2 have prepared the NFV packages and templates (e.g. NSD, VNFD) for the onboarding.	

5.3.2.4 Description

Table 5.3.2.4-1 describes the flow for onboarding an NS for NMT1. There is no difference to the standard flow. The description only highlights some aspects related to tenants, which is mainly in steps 2 and 6.

Table 5.3.2.4-1: Use case #2, base flow for onboarding an NS for NMT1

#	Flow	Description
1	NMT1 -> NFVO	NMT1 requests NFVO to onboard the NS, providing the NSD.
2	NFVO	NFVO executes the onboarding and registers NMT1 for this NS, see note 1 and note 2.
3	NFVO -> NMT1	NFVO acknowledges the onboarding.
4	NMT1 -> NFVO	NMT1 subscribes for notifications. See note 4.
5	NMT1 -> NFVO	NMT1 requests NFVO to onboard the VNFs, providing the VNF packages VNFDs.
6	NFVO	NFVO executes the onboarding and registers NMT1 for this VNF package. See note 1 and note 3.
7	NFVO -> NMT1	NFVO acknowledges the onboarding.
NOTE 1: NFVO registers the NMT(s) to be able to protect a package, an NS or VNF, against operations from a different user (i.e. management isolation). It is recommended that different levels of permission (e.g. use versus making changes such as scale, update, delete) can be specified. NOTE 2: NSs could be shared between NMTs, but this is not covered in this Use Case. See Use Case #9 in clause 5.10. For the case of multiple NMTs for an NS, it is recommended that different levels of permission (e.g. use, scale, update, delete) can be specified. NOTE 3: VNFs could be shared between NSs of different NMTs, but this is not covered in this Use Case. See separate Use Case in clause 5.10. For the case of multiple NMTs for a VNF, it is recommended that different levels of permission (e.g. use, scale, update, delete) can be specified. NOTE 4: Subscription can also be done earlier.		

Table 5.3.2.4-2 describes the flow for instantiating an NS for NMT1. There is no difference to the standard flow. The description only highlights some aspects related to tenants, which is mainly during steps 2, 7, 9 and 10.

Table 5.3.2.4-2: Use case #2, base flow for instantiating an NS for NMT1

#	Flow	Description
1	NMT1 -> NFVO	NMT1 requests instantiation of the NS.
2	NFVO	NFVO validates the requests. This includes that NFVO checks whether NMT1 is allowed to instantiate the NS. NFVO checks that all VNFs are onboarded and are allowed to be instantiated by NMT1. NFVO checks resource availability for the VNF instantiation. See note 1 and note 2.
3	NFVO -> NMT1	NFVO acknowledges the NS instantiation request.
4	NMT1 -> NFVO	NMT1 subscribes for the relevant notifications. See note 3.
5	NFVO	NFVO validates the requests. This includes that NFVO checks whether NMT1 is allowed to subscribe to these notifications.
6	NFVO -> NMT1	NFVO acknowledges the subscriptions.
7	NFVO -> VNFM	NFVO requests instantiation of the VNFs as specified for the NS according to the deployment flavour indicating resource groups appropriate for NMT1. See note 1 and note 4.
8	VNFM	VNFM validates the request. This includes package validation.
9	VNFM -> VIM	VNFM requests the resources for the VNFs indicating resource groups appropriate for NMT1. See note 5.
10	VIM	VIM allocates the resources keeping track of the resource groups.
11	VIM -> VNFM	VIM provides VNFM with the resources for the VNFs for NMT1.
12	VNFM -> VNF	VNFM finalizes the instantiation which can include configuration and VNF specific operations.
13	VNFM -> NFVO	VNFM acknowledges the VNF instantiation.
14	NFVO -> NMT1	NFVO acknowledges the NS instantiation.
NOTE 1: This step is simplified to avoid two flows for direct or indirect mode. Also allocation of other NS resources and the granting dialogue are not shown. NOTE 2: Resource availability includes availability within resource limits for NMT1. NOTE 3: Subscription can also be done earlier. NOTE 4: This Use Case does not cover nested NSs which will be covered in separate Use Case. NOTE 5: VIM distinguishes NMT1 and NMT2, so the NFVI is able to provide resource isolation, e.g. network isolation.		

Onboarding and instantiation for NMT2 are similar.

5.3.2.5 Post-Conditions

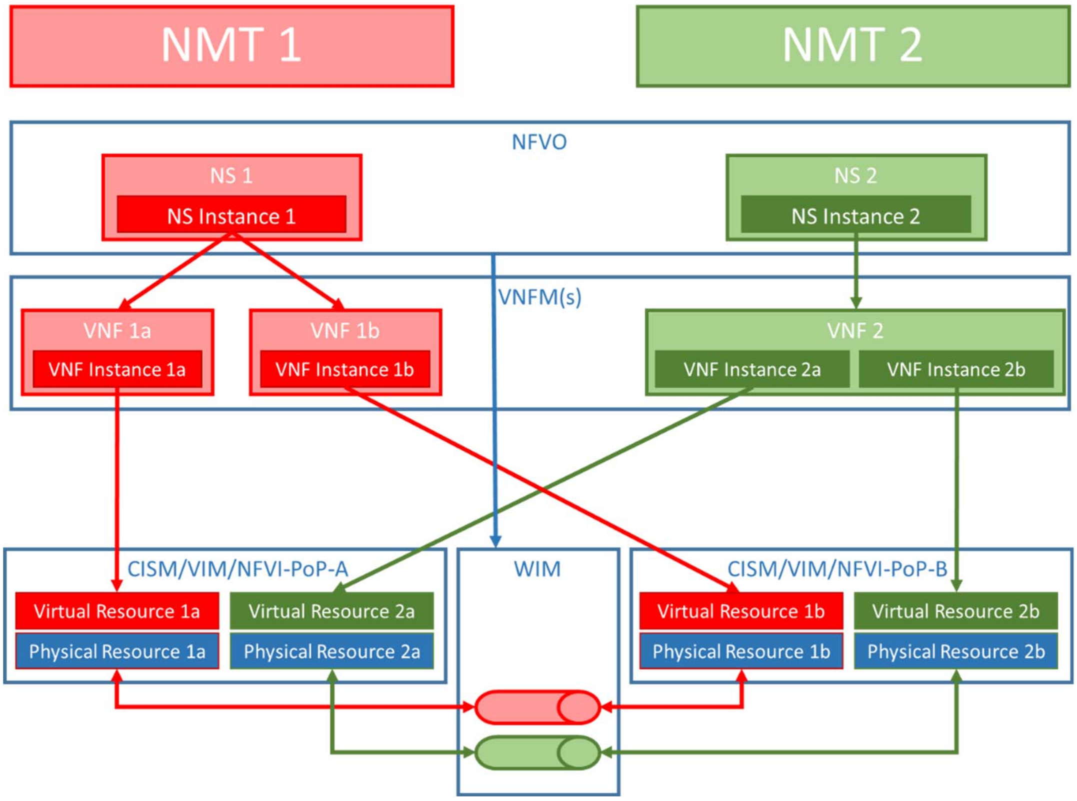
Table 5.3.2.5-1 describes the post-conditions.

Table 5.3.2.5-1: Use case #2, post-conditions

#	Post-condition	Description
1	VNFs and NSs are correctly instantiated.	
2	Resources are allocated as per NMT information.	
3	NFV-MANO FBs have all NMT information to provide management isolation.	This includes subscription to notifications.
4	NFVI has all information to isolate resources between NMTs.	This includes the appropriate information for network isolation.

5.3.3 Variants

Figure 5.3.3-1 shows an NFV environment with multiple NFVI-PoPs connected by a WIM and using a tunnel for isolation of the traffic used by multiple tenants.



NOTE: The figure above showing "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.3.3-1: Two users share the same NFV multi-site environment

5.3.4 Analysis

As shown in the flow in clause 5.3.2.4 and similar to Use Case #1, the information about expected isolation is provided to the VIM via Vi-Vnfm and Or-Vi reference points. In addition to Use Case #1, the NFVO also provides VNFM with the information about expected isolation via the Or-Vnfm reference point.

NOTE: There is a difference between direct and indirect mode of resource allocation by the VNFM. In indirect mode, i.e. when the VNFM allocates resources via the NFVO, a different mechanism could be used.

This Use Case can be implemented without providing direct information about tenancy on these reference points; it is sufficient to provide information about groups of resources, within which sharing is possible, whereas the groups itself are to be isolated against each other.

ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6], already introduce resource groups to identify the expected isolation on resource level. However, management of resource groups is not yet specified and not all operations and information elements include the appropriate information (e.g. in the interfaces for compute host reservation, start and end time can be specified by a given tenant, without providing appropriate parameters to bind the reservation to a tenant or resource group. Also the use of resource groups between ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6] is not completely consistent.

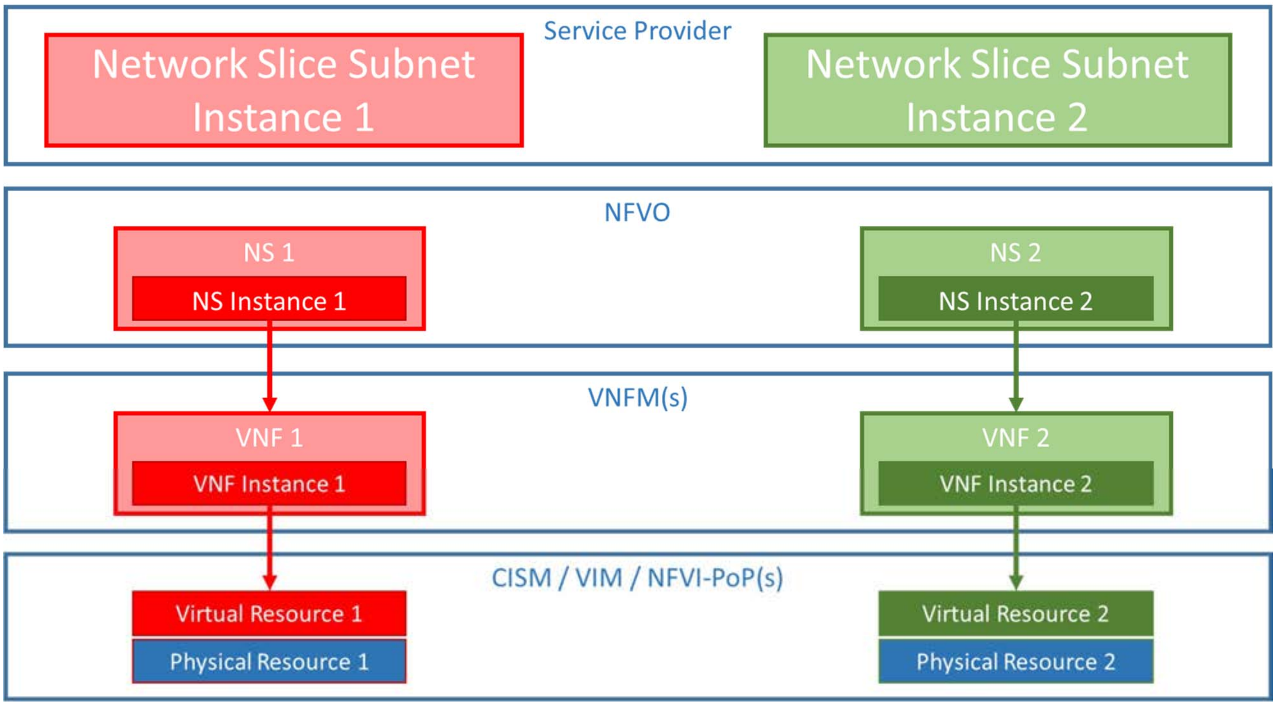
ETSI GS NFV-IFA 007 [i.7] already mentions resource groups. Attributes for resource groups identifying the tenants are specified in VirtualisedResourceQuotaAvailableNotification and GrantInfo information element, but not in other places. The flow in clause 5.3.2.4 shows that the information about resource groups can be provided also in other information elements or as parameters for LCM operations. See solution proposal #1.3 in clause 6.1.4 and recommendations in Table 6.1.6-2.

Also requirements for management isolation of the VIM will be recommended, which will protect resources assigned for a tenant against management from a different tenant. Requirements for management isolation will be recommended also for NFVO and VNFM, so an NMT can only access own NSs and VNFs. This includes the protection of VNF packages against usage of non-authorized tenants. See solution proposal #2.2 in clause 6.2.3 and related recommendations in Table 6.2.7-1.

5.4 Use Case #3: Network slicing by a single user

5.4.1 Motivation

This Use Case shows the use of network slicing. A service provider uses network slicing and creates two slice subnets by creating network service instances on the same NFV environment (i.e. the same NFVO and other FBs) and thus being built with resources of the same NFVI-PoP(s). The service provider expects isolation of the slice subnets. Thus NFVO is expected to provide isolation of the NS instances and their resources. Figure 5.4.1-1 illustrates the relation of the slice subnets, NS instances and their resources.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.4.1-1: Network slice subnets

In this Use Case it is expected that all NFV-MANO FBs are aware that the NS instances and their resources are expected to be isolated.

In difference to Use Case #2, the same service provider manages both NS instances. Therefore, in this Use Case NSs do not need protection against management operations from another tenant. The isolation of resources is sufficient. The current solution for NFV to support network slicing relies on the standard isolation that is provided between NS instances and assumes that no additional isolation constraints need to be defined, e.g. via the Os-Ma-nfvo reference point.

In this Use Case it is assumed that the NS instances of the different slice subnets can be deployed on different resources within the NFVI. For sharing of NFVI resources see other Use Cases.

NOTE: The network slice is not shown, since the containment of subnets and slices is invisible for NFV-MANO. The shown subnets can belong to the same slice or different slices.

5.4.2 Detailed User Story

5.4.2.1 Summary

In this Use Case, a single service provider instantiates two network slice subnets that are to be isolated. The Network Services that are used for the network slice subnets are instantiated via the same NFVO via the Os-Ma-nfvo reference point, see ETSI GS NFV-IFA 013 [i.13].

During the instantiation, the service provider provides the NFVO with some information whether NS instances are part of the same or different network slice subnets.

While in the previous Use Case, different NMTs can use separate interfaces, in this Use Case it is the same service provider instantiating multiple slice subnets. For this it specifies some information about expected isolation on the Os-Ma-nfvo reference point.

In this Use Case it is expected that NFV-MANO provides isolation of the NSs created for the different network slice subnets (see note). This includes resource and traffic isolation. In difference to the previous Use Case, management isolation is not expected within NFV-MANO.

NOTE: Service providers that do not expect isolation between some network services can indicate that to NFV-MANO in the same way as if NS instances would be part of the same network slice subnet.

The VIM is made aware of the isolation constraints indicating whether resources are allocated to the same or a different network slice subnet.

As specified in ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6], the VIM uses resource groups to identify the expected isolation.

5.4.2.2 Actor(s)

Table 5.4.2.2-1 describes the Use Case actors and roles.

Table 5.4.2.2-1: Use case #3, actors and roles

#	Actor	Description
1	OSS	OSS or other management system of the service provider. The description of this Use Case does not distinguish different components in the OSS/BSS layer. Therefore no network slice subnet management functionality is mentioned.
3	NFVO	NFV Orchestrator for the NS instances involved.
4	VNFM	VNF Manager for the VNFs involved.
5	VIM	VIM managing the NFVI hosting all resources involved.
6	VNFs	VNFs of the NS.

5.4.2.3 Pre-Conditions

Table 5.4.2.3-1 describes the pre-conditions.

Table 5.4.2.3-1: Use case #3, pre-conditions

#	Pre-condition	Description
1	NFV-MANO (VIM, NFVO and VNFM) is running.	
2	The OSS has prepared the NFV packages and templates (e.g. NSD, VNFD) for the onboarding.	

5.4.2.4 Description

Table 5.4.2.4-1 describes the flow for onboarding an NS to be used for the network slice subnet. There is no difference to the standard flow. The onboarding is not specific for a network slice subnet, but the onboarded NS can be instantiated multiple times for multiple network slice subnets.

Table 5.4.2.4-1: Use case #3, base flow for onboarding an NS

#	Flow	Description
1	OSS -> NFVO	OSS requests NFVO to onboard the NS, providing the NSD.
2	NFVO	NFVO executes the onboarding.
3	NFVO -> OSS	NFVO acknowledges the onboarding.
4	OSS -> NFVO	OSS subscribes for notifications. See note.
5	OSS -> NFVO	OSS requests NFVO to onboard the VNFs, providing the VNF packages and VNFDs.
6	NFVO	NFVO executes the onboarding.
7	NFVO -> OSS	NFVO acknowledges the onboarding.
NOTE: Subscription can also be done earlier.		

Table 5.4.2.4-2 describes the flow for instantiating an NS for the network slice subnet. There is no difference to the standard flow. The description only highlights some aspects related to identify the subnet, which is mainly during steps 2, 7, 9 and 10.

Table 5.4.2.4-2: Use case #3, base flow for instantiating an NS for NMT1

#	Flow	Description
1	OSS -> NFVO	OSS requests instantiation of the NS.
2	NFVO	NFVO validates the requests. NFVO checks that all VNFs are onboarded. NFVO checks resource availability for the VNF instantiation. See note 1.
3	NFVO -> OSS	NFVO acknowledges the NS instantiation request.
4	OSS -> NFVO	OSS subscribes for the relevant notifications. See note 2.
5	NFVO	NFVO validates the requests.
6	NFVO -> OSS	NFVO acknowledges the subscriptions.
7	NFVO -> VNFM	NFVO requests instantiation of the VNFs as defined for the NS according to the deployment flavour indicating proper resource groups to indicate the expected isolation of the network slice subnets. See note 1 and note 3.
8	VNFM	VNFM validates the request. This includes package validation.
9	VNFM -> VIM	VNFM requests the resources for the VNFs indicating resource groups to indicate the expected isolation of the network slice subnet. See note 4.
10	VIM	VIM allocates the resources keeping track of the resource groups.
11	VIM -> VNFM	VIM provides VNFM with the resources for the VNFs.
12	VNFM -> VNF	VNFM finalizes the instantiation which can include configuration and VNF specific operations.
13	VNFM -> NFVO	VNFM acknowledges the VNF instantiation.
14	NFVO -> OSS	NFVO acknowledges the NS instantiation.
NOTE 1: This step is simplified to avoid two flows for direct or indirect mode. Also allocation of other NS resources and the granting dialogue are not shown. NOTE 2: Subscription can also be done earlier. NOTE 3: This Use Case does not cover nested NSs which will be covered in separate Use Case. NOTE 4: VIM distinguishes the resource groups indicating the isolation constraints of the network slice subnets, so the NFVI is able to provide resource isolation, e.g. network isolation.		

5.4.2.5 Post-Conditions

Table 5.4.2.5-1 describes the post-conditions.

Table 5.4.2.5-1: Use case #3, post-conditions

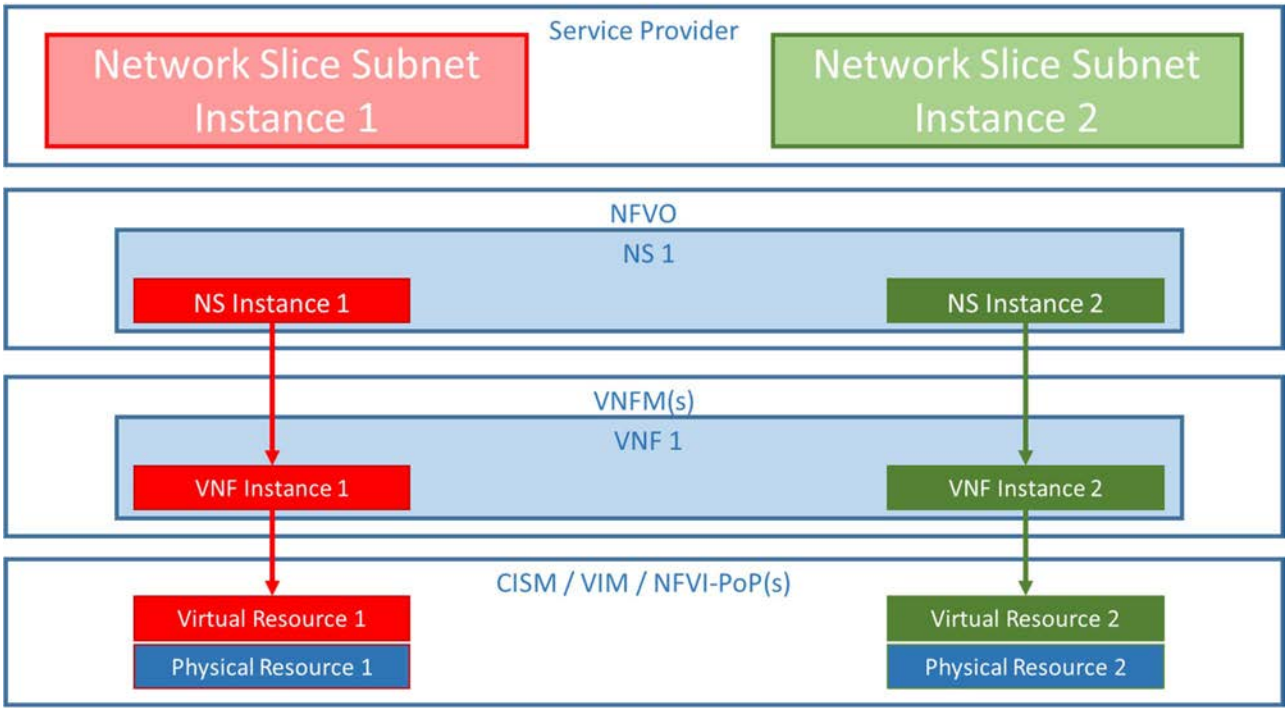
#	Post-condition	Description
1	VNFs and NSs are correctly instantiated.	
2	Resources are allocated as per information related to network slice subnets.	
3	NFVI has all information to isolate resources between network slice subnets.	This includes the appropriate information for network isolation.

5.4.3 Variants

5.4.3.1 Variant: Network Slice Subnets use instances of the same NS

In the Use Case above, it is assumed that subnet 1 and subnet 2 are implemented using different network services. In a variant of this Use Case, two subnets can be implemented by instances of the same NS. In that case, the instances can also use instances of the same VNF.

Figure 5.4.3.1-1 illustrates the relation of the slice subnets, NS instances and their resources in this variant.



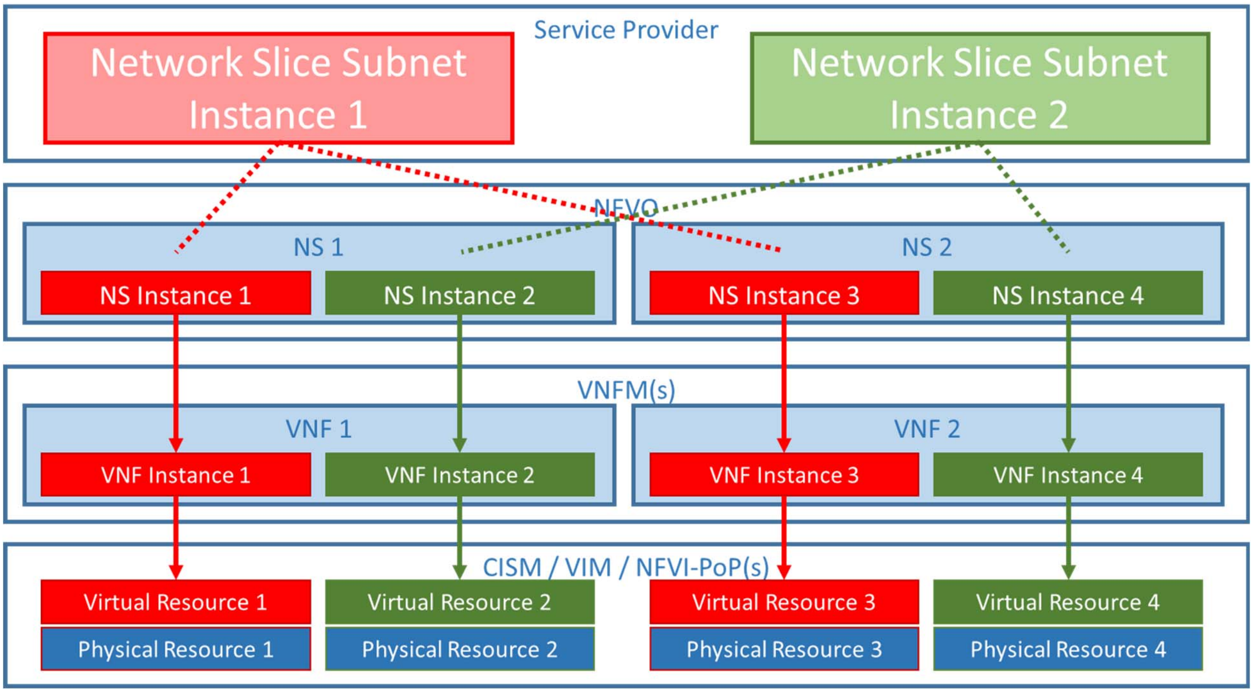
NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.4.3.1-1: Network slice subnets implemented by the same NS

The details of sharing an NS instance are not specific to network slicing. Sharing of NS instances is discussed in Use Case #9, see clause 5.10.3.

5.4.3.2 Variant: Multiple NS instances in the same network slice subnet

A slice subnet can contain multiple network services or network service instances. In Figure 5.4.3.2-1 it is illustrated that both network slice subnet instances contain instances from two network services. Within a network slice subnet, no isolation is expected. Thus the "red" virtual resources 1 and 3 do not expect isolation.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.4.3.2-1: Network slice subnets with multiple NS instances

5.4.4 Analysis

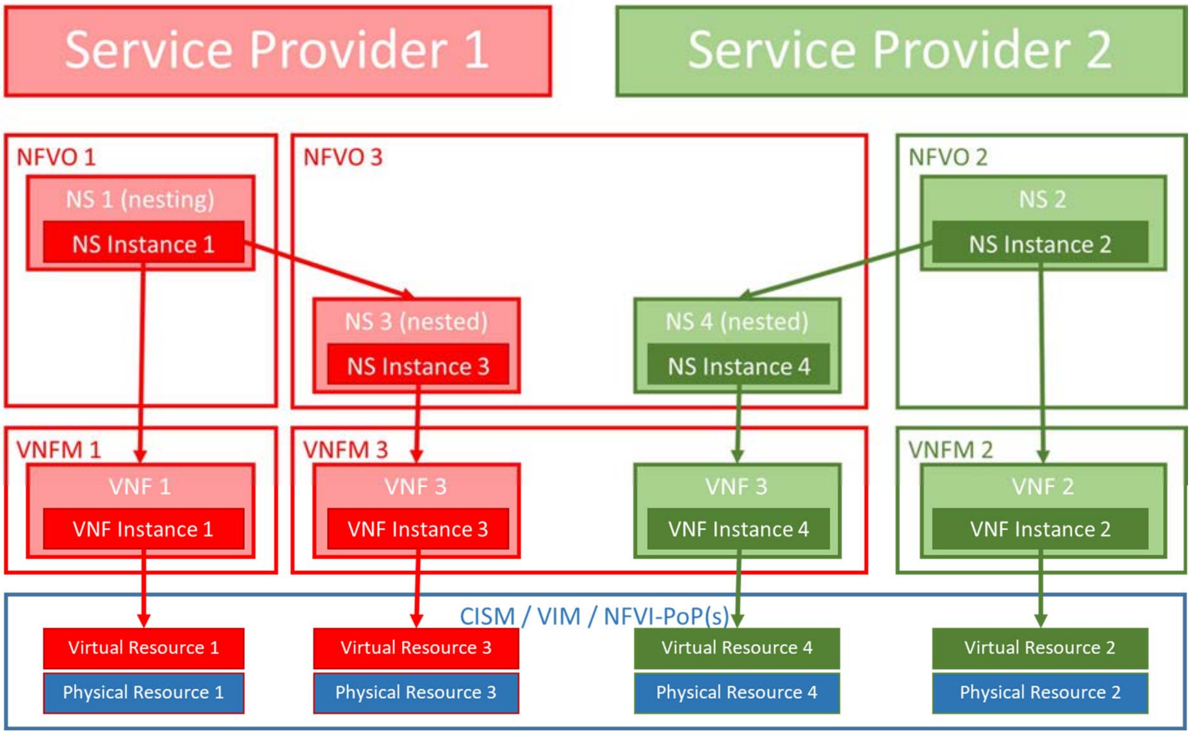
The isolation expectations in this Use Case are very similar as in Use Case #2. Resource groups can be used at the interface to VNFM and VIM in the same way as described in clause 5.3.4. In this Use Case, no management isolation is expected, but the same service provider will specify the isolation constraints on the Os-Ma-nfvo reference point.

In the easiest way, NFVO would isolate every NS instance. In that case, no additional information is shared between OSS and NFVO. In a more complex scenario, see the variant in clause 5.4.3.2, multiple Network Services or instances could share the same resource groups of VNFM and VIM if they are part of the same network slice subnet and do not expect isolation. In that case, additional information about expected isolation is provided on the Os-Ma-nfvo reference point (see ETSI GS NFV-IFA 013 [i.13]).

5.5 Use Case #4: Nested network services

5.5.1 Motivation

This Use Case shows nested network services as introduced with ETSI GR NFV-IFA 028 [i.15] and ETSI GS NFV-IFA 030 [i.16]. In Figure 5.5.1-1 NS 3 is a nested NS of NS 1, similarly, NS4 is a nested NS of NS 2. Both NS 1 and NS 3 belong to Service Provider 1, while NS 2 and NS 4 belong to Service Provider 2. Isolation between NS 1 and NS 3 and their constituents is not expected, similarly for NS 2 and NS 4. They could even share resources. But NS 3 and NS 4 and their resources are expected to be isolated.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.5.1-1: Nested Network Service

In this Use Case, it is expected that NFV-MANO FBs are aware which resources can be shared or are expected to be isolated. In particular this means:

- NS 1 and NS 3 and their constituents can share resources.
- NS 2 and NS 4 and their constituents can share resources.
- Resource isolation is expected between even and odd numbered NSs/VNFs/resources.
- NS 3 and NS 4 expect isolation, both for management access and for resources.

In this Use Case the term NFV-MANO tenant (NMT) is not used. Here two different types of NMTs can be seen. The service providers consume the NFVO interfaces via the Os-Ma-nfvo reference point and thus are NFV-MANO tenants. Also NFVO1 and NFVO2 consume NFVO interfaces of NFVO3 (via the Or-Or reference point specified in ETSI GS NFV-IFA 030 [i.16]) and thus can be seen as NFV-MANO tenants.

One of the goals of this Use Case is to illustrate that the NFVI resources of VNF1 and VNF3 (marked red in the diagram) are consumed by the same service provider and thus no isolation is expected between them. Similarly for the resources marked green and consume by Service Provider 2.

5.5.2 Detailed User Story

5.5.2.1 Summary

In this Use Case, the NSs are instantiated in the same way as in Use Case #2, clause 5.3.2. In addition, nested NSs are instantiated, via Or-Or reference point, see ETSI GS NFV-IFA 030 [i.16] and the flows in annex of ETSI GR NFV-IFA 028 [i.15].

In this Use Case it is again expected that NFV-MANO provides isolation of the NSs created by different Service Providers. This includes resource and traffic isolation similar to anti-affinity, and also management isolation, so a Service Provider can only manage NSs (including nested NSs) and resources under its responsibility.

The VIM is made aware whether resources are allocated to the same or a different user/consumer.

As specified in ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6], the VIM uses resource groups to identify the expected isolation.

As specified in some information elements in ETSI GS NFV-IFA 007 [i.7], the resource groups are also used on the Or-Vnfm reference point for tenant information.

5.5.2.2 Actor(s)

Table 5.5.2.2-1 describes the Use Case actors and roles. It is assumed that service provider 1 and service provider 2 have no business relationship and their network services are expected to be isolated.

Table 5.5.2.2-1: Use case #4, actors and roles

#	Actor	Description
1	SP1	OSS or other management system of service provider 1. SP1 expects isolation from SP2.
2	SP2	OSS or other management system of service provider 2. SP2 expects isolation from SP1.
3	NFVO1	NFV Orchestrator used by SP1.
4	NFVO2	NFV Orchestrator used by SP2.
5	NFVO3	NFV Orchestrator responsible for the nested NSs, which are managed by NFVO1 and NFVO2.
6	VNFM1	VNF Manager used by SP1.
7	VNFM2	VNF Manager used by SP2.
8	VNFM3	VNF Manager used in the domain of NFVO3. See Note.
9	VIM	VIM managing the NFVI hosting all resources involved.
10	VNFs	VNFs of the NS.
NOTE: There could be multiple VNFMs.		

5.5.2.3 Pre-Conditions

Table 5.5.2.3-1 describes the pre-conditions.

Table 5.5.2.3-1: Use case #4, pre-conditions

#	Pre-condition	Description
1	NFV-MANO (VIM, NFVO and VNFM) is running.	
2	Service Providers have established their business relationship with the provider(s) of the NFV environment allowing them to deploy their services.	This includes that NFV-MANO is aware of the expected isolation of NSs and their constituents.
3	Service Providers have prepared the NFV packages and templates (e.g. NSD, VNFD) for the onboarding.	

5.5.2.4 Description

Table 5.5.2.4-1 describes the flow for onboarding an NS for NMT1. There is no difference to the standard flow. The description only highlights some aspects related to tenants, which is mainly during steps 2 and 6, and similarly during step 8.

Table 5.5.2.4-1: Use case #4, base flow for onboarding an NS for SP1

#	Flow	Description
1	SP1 -> NFVO1	SP1 requests NFVO1 to onboard the NS, providing the NSD.
2	NFVO1	NFVO1 executes the onboarding and registers SP1 for this NS, see note 1 and note 2.
3	NFVO1 -> SP1	NFVO1 acknowledges the onboarding.
4	SP1 -> NFVO1	SP1 subscribes for notifications. See note 5.
5	SP1 -> NFVO1	SP1 requests NFVO1 to onboard the VNFs, providing the VNF packages VNFDs.
6	NFVO1	NFVO1 executes the onboarding and registers SP1 for the VNF packages. See note1 and note 3.
7	NFVO1 -> SP1	NFVO1 acknowledges the onboarding.
8	SP1 -> NFVO3	SP1 executes the same steps 1 to 7 for onboarding the nested NSs and the related VNFs. See note 5.
NOTE 1: NFVO registers the Service Provider(s) to be able to protect a package, an NS or VNF, against operations from a different user (i.e. management isolation). It is recommended that different levels of permission (e.g. use versus making changes such as scale, update, delete) can be specified. NOTE 2: NSs could be shared between Service Providers, but this is not covered in this Use Case. See Use Case #9 in clause 5.10. For the case of multiple Service Providers for an NS, it is recommended that different levels of permission (e.g. use, scale, update, delete) can be specified. NOTE 3: VNFs could be shared between NSs of different Service Provides but this is not covered in this Use Case. See separate Use Case in clause 5.10. For the case of multiple Service Providers for a VNF, it is recommended that different levels of permission (e.g. use, scale, update, delete) can be specified. NOTE 4: Subscription can also be done earlier. NOTE 5: There is no onboarding via Or-Or reference point. SP1 interacts directly with NFVO3. See also ETSI GR NFV-IFA 028 [i.15], annex A.		

Table 5.5.2.4-2 describes the flow for instantiating an NS for SP1. There is no difference to the standard flow. The description only highlights some aspects related to tenants, which is mainly during steps 2, 5, 9, 10, 11, 15, 18, 22 and 23. The steps for instantiating the nested NS via the Or-Or reference point are similar to the steps for instantiating the nesting NS via OS-Ma-nfvo, see ETSI GS NFV-IFA 030 [i.16] and ETSI GR NFV-IFA 028 [i.15].

Table 5.5.2.4-2: Use case #4, base flow for instantiating an NS for SP1

#	Flow	Description
1	SP1 -> NFVO1	SP1 requests instantiation of the NS.
2	NFVO1	NFVO1 validates the requests. This includes that NFVO1 checks whether SP1 is allowed to instantiate the NS. NFVO1 checks that all VNFs are onboarded and are allowed to be instantiated by SP1. NFVO1 checks resource availability for the VNF instantiation. See note 1 and note 2.
3	NFVO1 -> SP1	NFVO1 acknowledges the NS instantiation request.
4	SP1 -> NFVO1	SP1 subscribes for the relevant notifications. See note 3.
5	NFVO1	NFVO1 validates the requests. This includes that NFVO1 checks whether SP1 is allowed to subscribe to these notifications.
6	NFVO1 -> SP1	NFVO1 acknowledges the subscriptions.
7	NFVO1 -> VNFM1	NFVO1 requests instantiation of the VNFs as defined for the NS according to the deployment flavour indicating resource groups appropriate to SP1. See note 1, note 4 and note 6.
8	VNFM1	VNFM1 validates the request. This includes package validation.
9	VNFM1 -> VIM	VNFM1 requests the resources for the VNFs indicating resource groups appropriate to SP1. See note 5.
10	VIM	VIM allocates the resources keeping track of the resource groups.
11	VIM -> VNFM1	VIM provides VNFM1 with the resources for the VNFs for the NS.
12	VNFM1 -> VNF	VNFM1 finalizes the instantiation which can include configuration and VNF specific operations.
13	VNFM1 -> NFVO1	VNFM1 acknowledges the VNF instantiation.
14	NFVO1 -> NFVO3	NFVO1 requests instantiation of the nested NS (refer to NS3 in Figure 5.5.1-1). See note 6.
15	NFVO3	NFVO3 validates the requests. This includes that NFVO3 checks whether SP1 is allowed to instantiate the NS. Therefore NFVO3 checks the SP who requested the instantiation at NFVO1. It is expected that this information can be provided via the Or-Or reference point. See note 7. NFVO3 checks that all VNFs are onboarded and are allowed to be instantiated by SP1. NFVO3 checks resource availability for the VNF instantiation. See note 1 and note 2.
16	NFVO3 -> NFVO1	NFVO3 acknowledges the NS instantiation request.
17	NFVO1 -> NFVO3	NFVO1 subscribes for the relevant notifications. See note 3.
18	NFVO3	NFVO3 validates the requests. This includes that NFVO3 checks whether NFVO1 is allowed to subscribe to these notifications.
19	NFVO3 -> NFVO1	NFVO3 acknowledges the subscriptions.
20	NFVO3 -> VNFM3	NFVO requests instantiation of the VNFs as defined for the NS according to the deployment flavour indicating resource groups appropriate for SP1. See note 1, note 4 and note 6.
21	VNFM3	VNFM3 validates the request. This includes package validation.
22	VNFM3 -> VIM	VNFM3 requests the resources for the VNFs indicating resource groups appropriate for SP1. See note 5.
23	VIM	VIM allocates the resources keeping track of the resource groups.
24	VIM -> VNFM3	VIM provides VNFM3 with the resources for the VNFs for the nested NS.
25	VNFM3 -> VNF	VNFM3 finalizes the instantiation which can include configuration and VNF specific operations.
26	VNFM3 -> NFVO3	VNFM3 acknowledges the VNF instantiation.
27	NFVO3 -> NFVO1	NFVO3 acknowledges the instantiation of the nested NS.
28	NFVO1	When all VNFs and nested NSs are instantiated, NFVO1 completes the instantiation.
29	NFVO1 -> SP1	NFVO1 acknowledges the NS instantiation.

NOTE 1: This step is simplified to avoid two flows for direct or indirect mode. Also allocation of other NS resources and the granting dialogue are not shown.

NOTE 2: Resource availability includes availability within resource limits for SP1.

NOTE 3: Subscription can also be done earlier.

NOTE 4: This Use Case does not cover nested NSs which will be covered in separate Use Case.

NOTE 5: VIM distinguishes SP1 and SP2, so the NFVI is able to provide resource isolation, e.g. network isolation.

NOTE 6: The order of instantiation depends on the dependency attributes in the NS. Thus if a VNF of the nesting NS is dependent on the nested NS, step 14-27 will be executed first. If there are no dependencies, steps 7-13 and 14-27 can be executed in parallel.

NOTE 7: Tracking tenancy for the nested NS is not currently covered by the Or-Or reference point, see clause 5.3.2 of ETSI GS NFV-IFA 030 [i.16]. As described in the analysis of the present Use Case, clause 5.5.4, it is recommended to extend the Or-Or reference point accordingly.

Onboarding and instantiation for SP2 are similar.

5.5.2.5 Post-Conditions

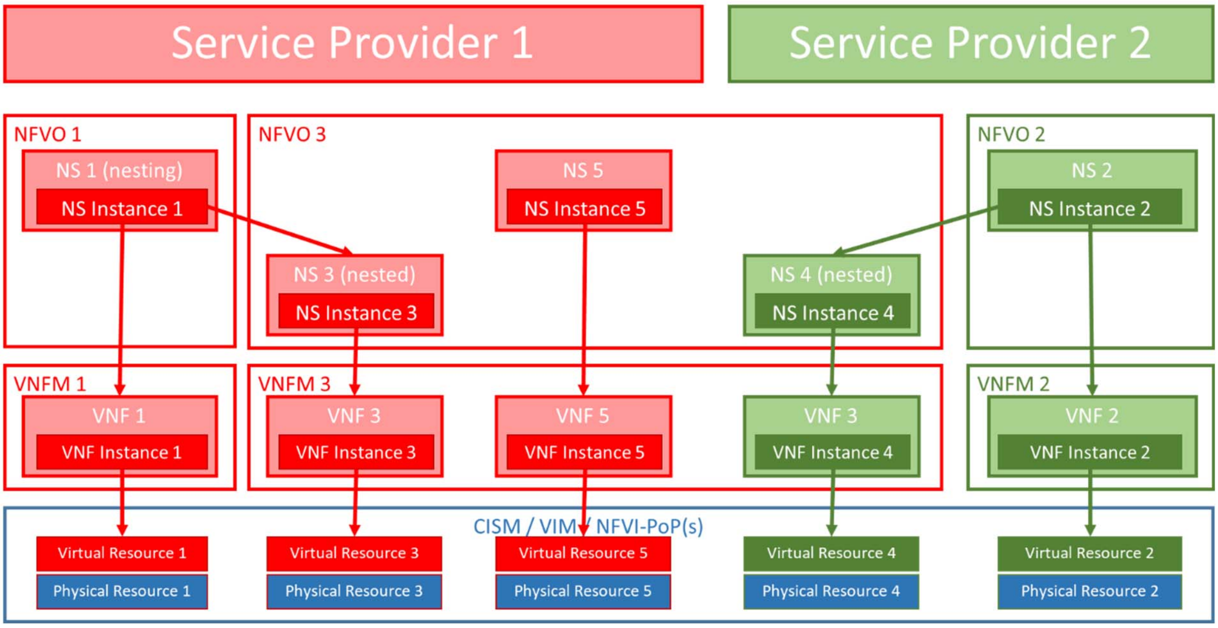
Table 5.5.2.5-1 describes the post-conditions.

Table 5.5.2.5-1: Use case #4, post-conditions

#	Post-condition	Description
1	VNFs and NSs are correctly instantiated.	
2	Resources are allocated as per Service Provider information.	
3	NFV-MANO FBs have all information to provide management isolation.	This includes subscription to notifications.
4	NFVI has all information to isolate resources between Service Providers.	This includes the appropriate information for network isolation.

5.5.3 Variants

Figure 5.5.3-1 shows in addition to Figure 5.5.1-1 a network service NS5 instantiated in NFVO3 side by side to the nested NS3.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.5.3-1: Nested Network Service and other NS by same tenant

In this variant, the resources of NS3 and NS5, both orchestrated from NFVO3, do not expect isolation. NFVO3 can know this only, if the tenancy information coming over Or-Or reference point from NFVO1 and the tenancy information coming directly from Service Provider 1 can be matched.

5.5.4 Analysis

As shown in the flow in clause 5.5.2.4 and similar to Use Case #1 and #2, the information about isolation constraints is expected to be provided on several reference points:

- to the VIM via Vi-Vnfm and Or-Vi reference points;
- to the VNFM via the Or-Vnfm reference point; and
- to the NFVO of nested NSs via the Or-Or reference point.

NOTE: There is a difference between direct and indirect mode of resource allocation by the VNFM. In indirect mode, i.e. when the VNFM allocates resources via the NFVO, a different mechanism could be used.

The variant in clause 5.5.3 shows that it might be useful to provide an identification of the tenant (that is of the service provider or consumer) on the Or-Or and Os-Ma-nfvo reference points. Otherwise it will be difficult for NFVO3 (see Figure 5.5.3-1) to know that NS3 and NS5 and their resources do not expect isolation. On the other reference points, it is sufficient to provide information about groups of resources, within which sharing is possible, whereas the groups themselves expect to be isolated against each other.

Key issue #2 in clause 6.2 analyses how to provide the tenancy information on the Or-Or and Os-Ma-nfvo reference points. Solution #2.1 introduces a tenancy management functionality, to be able to use identifiers for the consumers/service provider/tenants. It is recommended that the concept of providing the tenancy information to NFVO considers also that packages, NSs and VNFs can be shared between multiple Service Providers, which is elaborated in Use Case #9, see clause 5.10. Also it is recommended that it covers different levels of permission to access, use or change packages, NSs and VNFs. See solution proposal #2.2 in clause 6.2.3 and related recommendations in Table 6.2.7-1.

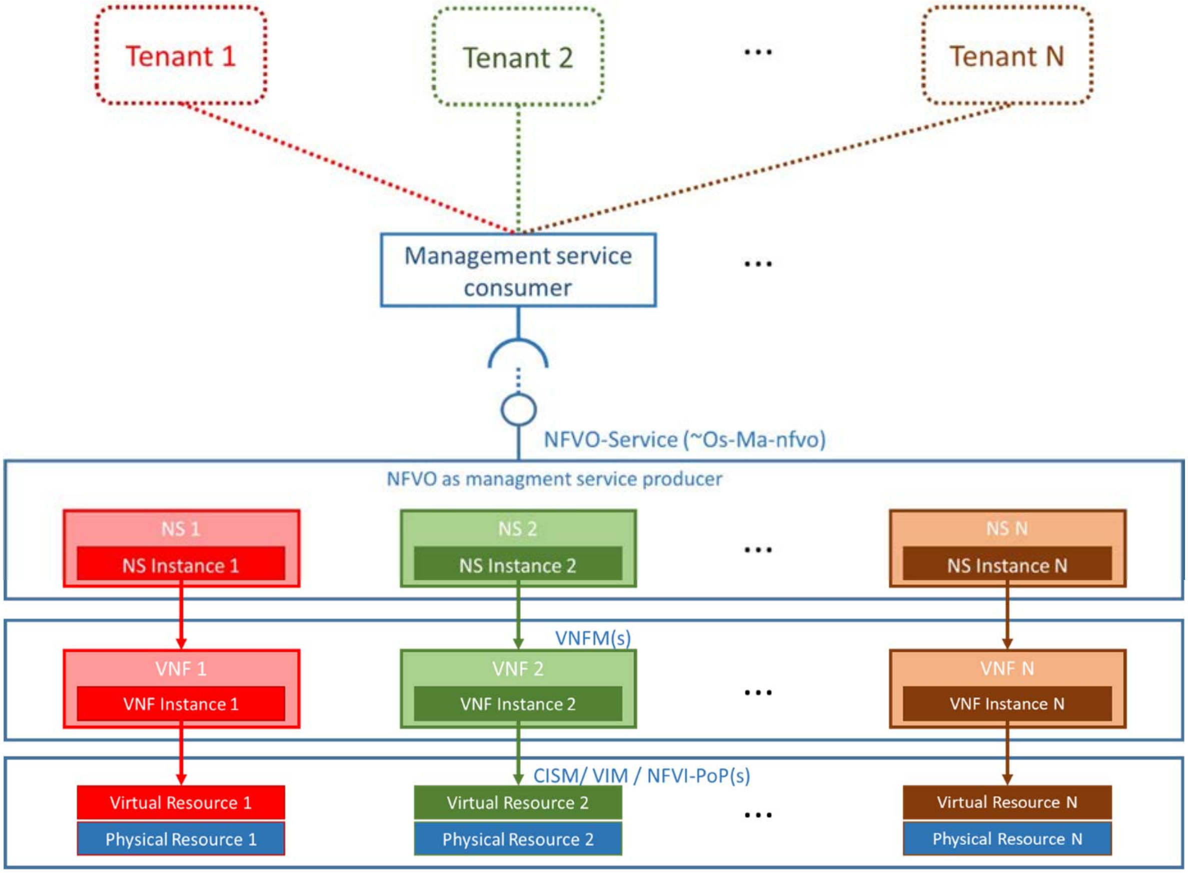
It is recommended to revisit ETSI GS NFV-IFA 030 [i.16] to add missing requirements, attributes and parameters on the Or-Or reference point, which is covered by recommendation Mtenant.tenantmgmt.04 in Table 6.2.7-1.

5.6 Use Case #5: Tenants of a service provider

5.6.1 Motivation

3GPP TR 28.804 [i.23] analyses environments where a service provider offers 3GPP management services to multiple tenants. For this Use Case, there are two different options foreseen, which are illustrated in Figures 5.6.1-1 and 5.6.1-2.

In option 1, the tenants are represented by a single management service consumer instance. Related management data isolation is currently not required by 3GPP. Only fault and performance data are identified per tenant.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.6.1-1: Option 1: Multiple tenants represented by single management service consumer, while consuming the same management service

In this option, the NFVO is not aware of the different tenants. Isolation between the NSs of the different tenants is not provided. Fault and performance data identification can be done outside the NFV system.

In option 2, each tenant is represented by a dedicated management service consumer. The related management data isolation is partially provided (e.g. by dedicated management service consumer).

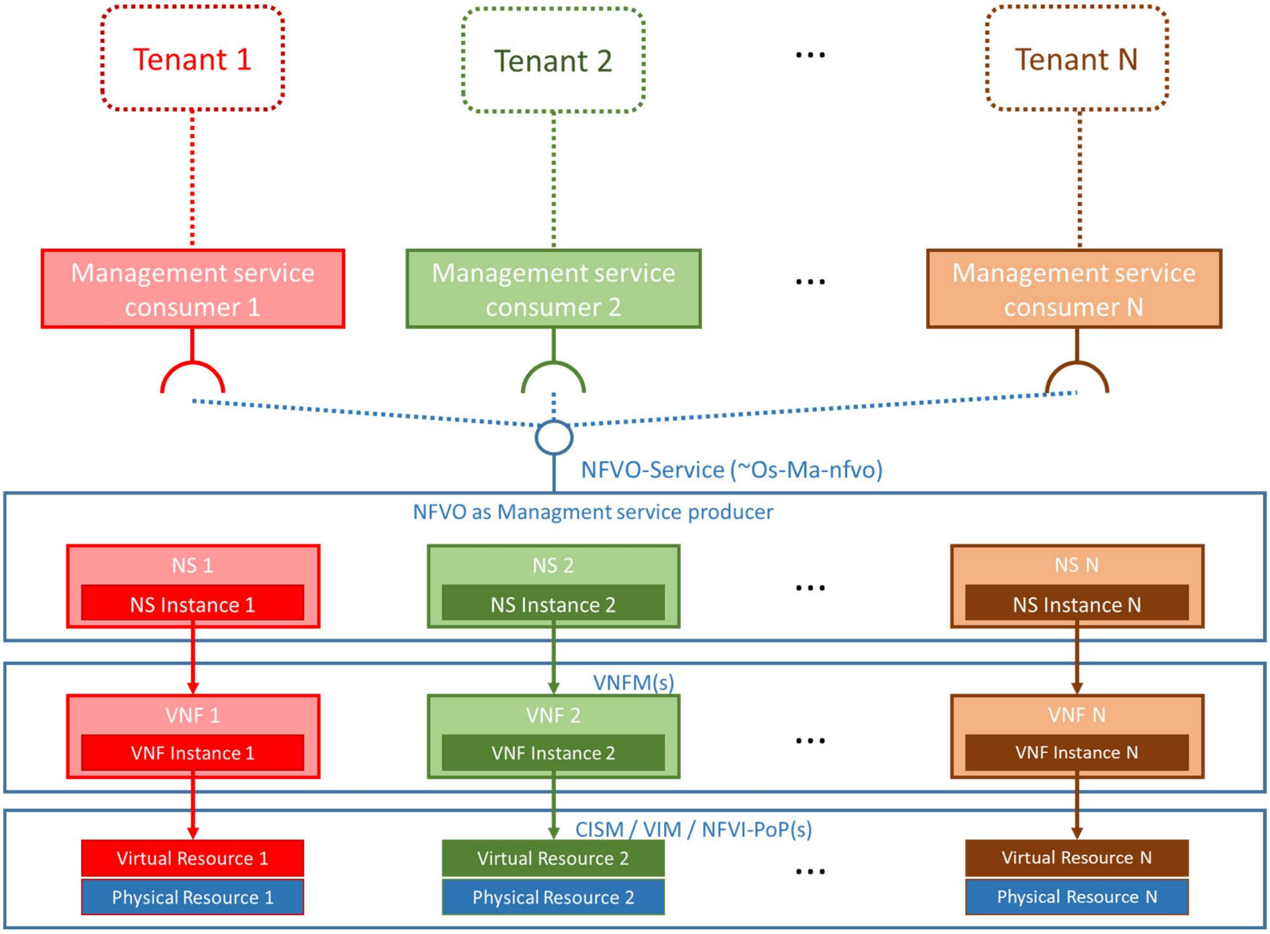


Figure 5.6.1-2: Option 2: Each tenant represented by dedicated management service consumer, while consuming the same management service

NOTE 1: The figure above showing "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

In this option, each management service consumer can act as a separate NFV-MANO tenant and isolation can be provided as illustrated in the other Use Cases.

NOTE 2: In this option, the management service consumers could be instances of the same NFV-MANO provider.

According to 3GPP TR 28.804 [i.23], combinations of these options will be possible.

5.6.2 Detailed User Story

In this Use Case, the flows are identical to Use Case #2, see clause 5.3. The only difference is that in option 1, the three tenants use the same management service consumer to interact with NFVO and to manage their network services, VNFs and resources.

5.6.3 Variants

This covers also Use Cases listed in the ENISA NFV 5G security report [i.27]. Multi-tenancy here includes also the customers of service providers, e.g. from vertical industries. In this case, and considering the scenarios introduced in clause 5.5.1, tenants from vertical industries would use the same NFV environment in parallel to the service provider itself. ENISA NFV 5G security report [i.27] describes the resulting security implications in more detail.

5.6.4 Analysis

Main aspects of this Use Case are covered in Use Case #2, see clause 5.3.

As in Use Case #2, the resource isolation can be provided without direct information about tenancy; it is sufficient to provide information about groups of resources, within which sharing is possible, whereas the groups itself are to be isolated against each other.

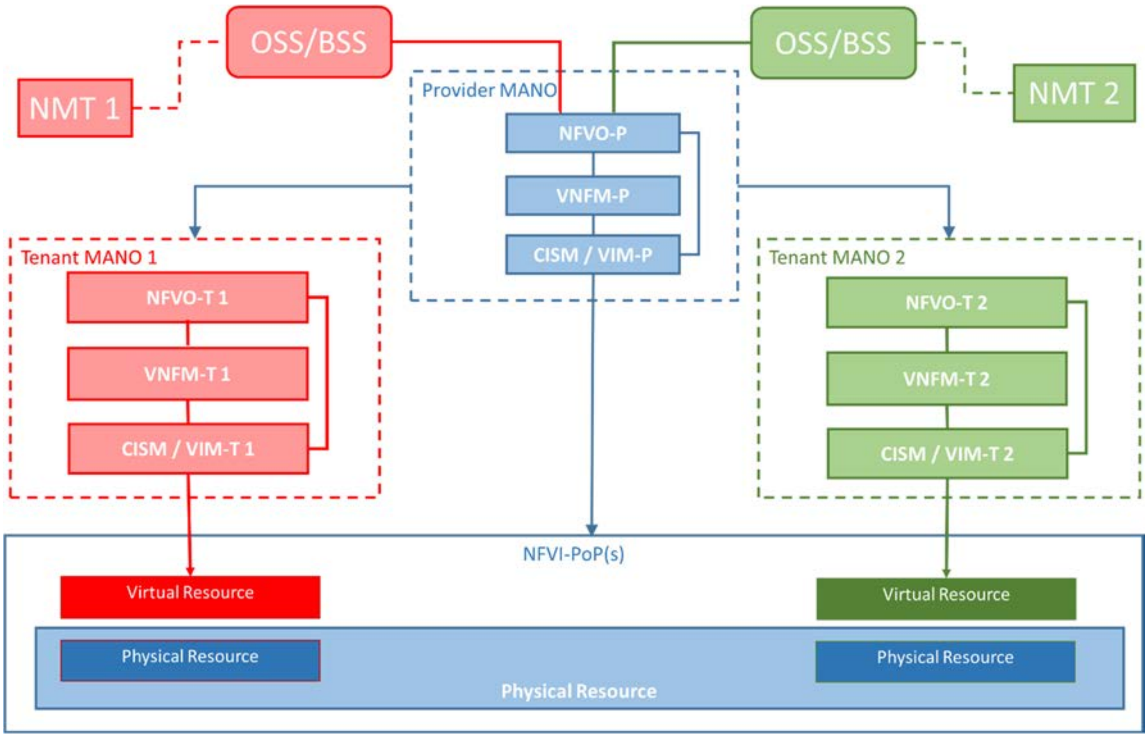
As in Use Case #2, it is recommended that the resource group concepts as described in ETSI GS NFV-IFA 005 [i.5], ETSI GS NFV-IFA 006 [i.6] and ETSI GS NFV-IFA 007 [i.7] are improved. Further details on recommended extensions of the concepts of resource groups are analysed in key issue #1, solution #1.3, see clause 6.1.4.

Also as in Use Case #2, it is recommended that constraints for management isolation are defined, which will protect NSs, VNFs and resources assigned for a tenant against management from a different tenant. In option 1 the management service consumer is acting for multiple tenants. The expected management isolation therefore could be provided by the management service consumer, or the management consumer could interact with NFVO using different tenancy information, e.g. different access tokens, see key issue #2, solution #2.2, clause 6.2.3.

5.7 Use Case #6: Two users with their own MANO stack managed by provider MANO

5.7.1 Motivation

This Use Case describes the scenario where the two users are provided with own NFV-MANO stacks, referred to as tenant MANO (MANO-T) stack, whereas the management autonomy of the MANO-T stack is managed and monitored by the provider MANO (MANO-P) system. In this Use Case it is assumed that the MANO-P system is also owned by the owner of the NFVI-PoP. The main motivation is not only to decentralize the NFV MANO system but also provide each NMT with the autonomy to manage their own services and resources. The level of management autonomy granted to the MANO-T stack can be determined and managed by the MANO-P system. The scope of management autonomy can also be negotiated between the MANO-T and the MANO-P. The MANO-P can also monitor the operations of the MANO-T stack(s) to ensure compliance with the agreed management autonomy. Figure 5.7.1-1 shows two MANO-T systems that are being used by two NMTs. However, both these MANO-T systems and the NFVI resources are under the full administrative control of the MANO-P. The functional and operational scope of the respective MANO-T systems is negotiated and determined by the MANO-P system, which will give the NMTs the autonomy to manage their own domains within the prescribed scope.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.7.1-1: Tenant MANO systems managed by the provider MANO system

5.7.2 Detailed User Story

5.7.2.1 Summary

This Use Case assumes the MANO-T systems as separate entity from the MANO-P system that can be instantiated and managed like an NS. In this Use Case both the NMTs will request the MANO-P for the provisioning and instantiation of the respective MANO-T systems, whereas the respective MANO-T systems are fully isolated. For the sake of explanation, the MANO-T system is considered to have the same functional composition as the NFV-MANO system. The request can be made via the NMT's management system, such as OSS/BSS over the Os-Ma-nfvo reference point, see ETSI GS NFV-IFA 013 [i.13]. The MANO-T system consists of the FBs referred to as tenant NFVO (NFVO-T), tenant VNFM (VNFM-T) and tenant VIM (VIM-T), to distinguish from the FBs of the MANO-P system which are referred to as provider NFVO (NFVO-P), provider VNFM (VNFM-P) and provider VIM (VIM-P) as depicted in Figure 5.7.1-1. It is assumed that the MANO-P system is able to distinguish between the requesting NMTs.

In the request, the NMTs can additionally specify the resources quota over which the requested MANO-T system will have management control. The NMTs can also request for a full MANO-T stack or a partial MANO-T stack. A full MANO-T stack will have all the main FBs of a NFV-MANO system (i.e. NFVO-T, VNFM-T and VIM-T), while a partial MANO-T stack will not have a complete NFV-MANO stack. For example, the NMT can request for a MANO-T stack composed of only an NFVO-T and VNFM-T FBs. In such a case, the VIM functionality will be provided by the VIM instance of the MANO-P system. The interaction between the MANO-P system and the MANO-T system can be realized over existing NFV-MANO reference points with new and/or extended interfaces, or new reference points can be realized.

In the request for the MANO-T system, the NMT can also specify the scope of the management autonomy that it requires of the MANO-T system.

NOTE: The scope of the management autonomy can also be requested and negotiated after the MANO-T system has been instantiated.

The scope of the management autonomy is requested and negotiated by the requesting NMT with the MANO-P system. The management autonomy specifies the type and boundaries on which lifecycle management operations a MANO-T system is allowed to execute within its reserved resource quota/pool/zone without involving the MANO-P system. For example, a MANO-T system might be allowed to instantiate NS instances within the resource quota of the NMT. It might also be allowed to scale the VNFC instance of the NMT's NS instance. However, the respective NMT might not be allowed to increase resource quota or scale-out or scale-up beyond a certain limit, or migrate virtualized resources or instantiate VNFs beyond a certain number, etc. MANO-T will reject any operation that is outside the scope of MLA, which can then prompt the NMT to negotiate a new MLA with the MANO-P. It is noted that in case of full-autonomy, the MANO-T system can execute full MANO functions even when MANO-P is down.

The management agreement negotiated between the NMTs and the MANO-P will constitute a Management Level Agreement (MLA), which will be specified in an MLA descriptor file. An MLA template is used for the purpose of MLA negotiation between the NMT and the MANO-P system, which consists of requested permissions about LCM interfaces, operations and operational bounds requested by the NMT. Thus, a MANO-T system will operate and function like a regular NFV-MANO system however; the functional and operation scope is bounded by the MLA. The MANO-P will be responsible for the monitoring against MLA violation and for enforcing compliance on LCM decisions on actions that fall within the MLA scope. The MANO-P is also responsible for the reliable availability and operation of the MANO-T system. From the MANO-P perspective, all LCM operations that are valid over VNFs are also valid over the MANO-T system instances such as NFVO-T, VNFM-T and VIM-T. This implies the need to have well defined reference points between the corresponding FBs of the MANO-P system and the MANO-T system.

5.7.2.2 Example flow instantiating MANO-T instance for an NMT

5.7.2.2.1 Actor(s)

Table 5.7.2.2.1-1 describes the Use Case actors and roles.

Table 5.7.2.2.1-1: Use case #6, actors and roles

#	Actor	Description
1	NMT1	OSS or other management system of service provider 1.
2	NMT2	OSS or other management system of service provider 2.
3	NFVO-P	NFV Orchestrator of the MANO-P system for the management of the MANO-T instances and NS instances involved.
4	VNFM-P	VNF Manager of the MANO-P system.
5	VIM-P	VIM of the MANO-P system managing the NFVI hosting all resources involved.
6	MANO-T	The NFV-MANO system that is provided to the tenant.
7	NFVO-T	NFV Orchestrator of the MANO-T system for the management of the NS instances involved.
8	VNFM-T	VNF Manager of the MANO-T system for the management of the VNFs involved.
9	VIM-T	VIM of the MANO-T system managing the NFVI resources that are assigned to the NMT.
10	Provider of NFV-MANO	The organization responsible for the provision and operation of the NFVI and MANO-P system.

5.7.2.2.2 Pre-Conditions

Table 5.7.2.2.2-1 describes the pre-conditions.

Table 5.7.2.2.2-1: Use case #6, MANO-T system instantiation process, pre-conditions

#	Pre-condition	Description
1	Provider NFV-MANO (VIM-P, NFVO-P and VNFM-P) is operational.	
2	NMT1 and NMT2 have established the necessary business relationship with the provider of the NFV environment allowing them to deploy their services via their respective OSS/BSS.	
3	NMT1 and NMT2 have prepared the necessary NFV packages and templates (e.g. NSD, VNFD) for the on-boarding.	
4	NMT1 and NMT2 have prepared the necessary MLA template, which includes the MLA parameters specifying the interfaces and the operations management autonomy requirements.	The MLA template will list the necessary interfaces and operations that the NMT wants to execute via MANO-T as part of deploying and managing its NS instance(s) over the respective resource quota.

5.7.2.2.3 Description

Table 5.7.2.2.3-1 describes the flow for enabling the Provider of the NFV-MANO for deploying the MANO-T system for NMT1. The same flow applies for any other NMT that respectively wants to deploy MANO-T system for managing own resources with the Provider of NFV-MANO.

Table 5.7.2.2.3-1: Use case #6, base flow for instantiating a MANO-T system for NMT1

#	Flow	Description
1	NMT1 → NFVO-P	NMT1 requests the NFVO of the MANO-P system, via the OSS/BSS, to reserve the total NFVI's virtualized resource quota i.e. resource quota needed for the MANO-T system and all planned NSs.
2	NFVO-P → VIM-P	The NFVO-P of the MANO-P system sends the NFVI resource quota request to VIM-P as specified in ETSI GS NFV-IFA 005 [i.5].
3	VIM-P	The VIM-P of the MANO-P system parses the virtualized resource quota request from the NFVO-P and sets up the required virtualized resource quota.
4	VIM-P → NFVO-P	The VIM-P of the MANO-P system returns to the NFVO-P information on the newly created resource quota plus any additional information about the created quota request operation.
5	NFVO-P → NMT1	The NFVO-P of the MANO-P system informs the NMT1 of the newly reserved virtualized resource quota.
6	NMT1 → NFVO-P	NMT1 requests the NFVO-P of the MANO-P system, via the OSS/BSS, to on-board the MANO-T system, by providing the MLA template.
7	NFVO-P	The NFVO-P of the MANO-P system parses the MLA template to verify if the MLA parameters (i.e. management interfaces and operations) for which the NMT1 is requesting permission for; can be deployed. See note 1.
8	NFVO-P → NMT1	If the requested MLA parameters is not acceptable by the NFVO-P, the NFVO-P of the MANO-P system, via the provider's management system, responds with a reject notification to the NMT1 request, and specifies the management interfaces and the operations, which the NMT1 can execute via the MANO-T system. See note 1.
9	NMT1 → NFVO-P	The NMT1 can revise the MLA parameters based on the information received in the reject notification and requests the NFVO-P of the MANO-P system, via the OSS/BSS, to on-board the MANO-T system, by providing the revised MLA template.
10	NFVO-P	The NFVO-P of the MANO-P system parses the revised MLA template to verify if the MLA parameters (i.e. management interfaces and operations) for which the NMT1 is requesting permission; can be deployed. See note 1.
11	NFVO-P →NMT1	If the requested parameters in the revised MLA template is acceptable to the NFVO-P, the NFVO-P of the MANO-P system, via the OSS/BSS, sends a positive acknowledgment and, optionally, additional information. See note 2.
12	NMT1 → NFVO-P	The NMT1 acknowledges the management interfaces and the operations permitted by the NFVO-P for the NMT1 to execute via the MANO-T system.
13	NFVO-P	The NFVO-P of the MANO-P system instantiates the MANO-T FBs (NFVO-T and/or VNFM-T and/or VIM-T) and configures the respective FBs of the instantiated MANO-T system as per the MLA. See note 3.
NOTE 1: This exchange of MLA parameters between the NFVO-P and the NMT1 is part of the MLA negotiation process where the MANO-P and NMT1 agree on the set of interfaces and operations that the NMT1's MANO-T system can have permission to execute. The MANO-P can allow or disallow some or all of the management interfaces and operations for which the NMT1 has requested permission to execute in the MLA template, which depends on the policy of the Provider of NFV-MANO. NOTE 2: The additional information can contain information about the MANO-T system to be instantiated (e.g. MANO-T system instance id, agreed MLA parameters, access information, prohibited MANO operations, etc.), and additionally can contain subscription offerings related to enhancing the operational scope of the MANO-T system. NOTE 3: The instantiation process of the MANO-T system is similar to that of an NS instance.		

5.7.2.2.4 Post-Conditions

Table 5.7.2.2.4-1 describes the post-conditions.

Table 5.7.2.2.4-1: Use case #6, MANO-T system instantiation process, post-conditions

#	Post-condition	Description
1	The MANO-T FBs are correctly instantiated.	
2	The MANO-T FBs are configured based on the agreed MLA.	

5.7.2.3 Example flow of MANO-T instance performing NS LCM operation

5.7.2.3.1 Actor(s)

The actors are similar to those described in Table 5.7.2.2.1-1.

5.7.2.3.2 Pre-Conditions

The preconditions are similar to those described in Table 5.7.2.2.4-1.

5.7.2.3.3 Description

Table 5.7.2.3.3-1 describes the flow depicting a scenario where a MANO-T system is performing LCM operations on an instantiated NS instance as per the MLA bounds. This scenario assumes that the MLA imposes a limit on the number of VNFs that the MANO-T system is allowed to scale-out. The same flow applies for any other NMT that respectively performs LCM operations within the bounds prescribed in the MLA.

Table 5.7.2.3.3-1: Use case #6, base flow of a MANO-T system performing LCM operations on an NS instance

#	Flow	Description
1	NMT1 → NFVO-T	NMT1, via the OSS, requests NFVO-T to onboard the NS, providing the NSD.
2	NFVO-T	NFVO-T executes the onboarding.
3	NFVO-T → NMT1	NFVO-T acknowledges the onboarding to the NMT1 via the OSS.
4	NMT1 → NFVO-T	NMT1, via the OSS, subscribes for notifications.
5	NMT1 → NFVO-T	NMT1, via the OSS, requests NFVO-T to onboard the VNFs, providing the VNF packages and VNFDs.
6	NFVO-T	NFVO-T executes the onboarding.
7	NFVO-T → NMT1	NFVO-T acknowledges the onboarding to the NMT1
8	MANO-T	The MANO-T system is monitoring the NS instance as per the NSD. Due to increase in traffic load, the MANO-T needs to scale-out by adding 3 more VNF instances, which is within the limits specified in the MLA. The MANO-T scales-out 3 instances of VNFs.
9	NFVO-T → NMT1	The NFVO-T, via the OSS, informs the NMT1 of the result of the LCM operation by sending a notification message with additional information e.g. the NS identifier, the traffic load, the number and identifiers of the scaled VNFs.
10	NMT1 → NFVO-T	The NMT1, via the OSS, sends an acknowledgment notification to the NFVO-T.
11	MANO-T	The MANO-T system continues monitoring the NS instance as per the NSD. Due to further increase in traffic load, the MANO-T needs to scale-out by an additional 2 more VNF instances, which exceeds the scale-out limit specified in the MLA.
12	NFVO-T → NMT1	The NFVO-T, via the OSS, informs the NMT1 of the required LCM operation by sending a notification message with additional information e.g. the NS identifier, the traffic load, the total number of additional VNFs required, the additional resource requirements, the MLA limitations, etc.
13	NMT1	The NMT1 can have different options to exercise in the situation where a LCM operation is needed but exceeds MLA limitations. See note 1 and note 2.
NOTE 1: The NMT1 informs the NFVO-T that it is re-negotiating an MLA with the MANO-P with enhanced scope. In this case, the MLA negotiation process is similar to the process described in Table 5.7.2.2.3-1 in clause 5.7.2.2.3. NOTE 2: The NMT1 can request the MANO-T and/or MANO-P to allow for the execution of LCM operation exceeding the MLA bounds. One reason for such a request would be if the NMT1 has the resources available within its reserved resource quota to allow exceeding the MLA bounds without exceeding the reserved resource quota.		

5.7.2.3.4 Post-Conditions

Table 5.7.2.3.4-1 describes the post-conditions.

Table 5.7.2.3.4-1: Use case #6, MANO-T system performing LCM operations, post-conditions

#	Post-condition	Description
1	The MANO-T system continues to perform LCM operations on NS instance within MLA prescribed bounds.	

5.7.2.4 Example flow of LCM operation exceeding MLA bounds

5.7.2.4.1 Actor(s)

The actors are similar to those described in Table 5.7.2.2.1-1.

5.7.2.4.2 Pre-Conditions

Table 5.7.2.4.2-1 describes the preconditions.

Table 5.7.2.4.2-1: Use case #6, MANO-T system performing LCM operations exceeding MLA, pre-conditions

#	Pre-condition	Description
1	The MANO-T system is monitoring the NS instance for the traffic load situation.	

5.7.2.4.3 Description

Table 5.7.2.4.3-1 describes the flow depicting a scenario where an LCM operation on an NS instance is required that exceeds MLA bounds. This scenario assumes that the MLA imposes a limit on the number of VNFs that the MANO-T system is allowed to scale-out. The traffic load situation demands the scaling-out of VNF instances exceeding the limitation in the MLA. The same flow applies for any other NMT that can perform LCM operations exceeding the bounds prescribed in the MLA.

Table 5.7.2.4.3-1: Use case #6, base flow of a MANO-T system managing a LCM operation request exceeding MLA bounds

#	Flow	Description
1	MANO-T	The MANO-T system is monitoring the NS instance as per the NSD. Due to further increase in traffic load, the MANO-T needs to scale-out by 2 additional VNF instances, which exceeds the scale-out limit specified in the MLA.
2	NFVO-T → NMT1	The NFVO-T, via the OSS, informs the NMT1 of the need to perform an LCM operation by sending a notification message with additional information e.g. the NS identifier, the traffic load, the total number of additional VNFs required, the available resources, the additional resources required, the MLA limitations, etc.
3	NMT1 → NFVO-T	In the case there are resources available within the tenant's resource domain, the NMT1 requests NFVO-T to seek permission from the NFVO-P to execute the scale operation. See note 1.
4	NFVO-T → NFVO-P	The NFVO-T sends a request notification to NFVO-P seeking permission to execute the scale operation. The request notification can carry additional information such as the total number of additional VNFs required, the types of VNFs, the reason for scaling (e.g. load condition), the available resources, the additional resource requirements, the NS identifier, etc.
5	NFVO-P	The NFVO-P parses the request in view of the available resources assigned to the tenant (i.e. NMT1) and the MANO-P policy as prescribed by the Provider of NFV-MANO.
6	NFVO-P → NFVO-T	The NFVO-P sends a response notification indicating one of the following actions: (a) The NFVO-P disallows the execution of the scale operation in case the tenant's resources are not enough or the MANO-P policy does not allow such exceptions. See note 2. (b) The NFVO-P allows the execution of the scale operation in case the tenant resources are available and the MANO-P policy allows for such an exception. Such an allowance can or cannot be with preconditions. See note 3.
7	MANO-T	The MANO-T executes the action that is appropriate to the type of response notification from NFVO-P and the supplementary encoded information in the notification: (a) In case the NFVO-P disallows the execution of the requested operation, then the MANO-T does nothing. (b) In case the NFVO-P allows the execution of the requested operation, then the MANO-T scales-out 2 instances of VNFs.
8	NFVO-T → NMT1	The NFVO-T, via the OSS, informs the NMT1 of the result of the LCM operation by sending a notification message with additional information e.g. the NS identifier, the traffic load, the number and identifiers of the scaled VNFs (in case the operation was allowed), the information in the response notification from NFVO-P received by NFVO-T (in step 6 above), etc. See note 4.
NOTE 1: The NMT1 can also send a request directly to NFVO-P for permission to execute the operation within the resource quota. NOTE 2: The NFVO-P can include additional information in the request reject response notification, such as the reject policy code, the MANO-P policy for accepting such exceptional requests, etc. NOTE 3: The NFVO-P allows the NFVO-T to execute an LCM operation that is outside the MLA bounds with possible preconditions such as specifying the duration for which such an operation is valid in the response notification. After the expiry of the duration, the additional VNFs is scaled-in. NOTE 4: In case the LCM operation is disallowed by the MANO-P system, then in that case the NMT1 can consider renegotiating an MLA with enhanced scope of operations as described in Table 5.7.2.2.3-1.		

5.7.2.4.4 Post-Conditions

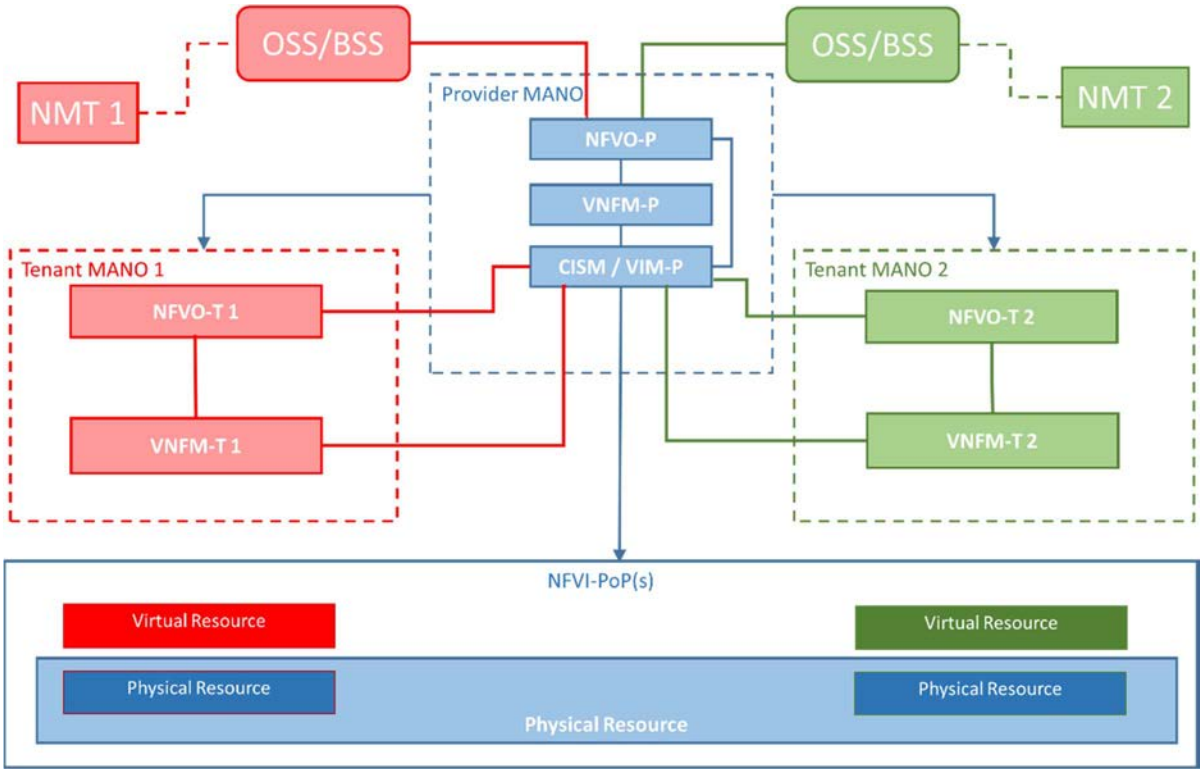
Table 5.7.2.4.4-1 describes the post-conditions.

Table 5.7.2.4.4-1: Use case #6, MANO-T system performing LCM operations exceeding MLA, post-conditions

#	Post-condition	Description
1	The MANO-T system continues to monitor and perform LCM operations on NS instance within MLA prescribed bounds.	

5.7.3 Variants

Figure 5.7.3-1 depicts a partial MANO-T deployment scenario by the NMTs. A deployment is considered partial when only specific FBs of the MANO-T system are deployed, either due to the NMT's request or due to some policy restrictions of the Provider of NFV-MANO. For example, the NMTs can request for the deployment of only NFVO-T and VNFM-T FBs, whereas the VIM services are provided by the VIM-P. In this scenario, the VIM-P is shared by the tenant MANO systems of NMT 1 and NMT 2 as illustrated in Figure 5.7.3-1.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.7.3-1: Tenant MANO systems sharing the provider VIM

5.7.4 Analysis

A key procedure for the instantiation of MANO-T system is the negotiation between the NMT and the MANO-P system in order to determine the operational scope and management autonomy of the MANO-T system. An MLA template is central to this negotiation process, which is executed over the Os-Ma-nfvo reference point utilizing relevant interfaces and information elements.

It is therefore recommended to revisit ETSI GS NFV-IFA 013 [i.13] and specify new interface for the execution of the MLA negotiation process over the Os-Ma-nfvo reference point, and specify relevant operations. For example, interfaces that will allow for the triggering of the MLA negotiation process, offline or runtime maintenance of the MLA template to support operations such as updating of the MLA template during the (re)negotiation process, and terminating an MLA template. This might also result in the specification of new information elements that can model the attributes to represent the MLA parameters and permissions agreed during the MLA negotiation process. The impact of the new interfaces and the information elements on existing interfaces of ETSI GS NFV-IFA 013 [i.13] is also recommended to be investigated and necessary extensions specified. For example, the Notify operation is likely to be extended with operations relevant to the maintenance notification of the MLA template.

Since the scope of the MLA negotiation depends on the internal policies of the Provider of NFV-MANO, it is recommended to review ETSI GS NFV-IFA 048 [i.30] in the context of defining policies governing the instantiation and the LCM of the MANO-T instances. These provisions can be in the form of extending existing information elements with new attributes, extending the definition of existing attributes, and/or defining new information elements. For example, the attribute *targetObjectType* of the Policy information element can be extended to allow values identifying the MANO-T instance and the associated FBs. It is to be noted that a separate Policy can be specified for each MANO-T instance.

As the MANO-P instantiates the MANO-T system similar to how it instantiates an NS, the MANO-T system can be considered as a managed object of the MANO-P. Similar to an NS which is composed of VNF instances, a MANO-T instance can be like an NS composed of NFVO-T, VNFM-T and VIM-T instances as VNF instances. Therefore, the LCM of the MANO-T system is managed by the MANO-P system. Besides the LCM of the MANO-T instance, the MANO-P system has the additional task of monitoring and enforcing the operation of the MANO-T systems within MLA bounds. This implies extending the operational scope of the NFVO-P of the MANO-P system, where it can process the MLA template and enforce the operations within MLA bounds.

This makes it necessary to specify descriptor files, similar to NS templates [i.14] and VNFD [i.11], for describing the MANO-T service and its associated FBs. Another option could be to extend the specification of the NS templates [i.14] and VNFD [i.11] to include artefacts of the MLA template. Moreover, reference points involved in the LCM of NS needs to be revisited, and is thus recommended to review ETSI GS NFV-IFA 005 [i.5], ETSI GS NFV-IFA 006 [i.6], ETSI GS NFV-IFA 007 [i.7] and ETSI GS NFV-IFA 008 [i.8]. For example, new interfaces and operations related to the monitoring and enforcement of the MANO-T operations within the MLA bounds can be included. It is also recommended to review the interfaces and information elements in ETSI GS NFV-IFA 005 [i.5], ETSI GS NFV-IFA 006 [i.6] in consideration of the variant described in clause 5.7.3.

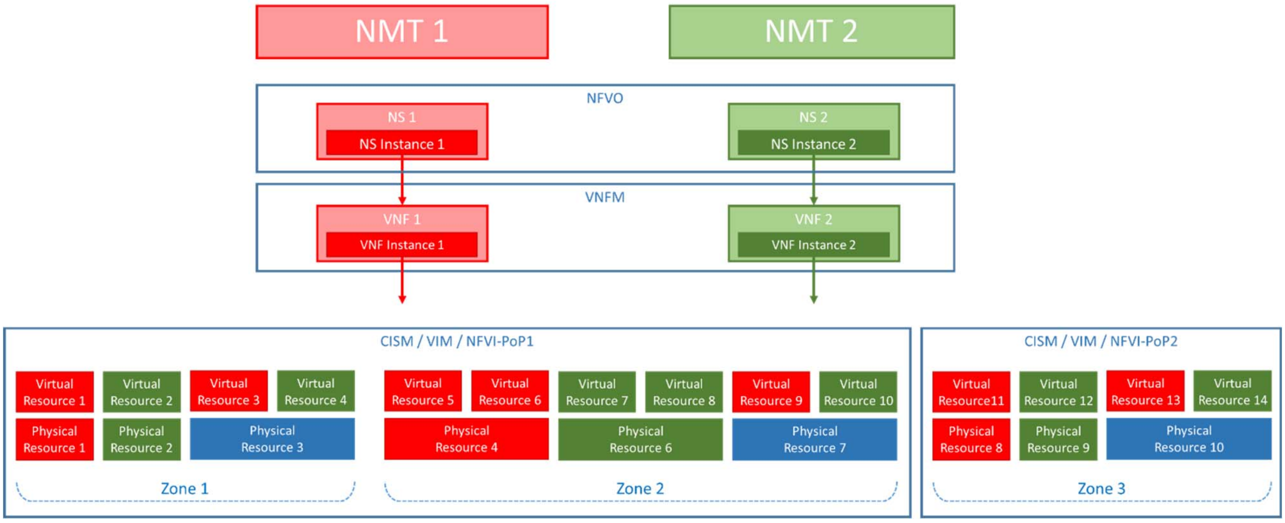
In view of the above analysis, the main challenges are the specification of the MLA template, and the management of the MANO-T instances. These two challenges are highlighted in clause 6.7 and different solution options are proposed in clause 6.7.2 and clause 6.7.3 for the respective challenges. Based on the different solution options, appropriate recommendations are provided in clause 6.7.4.

5.8 Use Case #7: Provide Isolation on different levels

5.8.1 Motivation

This Use Case illustrates how resource isolation can be done on various levels in an NFV architecture. This is a basic capability which is very important for security considerations, as being done e.g. in ETSI GS NFV-SEC 026 [i.26]. Security considerations can lead to additional recommendations, e.g. hardware enclaves and more fine-grained handling of affinity.

Figure 5.8.1-1 shows two NSs with resources that can be isolated in different ways. Some of the virtualized resources share physical resources, or physical resources can share zones in the same NFVI-PoP or be located in different NFVI-PoPs.



NOTE: "Virtual Resources" refers to different virtualization technologies, e.g. VMs and containers.

Figure 5.8.1-1: Resource isolation on different levels

Figure 5.8.1-1 shows isolation of virtual resource in different levels:

a) Virtual resources can share the same physical resource, as shown with Physical Resource 3, 7 and 10. These physical resources are hosting components of NS1 and NS2. Thus the isolation can be guaranteed by the virtualization layer, i.e. the hypervisor.

b) In case of virtual resources 1, 5, 6 and 11, virtual resources of NS1 (red) do not share the physical resources with resources of NS2 (green). Here the isolation is already guaranteed by separated hardware. However, physical servers can still share some network equipment.

c) The hardware in NFVI-PoPs can be organized into multiple zones. The entities within a zone can share physical environment, air-conditioning, power or networking (e.g. switches).

d) In case of geo-redundancy, resources can be deployed in different NFVI-PoPs. In the same way an additional isolation over NFVI-PoPs can be used.

Usually isolation is used to make sure that resources of one NS cannot affect another NS. In case A above, the virtualization layer provides the isolation by software means. The virtualized resources in this case can share CPUs or cores and the compute power can be divided between the tenants. Similarly the physical network bandwidth can be shared. Thus high usage by one virtualized resource could affect another tenant, if the virtualization layer does not manage limitations for a tenant. Also failures within one virtualized resource will not affect another tenant, if the virtualization layer provides proper isolation.

As shown in case B above, isolation can be provided at physical level, thus being independent of the isolation capabilities of the virtualization layer.

EXAMPLE: While VNF instances within the same NS instance do not expect strong isolation (however they can expect anti-affinity for redundancy and reliability reasons), the resources provided to different service providers can expect isolation on physical level.

Except for the use of geo-redundancy for reasons of disaster recovery, the high levels of isolation shown in cases c) and d) will typically be expected by service providers for business reasons, not for technical reasons.

5.8.2 Detailed User Story

5.8.2.1 Summary

In this Use Case, like in Use Case #2, clause 5.3, both NMTs instantiate their NSs with the same NFVO via the Os-Ma-nfvo reference point, see ETSI GS NFV-IFA 013 [i.13]. The Use Case illustrates how the different levels of isolation are related to the existing mechanism of affinity/anti-affinity rules.

ETSI GS NFV-IFA 010 [i.10] specifies requirements on NFVO (Nfvo.VnfRmpbNfvo.005 in clause 6.1.2) and VNFM (Vnfm.VnfRmpbVnfm.005 in clause 7.1.3) to "request to the VIM affinity and anti-affinity policies for the VNF's virtualised resources". These policies are defined using AffinityOrAntiAffinityGroup information elements (see ETSI GS NFV-IFA 014 [i.14], clause 6.3.5), which can be referenced from the NsDf, VnfProfile, VirtualLinkProfile or NsProfile. Thus the user can specify the levels of isolation by defining anti-affinity for these elements.

Every AffinityOrAntiAffinityGroup has an identification and defines whether members of the group will have affinity or anti-affinity to all other members of that group and on which level.

ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6] allow to dynamically create a Virtualised Compute Resource Affinity Or AntiAffinity Constraints Group.

The anti-affinity groups therefore provide a mechanism to indicate isolation constraints to the VIM.

This Use Case is described in two parts: clause 5.8.2.2 discusses the current concepts and how to define anti-affinity groups to specify the isolation constraints. In clause 5.8.2.3, a flow is provided similar to Use Case #2 in clause 5.3.2.4, that focuses on the information flow related to the levels of isolation.

5.8.2.2 Usage of anti-affinity groups for isolation

The main goal of introducing affinity and anti-affinity rules is coming from traffic optimization and reliability scenarios. VNF designers want to place VNFCs closely together if the traffic path always connects the same instances and thus use affinity rules. Anti-affinity rules are particularly important to achieve redundancy for fault tolerance. Therefore affinity and anti-affinity rules can be defined for the constituents of VNFs as described in ETSI GS NFV-IFA 011 [i.11].

In the same way, affinity and anti-affinity rules can be defined for the constituents of NSs and NS instances by the service provider when designing the network services, see ETSI GS NFV-IFA 014 [i.14].

In the case of multi-tenancy, anti-affinity is used to distinguish between the resources used by the different tenants. However, at design time, the members of an anti-affinity group are not known. E.g. if VNF1 of service provider 1 is expected to be isolated against VNF2 of service provider 2, an anti-affinity group could be used with VNF1 and VNF2 as members. This anti-affinity group cannot be part of the network service definition, since the service providers do not know of each other.

EXAMPLE:

Figure 5.8.2.2-1 illustrates two NMTs with 2 NSs each, every NS has 2 VNFs.

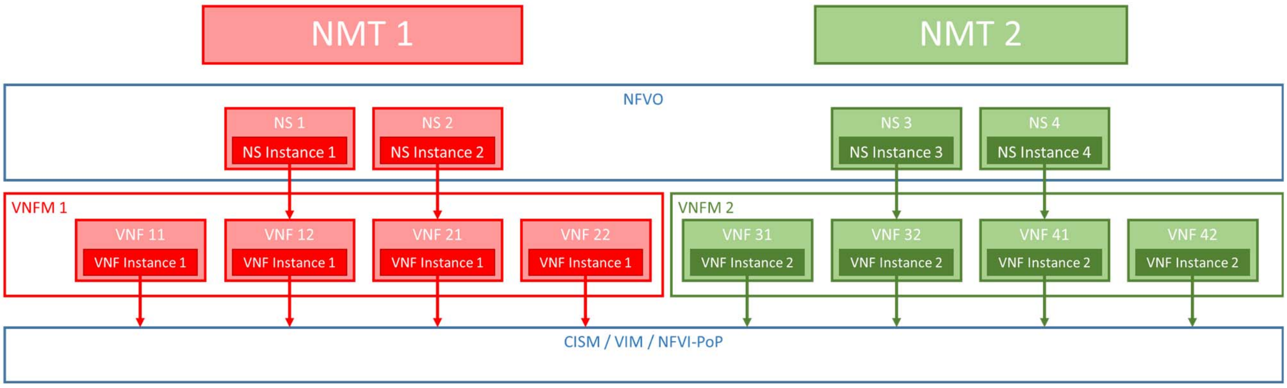


Figure 5.8.2.2-1: Exemplary scenario for the use of anti-affinity groups

In this example it is not possible to just define an anti-affinity group for all VNFs, because this would also force anti-affinity between the VNFs 11-22 of NMT1. To allow affinity of the VNFs inside an NS and at the same time anti-affinity of the VNFs to other NSs (in the figure above, affinity between red VNFs and anti-affinity between red and green), the localAffinityOrAntiAffinityRule can be used. This rule is part of the VNF profile as defined in clause 6.3.3 of ETSI GS NFV-IFA 014 [i.14], thus it can only be used if the VNFs in the example are based on the same VnfProfile.

NOTE 1: Both, affinityOrAntiAffinityGroup and localAffinityOrAntiAffinityRule are able to define the level of affinity, e.g. NFVI Pop, zone, zone-group or NFVI node.

NOTE 2: localAffinityOrAntiAffinityRule and affinityOrAntiAffinityGroupId are also available on VDU level (see clause 7.1.8.3 of ETSI GS NFV-IFA 011 [i.11]), but these cannot be used for multi-tenancy. Nevertheless, ETSI GS NFV-IFA 011 [i.11] in its clause B.2 has very useful examples showing the interworking of localAffinityOrAntiAffinityRule and affinityOrAntiAffinityGroupId.

In order to define affinity or anti-affinity between NSs, nested NSs can be used (see clause 6.3.2 of ETSI GS NFV-IFA 014 [i.14]). Figure 5.8.2.2-1 illustrates the nested NSs defining the affinity and anti-affinity for this scenario.

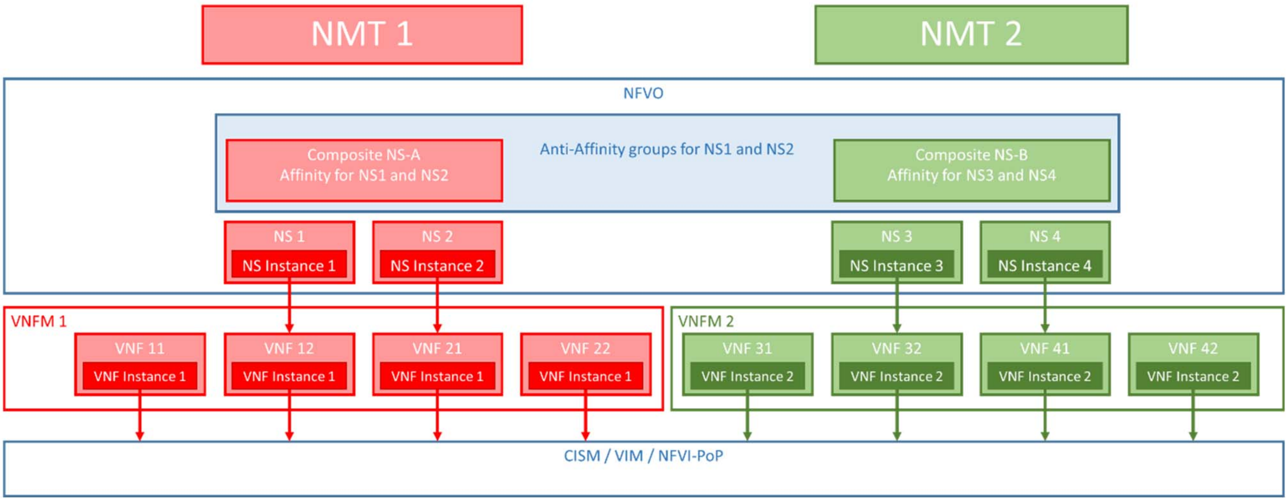


Figure 5.8.2.2-2: Example for the definition of affinity and anti-affinity groups for NSs

The affinity groups for NS 1 and NS 2 can be defined in the composite NS-A, while for NS 3 and NS 4 another NS-B can be used. For the anti-affinity between red and green NSs an additional anti-affinity group containing red and green NSs could be used (illustrated by the blue box, and defined in another composite NS). It is recommended that the interworking of the overlapping affinity/anti-affinity groups is defined, that is how conflicts between such definitions in different composite NSs can be resolved or rules are prioritized. Also here NS-A by NMT1 and NS-B by NMT2 are expected to use the same affinityOrAntiAffinityGroupId to specify the anti-affinity.

NOTE 3: For the nested NSs see also Use Case #4 in clause 5.5.

END OF EXAMPLE:

5.8.2.3 Example of flow instantiating network services with isolation

5.8.2.3.1 Actors

Table 5.8.2.3.1-1 describes the Use Case actors and roles. It is assumed that NMT1 and NMT2 have no business relationship and their network services are expected to be isolated. In addition, levels of isolation are specified in the affinity/anti-affinity groups.

Table 5.8.2.3.1-1: Use case #7, actors and roles

#	Actor	Description
1	NMT1	OSS or other management system of service provider 1. NMT1 expects isolation from NMT2.
2	NMT2	OSS or other management system of service provider 2. NMT2 expects isolation from NMT1.
3	NFVO	NFV Orchestrator for the NS instances involved.
4	VNFM	VNF Manager for the VNFs involved.
5	VIM	VIM managing the NFVI hosting all resources involved.
6	VNFs	VNFs of the NS.

5.8.2.3.2 Pre-Conditions

Table 5.8.2.3.2-1 describes the pre-conditions.

Table 5.8.2.3.2-1: Use case #7, pre-conditions

#	Pre-condition	Description
1	NFV-MANO (VIM, NFVO and VNFM) is running.	
2	NMT1 and NMT2 have established their business relationship with the provider(s) of the NFV environment allowing them to deploy their services.	This includes that NFV-MANO is aware of the expected isolation of NSs and their constituents.
3	NMT1 and NMT2 have prepared the NFV packages and templates (e.g. NSD, VNFD) for the onboarding.	
4	NS1 of NMT1 and NS2 of NMT2 reference anti-affinity groups in the NSDs and VNFDs as appropriate.	The anti-affinity groups express the isolation constraints including levels between the NSs, VNFs and VNFCs.

5.8.2.3.3 Description

Table 5.8.2.3.3-1 describes the flow for onboarding an NS for NMT1. There is no difference with the standard flow and to the Use Case #2. The description only highlights some aspects related to tenants and anti-affinity groups, which is mainly during steps 2 and 6.

Table 5.8.2.3.3-1: Use case #7, base flow for onboarding

#	Flow	Description
1	NMT1 -> NFVO	NMT1 requests NFVO to onboard the NS, providing the NSD.
2	NFVO	NFVO executes the onboarding and registers NMT1 for this NS, see note 1, note 2 and note 5.
3	NFVO -> NMT1	NFVO acknowledges the onboarding
4	NMT1 -> NFVO	NMT1 subscribes for notifications. See note 4.
5	NMT1 -> NFVO	NMT1 requests NFVO to onboard the VNFs, providing the VNF packages VNFDs.
6	NFVO	NFVO executes the onboarding and registers NMT1 for this VNF package. See note 1 and note 3.
7	NFVO -> NMT1	NFVO acknowledges the onboarding.
NOTE 1: NFVO registers the NMT(s) to be able to protect a package, an NS or VNF, against operations from a different user (i.e. management isolation). It is recommended that different levels of permission (e.g. use versus making changes such as scale, update, delete) can be specified. NOTE 2: NSs could be shared between NMTs, but this is not covered in this Use Case. See Use Case #9 in clause 5.10. For the case of multiple NMTs for an NS, it is recommended that different levels of permission (e.g. use, scale, update, delete) can be specified. NOTE 3: VNFs could be shared between NSs of different NMTs, but this is not covered in this Use Case. See separate Use Case in clause 5.10. For the case of multiple NMTs for a VNF, it is recommended that different levels of permission (e.g. use, scale, update, delete) can be specified. NOTE 4: Subscription can also be done earlier. NOTE 5: The NSD and VNFD can be used to specify the anti-affinity groups, which also specify the appropriate levels of isolation.		

Table 5.8.2.3.3-2 describes the flow for instantiating an NS for NMT1. There is no difference with the standard flow and to the Use Case #2. The description only highlights some aspects related to tenants and anti-affinity groups, which is mainly during steps 2, 7, 9 and 10.

Table 5.8.2.3.3-2: Use case #7, base flow for instantiating

#	Flow	Description
1	NMT1 -> NFVO	NMT1 requests instantiation of the NS.
2	NFVO	NFVO validates the requests. This includes that NFVO checks whether NMT1 is allowed to instantiate the NS. NFVO checks that all VNFs are onboarded and are allowed to be instantiated by NMT1. NFVO checks resource availability for the VNF instantiation. See note 1, note 2 and note 6.
3	NFVO -> NMT1	NFVO acknowledges the NS instantiation request.
4	NMT1 -> NFVO	NMT1 subscribes for the relevant notifications. See note 3.
5	NFVO	NFVO validates the requests. This includes that NFVO checks whether NMT1 is allowed to subscribe to these notifications.
6	NFVO -> NMT1	NFVO acknowledges the subscriptions.
7	NFVO -> VNFM	NFVO requests instantiation of the VNFs as defined for the NS according to the deployment flavour indicating resource groups appropriate for NMT1. See note 1 and note 4.
8	VNFM	VNFM validates the request. This includes package validation.
9	VNFM -> VIM	VNFM requests the resources for the VNFs indicating resource groups appropriate for NMT1. See note 5 and note 6.
10	VIM	VIM allocates the resources keeping track of the resource groups.
11	VIM -> VNFM	VIM provides VNFM with the resources for the VNFs for NMT1.
12	VNFM -> VNF	VNFM finalizes the instantiation which can include configuration and VNF specific operations.
13	VNFM -> NFVO	VNFM acknowledges the VNF instantiation.
14	NFVO -> NMT1	NFVO acknowledges the NS instantiation.
NOTE 1: This step is simplified to avoid two flows for direct or indirect mode. Also allocation of other NS resources and the granting dialogue are not shown. NOTE 2: Resource availability includes availability within resource limits for NMT1. NOTE 3: Subscription can also be done earlier. NOTE 4: This Use Case does not cover nested NSs which will be covered in separate Use Case. NOTE 5: VIM distinguishes NMT1 and NMT2, so the NFVI is able to provide resource isolation, e.g. network isolation. NOTE 6: Resource availability includes availability considering the anti-affinity groups as specified in the NSD and VNFD. The expected levels of isolation are declared in the anti-affinity groups.		

Onboarding and instantiation for NMT2 are similar.

5.8.2.3.4 Post-Conditions

Table 5.8.2.3.4-1 describes the post-conditions.

Table 5.8.2.3.4-1: Use case #7, post-conditions

#	Post-condition	Description
1	VNFs and NSs are correctly instantiated.	
2	Resources are allocated as per NMT information.	
3	NFV-MANO FBs have all NMT information to provide management isolation.	This includes subscription to notifications.
4	NFVI has all information to isolate resources between NMTs.	This includes the levels of isolation and the appropriate information for network isolation.

5.8.3 Variants

In the above Use Case, the NFV environment is setup in the same way as in Use Case #2, clause 5.3, where both NMTs instantiate their NSs with the same NFVO. That means, both NMTs use the same NFV-MANO stack. In this case, NFVO and VNFM have knowledge of all the used anti-affinity groups and can make sure that proper group identifiers are used when resource groups of the VIM are created (see step 9 of the flow in Table 5.8.2.3.3-2 above).

This situation changes if the NMTs use different NFV-MANO entities, e.g. as it is done in Use Case #1, clause 5.2.

As shown in Figure 5.2.1-1, in this variant both NMTs deploy NSs and VNFs in the same NFVI-PoP. Anti-affinity can be specified by using anti-affinity groups on the interface between VNFM1 and VIM in the same way as between VNFM2 and VIM. This can only be done if both VNF Managers use the same resource group ids if an anti-affinity is specified.

Thus this variant leads to the issue that the same anti-affinity groups are to be used by multiple VNFMs or NFVOs, serving different NMTs. The use of the same resource group ids for different VNFMs would create a security issue, in case the VNFM managed resources are intended to be assigned to different tenants.

5.8.4 Analysis

It is demonstrated that anti-affinity groups can specify isolation constraints, and can be used together with local affinity rules to define where no isolation is expected or it is even preferred to run together. Also, anti-affinity groups are well suited to specify the different levels of isolation.

However, the use of anti-affinity groups for specifying isolation constraints between different NMTs, has some drawbacks. It can only be done if the group ids of the anti-affinity groups are shared between different VNFMs. Also with higher numbers of NMTs it can be very complex.

Key issue #1, solution #1.1 in clause 6.1.2 illustrates the use of anti-affinity groups, while key issue #1, solution #1.3 in clause 6.1.4 illustrates the use of infrastructure resource groups to specify the isolation between NSs of different NMTs. Key issues #2 in clause 6.2 discusses the use of tenant identifications and tenant management.

Summarizing, affinity/anti-affinity groups provide a means to define the isolation, however, it leads to complex definitions using composite NSs. Therefore, in key issue #2, other solutions are discussed.

5.9 Use Case #8: Isolation of containerized VNF instances

5.9.1 Motivation

OS Containers introduce other levels of isolation. Multiple workloads from different VNFs or NSs can join not only the same hardware, but also the same Operating System kernel.

Figure 5.9.1-1 illustrates the additional isolation needs of containerized workloads:

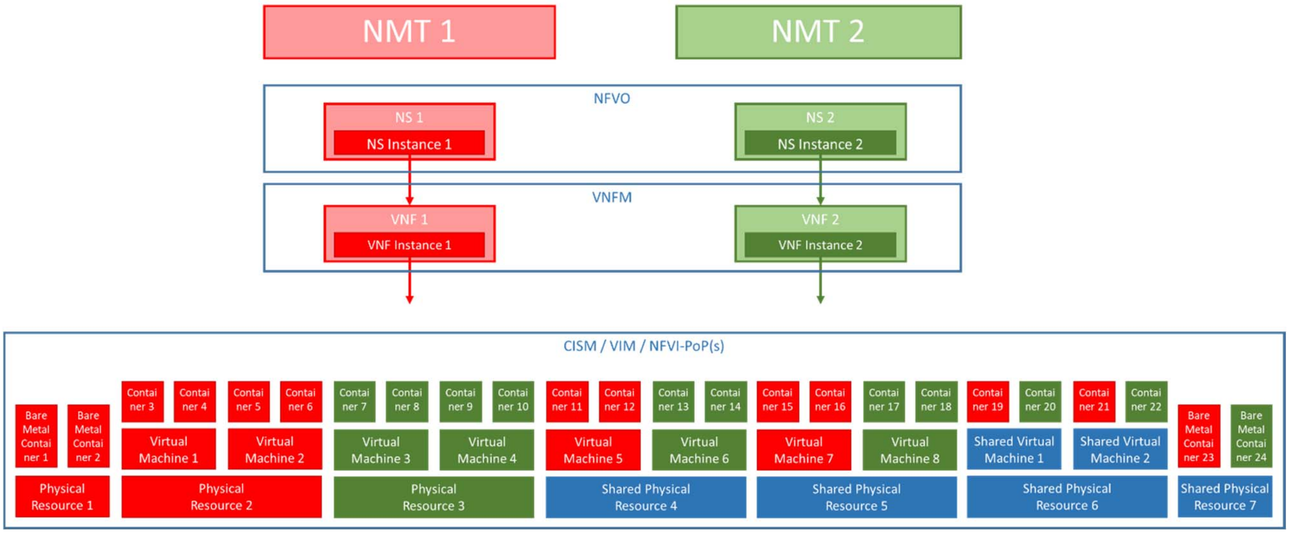


Figure 5.9.1-1: Isolation of containerized workloads on different levels

In addition to the levels A-D described in Use Case #7 (clause 5.8) containerized workloads can share the same virtual machine and in that case their isolation depends on the capabilities of the container technology only.

Container management technologies provide mainly two mechanisms to manage the different isolation expectations of the containerized workloads:

- **Namespaces** are used to provide isolation support of the Operating System and container infrastructure service (CIS), as described in ETSI GS NFV-IFA 040 [i.19]. As described in ETSI GS NFV-IFA 040 [i.19], containerized workloads are modelled via MCIOs and MCIOPs. As described in ETSI GS NFV-IFA 011 [i.11], the MCIOP profile can specify affinity or anti-affinity on the level of MCIOPs, indicating whether or not containerized workloads deployed based on the MCIOPs can share the same namespace. Isolation by using different namespaces is sometimes called soft isolation.

NOTE 1: The Container Infrastructure Service Management (CISM) manages containerized workloads across multiple VMs or physical servers by managing affinity/anti-affinity on the level of nodes (CIS instances) and namespaces.

NOTE 2: Container technologies can provide different capabilities to isolate workloads via namespaces, i.e. within a CIS cluster. As an example see kata containers [i.28].

- **Clusters** can be used to provide isolation between groups of resources and thus can provide stronger isolation. Containerized workloads in a cluster can share resources, while different clusters mean different resources. The management for CIS clusters is further described in ETSI GS NFV-IFA 036 [i.18].

Thus for the case of containerized workloads, additional levels of isolation are possible:

a) Containerized workloads that can share resources freely are deployed in the same namespace.

b) Containerized workloads with weak isolation constraints can use Operating System level protection and are deployed in different namespaces in the same CIS cluster.

NOTE 3: See examples in Figure 5.9.1-1, containers 19&20 or 21&22. For the case of containers on bare metal, see containers 23&24. Operating System level isolation is not shown in Figure 5.9.1-1. See Figure 5.9.1-2 for the illustration of clusters and namespaces.

c) Containerized workloads with strong isolation constraints are deployed in different CIS clusters and the levels of isolation as in Use Case #7 can be used to isolate between the clusters.

Figure 5.9.1-2, copied from ETSI GS NFV-IFA 036 [i.18], illustrates the usage of namespaces and clusters for the deployment of containerized VNFs:

- In the cases where VNF instances share the same namespace (e.g. namespace 3), there is no isolation.

- In the cases where VNF instances are deployed in different namespaces, but the same cluster (e.g. namespace 1 and 2 are in CIS cluster 1, namespace 4 and 5 are in cluster 3), the isolation is provided by the operating system and the container runtime environment (i.e. by the CIS). Additional isolation and security can be achieved as described in ETSI GS NFV-SEC 023 [i.21].

- In the cases where VNF instances are deployed in different clusters, isolation is achieved by different resource usage, in the same way as in the Use Cases #1, #2, #4, etc.

NOTE 4: The CISM instances manage workloads in a CIS cluster and the namespaces, see ETSI GS NFV-IFA 040 [i.19]. Containerized workloads can be deployed in VMs or physical servers, see ETSI GS NFV-IFA 036 [i.18].

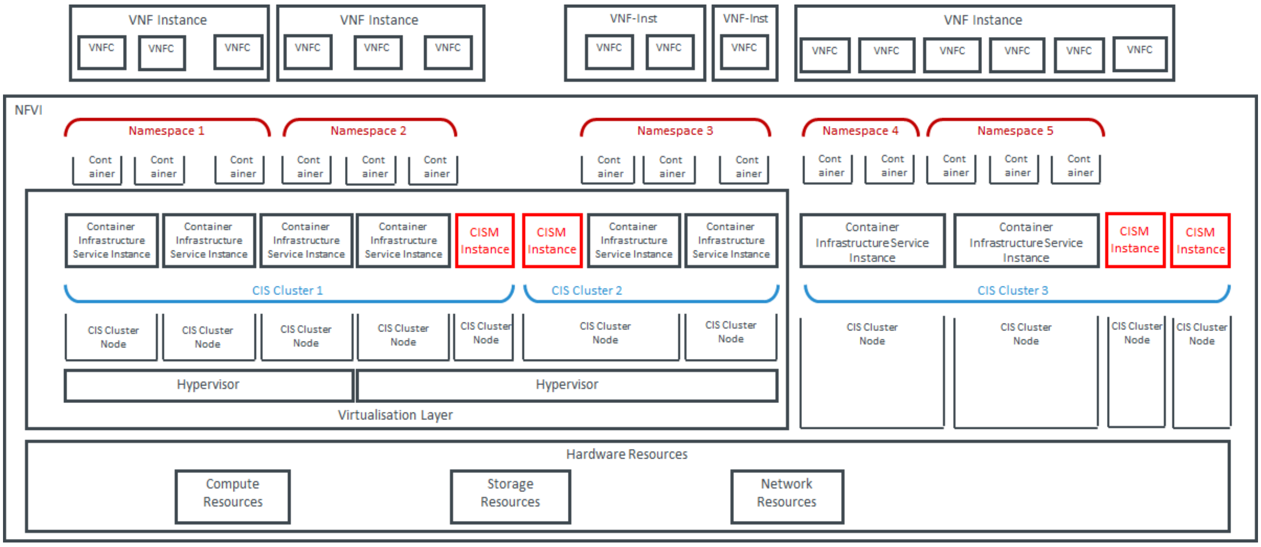


Figure 5.9.1-2 (copy from ETSI GS NFV-IFA 036 [i.18], Figure 4.4.1-1): Deployment example with VNFs, CIS clusters and namespaces

5.9.2 Detailed User Story

5.9.2.1 Summary

This user story extends the description of Use Case #7 in clause 5.9.2 by including the aspect of containerization.

For containerized VNFs, affinity and anti-affinity can be defined also for each MCIOP as specified in ETSI GS NFV-IFA 011 [i.11]. Also container namespace can be used as an additional level for the affinity or anti-affinity constraints.

CISM provides flexible dynamic placement of containers and therefore needs to be aware of the affinity and anti-affinity levels that can be applied within its scope of responsibility, that is within its CIS cluster. CISM can thus apply affinity and anti-affinity on the level of namespaces or CIS cluster nodes. Affinity and anti-affinity constraints for CIS cluster level or NFVI related levels need to be applied when the NFVO selects the CIS cluster for a VNF. At the time of cluster selection, also affinity and anti-affinity constraints of the CIS cluster nodes are considered. Isolation needs can be defined using affinity and anti-affinity on level of VMs or physical servers.

5.9.2.2 Actors

Table 5.9.2.2-1 describes the Use Case actors and roles. It is assumed that NMT1 and NMT2 have no business relationship and their network services are expected to be isolated. In addition, levels of isolation are specified in the affinity/anti-affinity groups.

Table 5.9.2.2-1: Use case #8, actors and roles

#	Actor	Description
1	NMT1	OSS or other management system of service provider 1. NMT1 expects isolation from NMT2.
2	NMT2	OSS or other management system of service provider 2. NMT2 expects isolation from NMT1.
3	NFVO	NFV Orchestrator for the NS instances involved.
4	VNFM	VNF Manager for the VNFs involved.
5	VIM	VIM managing the NFVI hosting all resources involved.
6	VNFs	VNFs of the NS.
7	CISM	Container Infrastructure Service Management for the containerized VNFs involved.

5.9.2.3 Pre-Conditions

Table 5.9.2.3-1 describes the pre-conditions.

Table 5.9.2.3-1: Use case #8, pre-conditions

#	Pre-condition	Description
1	NFV-MANO (VIM, NFVO and VNFM) is running.	
2	NMT1 and NMT2 have established their business relationship with the provider(s) of the NFV environment allowing them to deploy their services.	This includes that NFV-MANO is aware of the expected isolation of NSs and their constituents.
3	NMT1 and NMT2 have prepared the NFV packages and templates (e.g. NSD, VNFD) for the onboarding.	
4	NS1 of NMT1 and NS2 of NMT2 reference anti-affinity groups in the NSDs, VNFDs and MCIOPs as appropriate.	The anti-affinity groups express the isolation constraints including levels between the NSs, VNFs, VNFCs and MCIOPs.
5	CIS cluster(s) as needed for the placement constraints specified in the NSD and VNFD are available.	ETSI GS NFV-IFA 036 [i.18] illustrates in a Use Case that the creation of additional CIS clusters can be triggered during NS instantiation.

5.9.2.4 Description

The onboarding is the same as in Table 5.9.2.4-1.

Table 5.9.2.4-1 describes the flow for instantiating an NS with containerized VNF(s) for NMT1. There is no difference to the standard flow and to the Use Case #2. The flow adds to the flow in Table 5.8.2.3.3-2 the necessary steps related to containerized VNFs.

Table 5.9.2.4-1: Use case #8, Base flow for instantiating

#	Flow	Description
1	NMT1 -> NFVO	NMT1 requests instantiation of the NS.
2	NFVO	NFVO validates the requests. This includes that NFVO checks whether NMT1 is allowed to instantiate the NS. NFVO checks that all VNFs are onboarded and are allowed to be instantiated by NMT1. NFVO checks resource availability for the VNF instantiation. See note 1, note 2 and note 6.
3	NFVO -> NMT1	NFVO acknowledges the NS instantiation request.
4	NMT1 -> NFVO	NMT1 subscribes for the relevant notifications. See note 3.
5	NFVO	NFVO validates the requests. This includes that NFVO checks whether NMT1 is allowed to subscribe to these notifications.
6	NFVO -> NMT1	NFVO acknowledges the subscriptions.
7	NFVO -> VNFM	NFVO requests instantiation of the VNFs as defined for the NS according to the deployment flavour indicating resource groups appropriate for NMT1. See note 1 and note 4.
8	VNFM	VNFM validates the request. This includes package validation.
9	VNFM -> NFVO	VNFM requests the NFVO to grant the resources for the new VNF instance(s)
10	NFVO	NFVO analyses the resources for the new VNF instance and the given placement constraints, e.g. anti-affinity.
11	NFVO	NFVO selects CIS cluster(s) and namespace(s) for the deployment of the containerized workloads of the VNF according to the placement constraints in the NSD and VNFD and appropriate for NMT1. See note 7 and note 9.
12	NFVO -> VNFM	NFVO confirms the granted resources.
13	VNFM -> VIM	VNFM requests the virtualized resources for the VNFs indicating resource groups appropriate for NMT1. See note 5, note 6 and note 8.
14	VIM	VIM allocates the resources keeping track of the resource groups.
15	VIM -> VNFM	VIM provides VNFM with the virtualized resources for the VNFs for NMT1.
16	VNFM -> CISM	VNFM requests the CISM responsible for the relevant CIS cluster(s) to install the MCIOP(s) referred to in the VNFD indicating namespaces and other constraints appropriate for NMT1, e.g. affinity and anti-affinity.
17	CISM -> VNFM	CISM acknowledges the instantiation of the MCIOP(s).
18	VNFM -> VNF	VNFM finalizes the instantiation which can include configuration and VNF specific operations.
19	VNFM -> NFVO	VNFM acknowledges the VNF instantiation.
20	NFVO -> NMT1	NFVO acknowledges the NS instantiation.
NOTE 1: This step is simplified to avoid two flows for direct or indirect mode. NOTE 2: Resource availability includes availability within resource limits for NMT1. NOTE 3: Subscription can also be done earlier. NOTE 4: This Use Case does not cover nested NSs which will be covered in separate Use Case. NOTE 5: VIM distinguishes NMT1 and NMT2, so the NFVI is able to provide resource isolation, e.g. network isolation. NOTE 6: Resource availability includes availability considering the anti-affinity groups as specified in the NSD and VNFD. The expected levels of isolation are declared in the anti-affinity groups. NOTE 7: Affinity and anti-affinity constraints defined in the MCIOP are not relevant for the CIS cluster selection, but only relevant for the scheduling of the CISM. NOTE 8: This steps refers to virtualized resources e.g. for VM based parts. NOTE 9: It is assumed in this Use Case that appropriate CIS cluster(s) are available. Other cases are illustrated in ETSI GS NFV-IFA 036 [i.18].		

Instantiation for NMT2 is similar.

5.9.2.5 Post-Conditions

Table 5.9.2.5-1 describes the post-conditions.

Table 5.9.2.5-1: Use case #8, post-conditions

#	Post-condition	Description
1	VNFs and NSs are correctly instantiated.	
2	Resources are allocated as per NMT information.	
3	NFV-MANO FBs have all NMT information to provide management isolation.	This includes subscription to notifications.
4	NFVI has all information to isolate resources between NMTs.	This includes the levels of isolation and the appropriate information for network isolation.
5	The deployment of the containerized workloads of the NSs and VNFs to CIS clusters is done according to the isolation needs of the NMTs.	
6	The configuration of the CISM namespaces and placement constraints and the deployment of the MCIOPs is done according to the isolation needs of the NMTs.	

5.9.3 Variants

The variants described in Use Case #7, clause 5.8.3 are possible also for containerized VNFs.

In addition, CIS clusters can be instantiated specifically for NMTs. NMTs can directly consume CCM APIs or dedicated CIS clusters for the NMTs can be allocated via the NFVO consuming CCM APIs during NS LCM. See Figure 5.9.3-1.

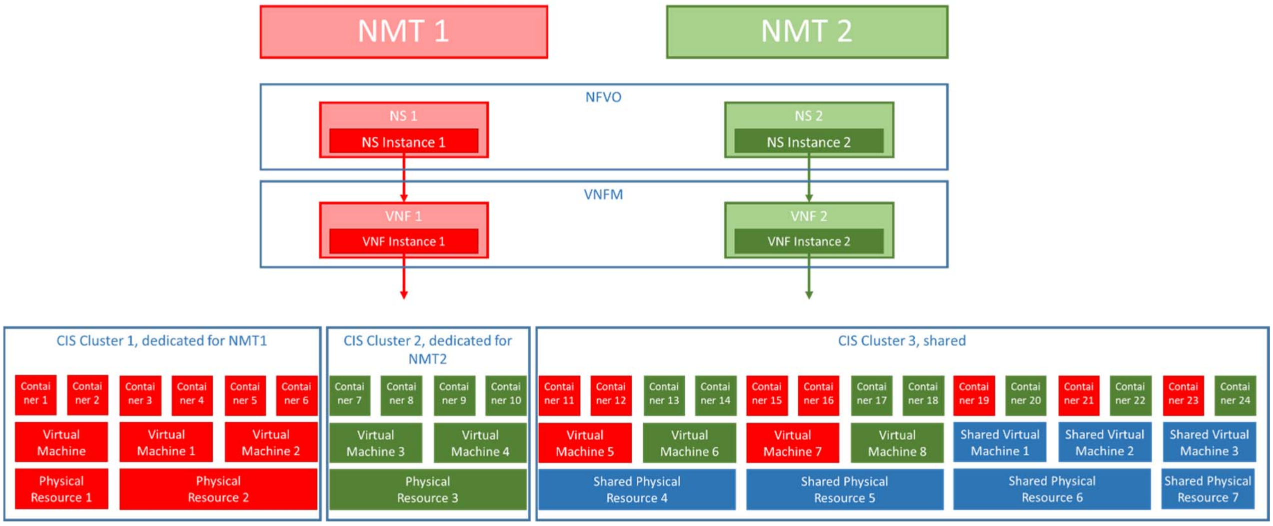


Figure 5.9.3-1: Relations of NMTs and CIS clusters

In this example, CIS cluster 1 is dedicated for NMT 1, providing isolation at the CIS cluster node level to CIS cluster 2 dedicated for NMT 2. Within CIS cluster 3, workloads of NMT 1 can be isolated from workloads of NMT 2 using namespaces.

5.9.4 Analysis

It is illustrated that in addition to Use Case #7, clause 5.8, the NFV-MANO is responsible for selecting appropriate CIS clusters and namespaces for the containerized workloads according to the isolation needs of the NMTs.

It is therefore recommended to provide requirements of the NFVO and VNFM to support CIS cluster and namespace selection according to multi-tenancy during NS and VNF LCM operations. See recommendation Mtenant.tenantmgmt.07 in clause 6.2.7.

The CISM can be unaware of the NMTs, since the isolation needs can be expressed by the deployment to CIS clusters and namespaces. However, the CISM service APIs are covered in key issue #2, solution #2.2, clause 6.2.3, to support management isolation, that is disallow management of containerized workloads by the wrong tenant.

The CCM service API is directly exposed to consumers in the same way as to the NFVO. Therefore, recommendations for CCM to support multi-tenancy in the same way as NFVO are provided in clause 6.2.7. See especially recommendation Mtenant.tenantauth.03.

A special case arises if NMT1 creates a cluster, and NMT2 wants to deploy containerized workloads on that cluster. This can be supported by appropriate permissions.

Further analysis and security recommendations are presented in ETSI GS NFV-SEC 023 [i.21].

5.10 Use Case #9: Multiple NMTs use the same entity

5.10.1 Introduction

This Use Case shows multiple NMTs using the same entity.

Depending on the type of the shared entities two different cases are possible:

a) NMTs can share VNF packages and descriptors, but each NMT creates own instances.
In this case, no sharing of compute, storage or network resources is happening. Ownership and usage rights for packages and descriptors can be defined during onboarding. During instantiation, the tenancy can be controlled. In this case isolation of the compute, storage or network resources can be done as in Use Cases without shared packages or descriptors. See clause 5.10.2 for more details.

b) In the case that NMTs share instances, there will be compute, storage or network resources that are commonly used by both (or multiple) tenants. The shared infrastructure resources are not aware of different tenants. The shared instantiated entities can provide their own tenant control. A well-known example is a database, that can be instantiated as a VNF or NS, and can be used by multiple tenants, while it provides its own mechanisms for data isolation and access control. Clause 5.10.3 illustrates various cases of sharing VNFs or NSs.

While case A can be a common case (e.g. there could be a common package repository), case B can easily lead to situations, where security cannot be guaranteed. Therefore, restrictions for the sharing can be recommended.

Finally a special case will be presented about shared storage, see clause 5.10.4.

Management aspects of sharing:

Typically, a shared package or NSD can be onboarded by one of the tenants, and then shared to others; a shared VNF or nested NS can be instantiated by one of the tenants, and then added to another user's NS via update NS operation. This shows that one tenant has full management rights for the shared entity while additional tenants might have reduced permissions to use or manage the shared instance.

This leads to the following questions:

- Will an "owner" be defined for an entity? E.g. the NMT who onboards a VNF package would be its natural owner and as a default have full permissions for all LCM operations.
- Will it be possible that the "owner" of a VNF instance is different from the "owner" of the VNF package? E.g. the NMT who onboards a VNF package would allow another tenant to create a VNF instance from a package and have full permissions for all LCM operations on the instance.
- Can ownership (both of VNF instances or VNF packages) be transferred to another NMT?
- Will the "owner" have the possibility to add NMT's usage rights for a VNF instance dynamically?
- Following permissions for a VNF could be defined and can be different for each tenant:
    - Invoke LCM operations on an VNF or VNF instance.
    - Invoke package management operations on the VNF package.
    - Receive LCM notifications related to a VNF instance.
    - Receive FCAPS notifications related to a VNF instance.

- Manage the permissions of a VNF instance.
- Manage the isolation of a VNF.

The above questions are also applicable on the NS level.

Management of tenants, permissions, privileges and isolation needs is discussed further in key issue #2, clause 6.3.

5.10.2 Sharing of VNF packages and descriptors

5.10.2.1 Motivation

In the following cases sharing of VNF packages and descriptors is considered. In these cases, there is no sharing of NS or VNF instances between the tenants, but the tenants instantiate the same VNFs or even NSs. In these cases, the tenants instantiate from the same packages. The packages can come from a common repository.

In the case that each tenant onboards his own copy of the NSD or VNF package to the NFVO, there is no sharing visible to the NFVO. However, it is necessary to verify that an NSD or a VNF package can be onboarded separately by the tenants. There could be issues associated with the uniqueness of identifiers, e.g. the vnfdId is managed by the VNF provider and identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [i.11], clause 7.1.2.2. See solution proposals #3.1 and #3.2 in clause 6.3.

In the case that an NSD or a VNF package is onboarded once and then instantiated by multiple tenants, the tenants will share the package identifiers and provide permissions to use an onboarded package. The detailed user story in clause 5.10.2.2 describes this case.

NOTE 1: Sharing of the information about NSDs/VNFs in a repository between the tenants is outside the scope of the present document.

NOTE 2: The figures in this clause showing "Virtual Resources" refer to different virtualization technologies, e.g. VMs and containers.

Figure 5.10.2.1-1 illustrates two NMTs sharing an NS. In this case, usually VNFs, as constituents of these NSs, are also shared.

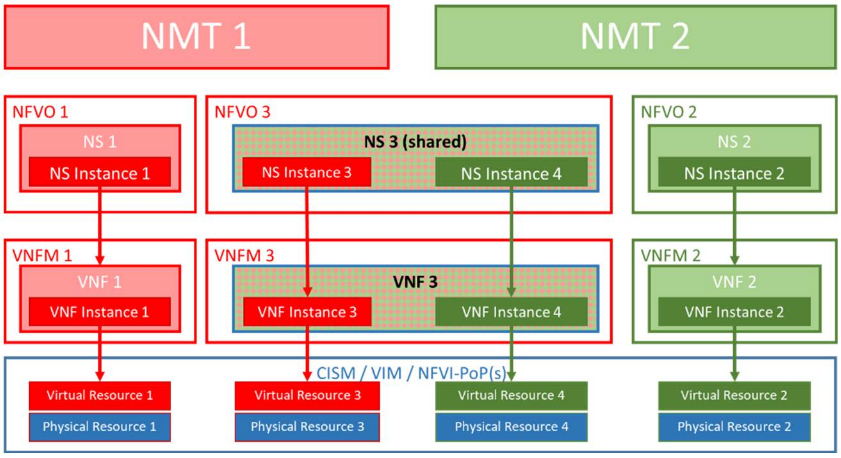


Figure 5.10.2.1-1: Two NMTs use separate NS instances from the same NSD

Figure 5.10.2.1-2 illustrates two NMTs sharing VNFs, but not NSs. In this figure, both NMTs use the same NFVO. In case they would use different NFVOs, access to the VNF package is a prerequisite for both NFVOs.

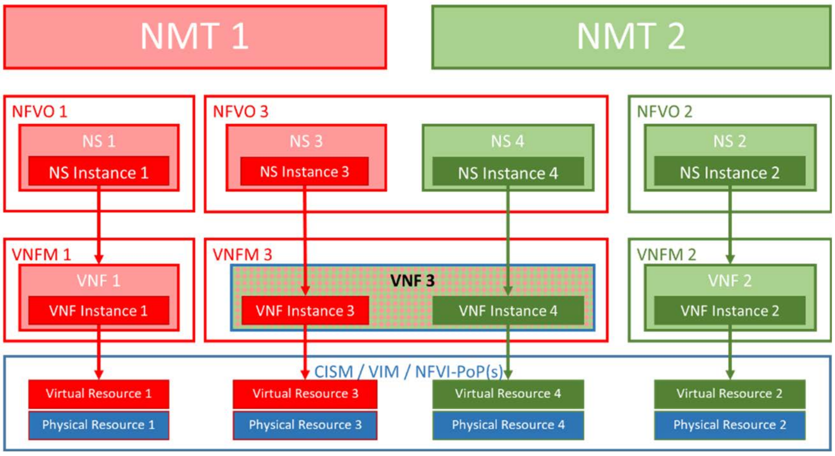


Figure 5.10.2.1-2: Two NMTs use separate NS instances from the same NSD

Figure 5.10.2.1-3 illustrates the use of a shared nested NS with separate NS instances for each NMT.

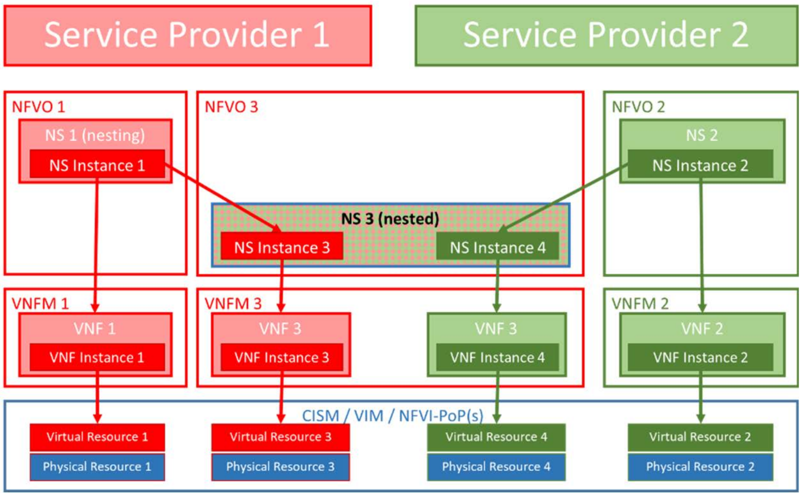


Figure 5.10.2.1-3: Different service providers use same nested NS but different instances

In all these cases, the VNF packages are signed by the vendor, and their usage is protected by permissions. Additional protection by signing packages by their owner (who has bought the package) and agreements about licensing will be introduced.

According to ETSI GS NFV-IFA 011 [i.11], clause 6.2.4, ETSI GS NFV-SOL 004 [i.35], clause 5 and ETSI GS NFV-SEC 021 [i.36], VNF packages are protected in several ways using certificates and signatures. Service providers can optionally use private signatures, which would prohibit that a VNF package can be shared between service providers. This mechanism is optional, therefore the sharing of VNF packages is possible, where service provider policies allow it.

NOTE 3: Instances of shared VNFs will all be based on the same VNFD, however, instantiation parameters can be different. Care is expected in designing application software to be made aware of multi-tenancy. Impacts on the application software to support being shared between tenants are not analysed in the present document.

In the case of containerized VNFs, it is a typical case to store software images in a central repository. ETSI GS NFV-IFA 040 [i.19] defines the CIR. While a common repository for NSDs and VNF packages is not defined currently in the scope of NFV-MANO, CIR services are defined in NFV. Similar capabilities for the protection of using packages from the repositories by multiple tenants are recommended. See solution proposal #3.5 in clause 6.3.6 for a central repository for VNF packages or NSD file artifacts and solution proposal #3.7 in clause 6.3.8 for multi-tenancy aspects of the CIR.

5.10.2.2 Detailed User Story

5.10.2.2.1 Summary

The following flow illustrates the interactions based on the scenario in Figure 5.10.2.1-1. The NSD and VNF are onboarded once and used for instantiation for NMT1 and NMT2. The other scenarios use similar mechanisms on different levels.

In this Use Case, the flows look different for the NMT1 and NMT2, while the onboarding and instantiation details within NFV-MANO (e.g. interworking of NFVO and VNFM) are the same as in Use Case #2, clause 5.3.

5.10.2.2.2 Actor(s)

Table 5.10.2.2.2-1 describes the Use Case actors and roles.

Table 5.10.2.2.2-1: Use case #9, actors and roles

#	Actor	Description
1	NMT1	OSS or other management system of service provider 1.
2	NMT2	OSS or other management system of service provider 2.
3	NFVO3	NFV Orchestrator for the shared NS instances involved.
4	VNFM3	VNF Manager for the VNFs involved.
5	VIM	VIM managing the NFVI hosting all resources involved.
6	NS3	The shared NS3, refer to Figure 5.10.2.1-1.
7	VNF3	The shared VNF3, refer to Figure 5.10.2.1-1.

5.10.2.2.3 Pre-Conditions

Table 5.10.2.2.3-1 describes the pre-conditions.

Table 5.10.2.2.3-1: Use case #9, pre-conditions

#	Pre-condition	Description
1	NFV-MANO (VIM, NFVO and VNFM) is running.	
2	NMT1 and NMT2 have established their business relationship with the provider(s) of the NFV environment allowing them to deploy their services.	This includes that NFV-MANO is aware of the expected isolation of NSs and their constituents, including the option to share NSs and VNFs.
3	NMT1 and NMT2 have a business agreement on sharing the VNFs.	
4	NMT1 and NMT2 have established a mechanism to share information about the entities they want to share.	These mechanism are outside the scope of the present document.
5	Permissions to use the shared network service are set accordingly.	

5.10.2.2.4 Description

Table 5.10.2.2.4-1 describes the flow for onboarding an NS for NMT1. There is no difference with the standard flow. The description only highlights some aspects related to tenants, which is mainly during steps 2 and 6.

Table 5.10.2.2.4-1: Use case #9, base flow for onboarding a shared NS

#	Flow	Description
1	NMT1 -> NFVO3	NMT1 requests NFVO to onboard the NS3, providing the NSD.
2	NFVO3	NFVO3 executes the onboarding and registers NMT1 for this NS, see note 1.
3	NFVO3 -> NMT1	NFVO acknowledges the onboarding
4	NMT1 -> NFVO3	NMT1 subscribes for notifications. See note 2.
5	NMT1 -> NFVO3	NMT1 requests NFVO to onboard the VNFs, providing the VNF packages VNFDs.
6	NFVO3	NFVO3 executes the onboarding and registers NMT1 for this VNF package. See note1.
7	NFVO3 -> NMT1	NFVO acknowledges the onboarding
8	NMT1 <-> NMT2	NMT1 shares the identifiers for the NS (nsdId, see ETSI GS NFV-IFA 013 [i.13]) with the NMT2, see note 3.
9	NMT2 -> NFVO3	NMT2 subscribes for notifications. See note 2.
NOTE 1: NFVO registers the NMT1 to be able to protect the NSD and VNF package against operations from an unauthorized user (i.e. management isolation). It is recommended that different levels of permission (e.g. use versus making changes such as scale, update, delete) can be specified. NOTE 2: Subscription can also be done earlier. NOTE 3: Depending on the permissions granted to NMT2, available NS profiles, etc. more information can be shared. The mechanisms for this step are outside the scope of the present document.		

The flow for instantiating the NS is identical to the flow in Use Case #2, Table 5.3.2.4-2, for both, NMT1 and NMT2. Based on step 8 in Table 5.10.2.2.4-1, NMT2 has all information for the instantiation, and based on step 9, it has the privilege to execute it.

5.10.2.2.5 Post-Conditions

Table 5.3.2.5-1 describes the post-conditions.

Table 5.3.2.5-1: Use case #9, post-conditions

#	Post-condition	Description
1	VNFs and NSs are correctly instantiated.	
2	Resources are allocated as per NMT information.	
3	NFV-MANO FBs have all NMT information to provide management isolation, as well as to allow operations by the correct tenants.	This includes that only the "own" instances can be managed and queried, and also the appropriate subscription to notifications. See key issue #2. Solution proposal #2.4 addresses this point.
4	NFVI has all information to isolate or share resources between NMTs.	This includes the appropriate information for network isolation.
5	Each NMT has all information to manage its own NS and VNF instances.	This includes all information for FCAPS according to the permissions.

5.10.3 Sharing of NS or VNF instances

5.10.3.1 Motivation

This Use Case shows multiple NMTs using the same entity. This can happen at several levels:

- NMTs use the same NS instance;
- NS instances (same or different NSD) of different NMTs share a VNF instance;
- NS instances (same or different NSD) of different NMTs share a nested NS; or
- NS/VNF instances of different NMTs share some PaaS Services offered by the NFVI platform.

These different variants of the Use Case share similar aspects about the handling of permissions of NMTs to manage (create, instantiate, scale, update, delete) NSs, VNFs or their instances as well as VNF packages.

In all these cases, some information (e.g. identifiers, configuration information, scaling levels or other instantiation parameters) is common between the tenants about the shared entities. How this information is shared between the tenants is outside the scope of the present document. Also, the tenants agree on the use of licenses, keys and certificates for the shared VNFs. This agreement is assumed to be done, but how it is achieved and documented is outside the scope of the present document.

Figure 5.10.3.1-1 illustrates two NMTs using the same NS instance.

Figure 5.10.3.1-2 illustrates two NMTs with separate NSs (i.e. different NSD), but referencing the same VNF (i.e. same VNF package, same VNFD). NS instance 4 re-uses the VNF instance 3 and shares it with NMT1.

Figure 5.10.3.1-3 shows a similar sharing of a VNF instance, but from two different NSs (i.e. different NSD).

Figure 5.10.3.1-4 shows the use of nested NSs for the sharing.

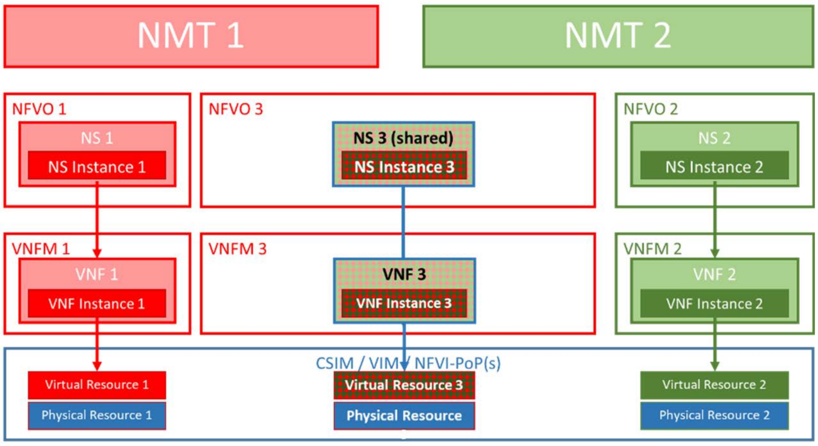


Figure 5.10.3.1-1: Two NMTs use the same NS instance

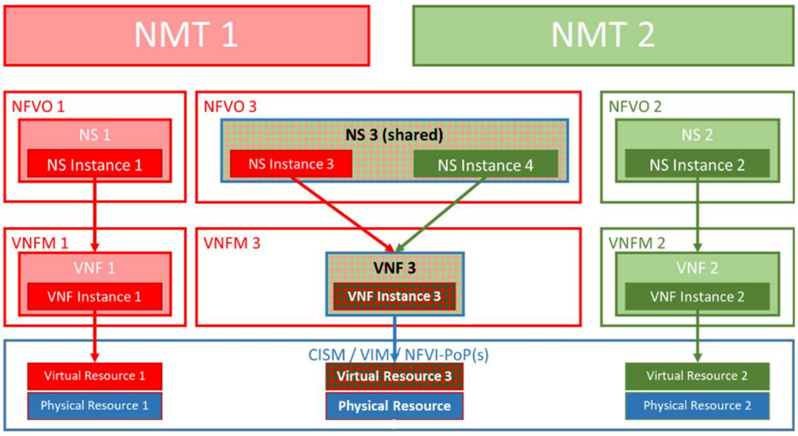


Figure 5.10.3.1-2: NS instances of different NMTs use same VNF

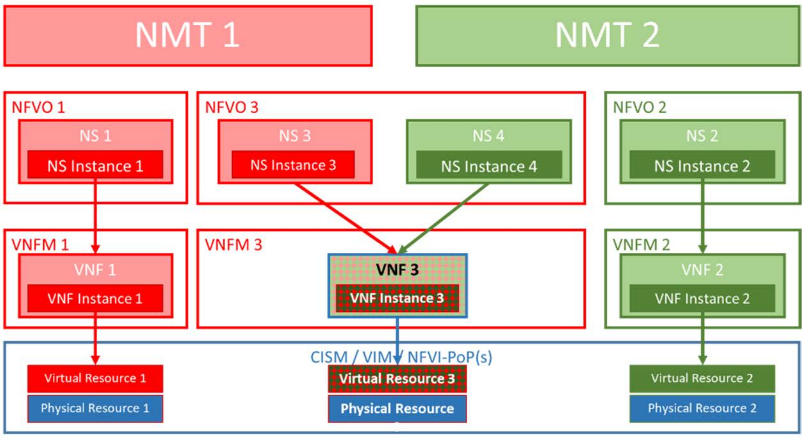


Figure 5.10.3.1-3: Different NSs use same VNF instance

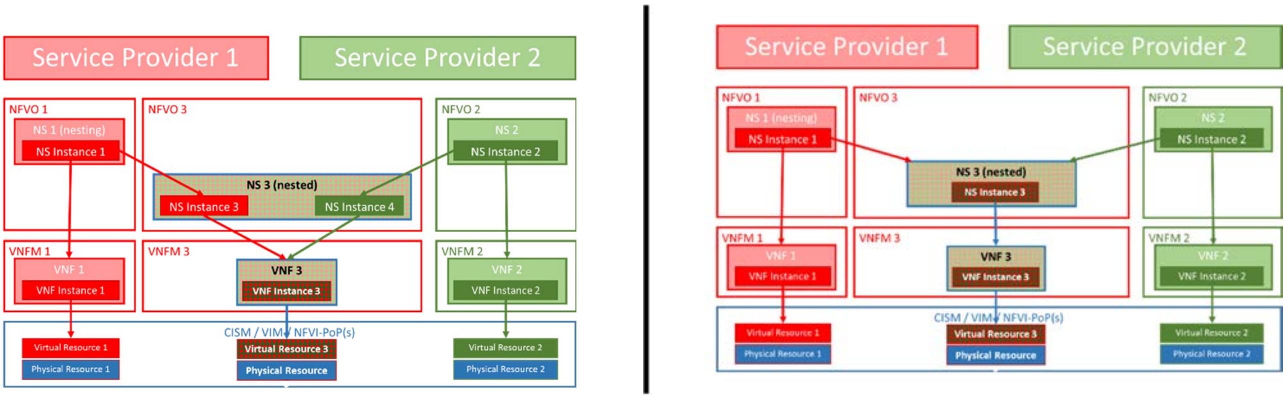


Figure 5.10.3.1-4: Different service providers use same nested NS and share VNF instances

In all these cases, NFV-MANO registers multiple NMTs for the entities they consume. It is recommended to allow to define different permissions about management (e.g. create, instantiate, delete, update), operation (e.g. enable, disable, scale) or usage (e.g. instantiate from a package, reference a VNF from NSD) of the different entities.

In some of the cases, there are virtual resources that are shared between NMT1 and NMT2 and also resources that are not shared. In these cases, isolation is also expected between virtual resources 1 and virtual resources 3.

The flow for instantiation in the Use Case is identical to the flow in Use Case #2, clause 5.3, or for the scenarios with nested NSs Use Case #4, clause 5.5.

5.10.3.2 Isolation aspects of shared entities

If an entity is a VNF instance and used by NMT1 and NMT2 (see Figures 5.10.3.1-1 and 5.10.3.1-4), there is an expectation of NMT1 to isolate it from other entities of NMT2, and vice versa. The expectations can only be fulfilled if the shared instance, whose resources are all shared, communicates with other VNFs of NMT1 (or NMT2) only via external interfaces. It is recommended to create a separate isolation group for the shared instance, so NFV-MANO can define separate isolation constraints for the shared group of entities. The shared group of entities can be configured then to allow usage by NMT1 and NMT2, but not by others.

NOTE 1: The isolation group is not to be configured explicitly, but just indicates the group of entities that are used by the same set of NMTs, and thus have common isolation constraints.

NOTE 2: The present document does not address the impacts on the application software of VNF instances to support being shared between tenants.

If the entity is an NS instance (see Figures 5.10.3.1-1 and 5.10.3.1-4, right side), the situation is similar. In this case, it can be created without using external interfaces within an NS.

In case the entity is a VNF or NS, and NMT1 and NMT2 use different instances each (see Figures 5.10.3.1-2, 5.10.3.1-3 and 5.10.3.1-4, left side), the situation is simpler. Different instances of the same VNF or NS do not share resources or interfaces. The only recommendation coming from this case is to structure the definition of tenant isolation in NFV-MANO in a way that provides isolation at instance level. This is then the same case as in Use Case #3, clause 5.4, network slicing. There slice instances are network service instances that are isolated.

5.10.4 Special case: Shared storage

While clauses 5.10.2 and 5.10.3 illustrate sharing VNFs, NSs or their instances, there are also Use Cases to share constituents of VNFs or NSs. The NFV principles do not foresee sharing of VNFCs, but the sharing of storage resources is an important case and therefore is discussed here.

ETSI GR NFV-IFA 037 [i.46] describes in clause 5.7.4 the gap of missing specification on how a VNF is expected to reuse shared storage. The same document defines in clause 6.4 recommendations 5gnfv.desc.004 and 5gnfv.desc.005 about sharing virtualised storage resources. However, in this context, multi-tenancy is not considered, but the focus is on general sharing aspects.

In a multi-tenancy sharing scenario, authorization and authentication mechanisms as illustrated in key issue #2 can control whether tenants can establish the use of shared resources. E.g. a tenant needs proper privilege to query the necessary information about the shared virtualized storage resource.

Additional access control during run-time is of course necessary, but depends strongly on the capabilities of the system providing the virtualized storage resource. This is out of scope of the present document. As indicated in the annex on multi-tenancy in Anuket project (clause A.3), Anuket Reference Model for Cloud Infrastructure (RM) [i.38] provides some related information.

See key issue #5 in clause 6.5, especially solution proposal #5.1.

NOTE: ETSI GR NFV-IFA 037 [i.46] does not mention sharing of network resource explicitly. Sharing of network resources is also not analysed in the present document.

5.10.5 Special case: Common Services

A special case to be considered under multi-tenancy is the sharing of PaaS Services. PaaS Services can be VNF Common Services, VNF Dedicated Services or other services, for example related to connectivity support for VLs. Clause 5.8 of ETSI GS NFV-IFA 010 [i.10] introduces references for the general concepts of PaaS Services.

In particular, the case of VNF Common Services is relevant to multi-tenancy in NFV. As defined in ETSI GS NFV-IFA 010 [i.10], a VNF Common Service is a modular service or a function with a lifecycle independent from its consumers and is consumable by either one or multiple services, such as VNF instances. Therefore, a PaaS Service can be shared among multiple VNF instances, and if the VNF instances are managed or assigned to different tenants, then the same PaaS Service instance might be used by multiple tenants.

An example of a VNF Common Service can be a Configuration Server as introduced in clause 7.3.3.1.2.1 of ETSI GR NFV-EVE 022 [i.48] and further specified in ETSI GS NFV-IFA 049 [i.49].

5.10.6 Analysis with regards to License management

Regarding the license management of the VNF in these several variants, even if the NS or VNF are shared, the license entitlement rights are expected to be managed independently. Each tenant can use the same licence management system but uses different licensing model and different license entitlement rights set.

In the variant 1 and variant 4 the two tenants use the same VNF instance. For these cases, there is no way to distinguish the VNF instance usage coming from the NMT1 or the NMT2. For these variants, only a specific agreement between the tenants on the use of the license entitlement rights can be used. The VNF-LM considers a unique license entitlement rights set for both tenants, as if just one tenant was concerned. The tenants could then share the charging of the VNF usage, but this is out of scope of NFV.

For variant 2 and variant 3, the two tenants use the same VNF Package but there is an instantiation of the VNF for each tenant. The VNF instance ID, in this case, is different for the NMT1 and the NMT2.

It is recommended that the VNF-LM is provided with information to discriminate the VNF instance for NMT1 and NMT2, to allocate during the instantiation or scaling the corresponding license entitlement right unit. For the case when different VNF-LM are used for the two tenants, it is recommended that the VNF-LM is provided with information to discriminate the VNF instance for which it manages the license entitlement rights. See solution proposal #3.8 in clause 6.3.9 and recommendation Mtenant.sharing.01.

As described in clause 5.7 of ETSI GR NFV-IFA 034 [i.17], the VNF-LM uses either the Os-Ma-nfvo or the Ve-Vnfm reference points for the management of licenses of VNF instances during the LCM of the VNF instances. The VNF-LM can also use the tenant information retrieved on these reference points on VNF package management and management of privileges proposed in key issue #4.

5.10.7 Conclusions of Use Case #9

Sharing of VNF packages and sharing of common services seem very important scenarios to be supported by NFV. The present document provides a starting point for studying the consequences and also provides first recommendations for supporting the sharing of VNF packages and the sharing of common services. However, a full analysis of this topic is beyond the scope of the present document version.

Sharing VNF or NS instances might lead to critical security and reliability issues that need to be carefully analysed. Therefore, it may be disallowed. A final decision whether NFV supports sharing of instances is out of scope of the present document.

6 Key Issues

6.1 Key Issue #1: Specifying Resource Isolation

6.1.1 Description

Resource isolation according to ETSI GS NFV 004 [i.4] is enforced by the NFVI. As illustrated in Use Cases #1 and #2 and referenced in most other Use Cases, during the lifecycle operations of NSs and their constituents hosted in the NFVI, the NFV-MANO functional blocks provide the information about isolation expectations to the VIM via resource groups.

The VIM provides operations to dynamically create virtualised resource affinity-or-anti-affinity constraints groups (see ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6]. NFVO and VNFM can use these operations to create the resource groups necessary to reflect the affinity or anti-affinity defined in the NSD and VNFD as shown in solution proposal #1.1.

Additionally, ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6] define infrastructure resource group as a logical resource collection grouping virtual resource instances assigned to a tenant along with Software Images.

Affinity-or-antiaffinity groups and infrastructure resource groups are identified by group ids.

Solution proposal #1.1 illustrates the usage of affinity-or-antiaffinity groups to define isolation, while solution proposal #1.3 uses infrastructure resource groups.

In deployments with multiple VIMs and multiple NFVOs or VNFMs creating resource groups, the uniqueness of group ids becomes an issue which is discussed in solution proposal #1.2. Additionally, when multiple NFV-MANO tenants use different NFVOs or VNFMs, but share a VIM (e.g. in Use Case #1), the tenants might wish to share group identifiers for anti-affinity groups. This leads to various difficulties which are discussed in solution proposal #1.2.

NOTE 1: Tenant management can provide an additional means to define isolation needs, which is discussed in key issue #2, clause 6.2.

NOTE 2: This key issue discusses resource isolation only. Affinity-or-antiaffinity groups and infrastructure resource groups do not provide management isolation. See key issue #2, clause 6.2 about management isolation, including the protection of the usage of resource groups by different tenants.

6.1.2 Solution Proposal #1.1: Affinity-or-antiaffinity Groups

The current NFV-MANO description uses resource groups in the interface to VIM and affinity/anti-affinity groups in the modelling:

1) At the network service level, rules are defined for the constituents of the NS:

    - NSD specifies AffinityOrAntiAffinityGroup as part of the deployment flavour
    (see ETSI GS NFV-IFA 014 [i.14], clause 6.3.2).
    These AffinityOrAntiAffinityGroups apply for the VNF instances created using different VNFDs, the Virtual Link instances created using different NsVirtualLinkDescs or the nested NS instances created using different NSDs.

    - Similarly, the NsProfile references affinity or anti-affinity groups which express affinity or anti-affinity relationships between the NS instance(s) created using this NsProfile and the NS instance(s) created using other NsProfile(s) in the same group.
    (see ETSI GS NFV-IFA 014 [i.14], clause 6.3.11).

    - During instantiate NS operation, the user can specify AffinityOrAntiAffinityRule as an input parameter

    - (see ETSI GS NFV-IFA 013 [i.13], clause 7.3.3).
    These AffinityOrAntiAffinityRules are applied in addition to those of the NSD. After instantiation they are stored in the NsInfo (see ETSI GS NFV-IFA 013 [i.13], clause 8.3.3.2).

    - The rule in this case can explicitly reference existing VNF instances, which is in addition to the AffinityOrAntiAffinityGroup defined in the NSD and mentioned above. See details in ETSI GS NFV-IFA 013 [i.13], clause 8.3.4.26.

    - AffinityOrAntiAffinityGroup for the constituents of an NS maps to TOSCA policy with type tosca.policies.nfv.NsAffinityRule or tosca.policies.nfv.NsAntiAffinityRule as specified in ETSI GS NFV-SOL 001 [i.9], clause 7.10.1.

2) At the VNF level, rules are defined for the constituents of the VNF and for the relation between instances created from the same VNFD:

    - VNFD specifies AffinityOrAntiAffinityGroup as part of the deployment flavour
    (see ETSI GS NFV-IFA 011 [i.11], clause 7.1.8.2).
    This AffinityOrAntiAffinityGroup applies for the virtualisation containers (e.g. virtual machines) to be created using different VDUs or internal VLs to be created using different VnfVirtualLinkDesc(s) in the same affinity or anti-affinity group.

    - VNFD specifies AffinityOrAntiAffinityGroup and localAffinityOrAntiAffinityRule in the VNF profile (see ETSI GS NFV-IFA 014 [i.14], clause 6.3.3).
    The localAffinityOrAntiAffinityRule applies between VNF instances created from this profile.
    The affinityOrAntiAffinityGroupId in the VnfProfile references the affinity or anti-affinity groups which expresses affinity or anti-affinity relationships between the VNF instance(s) created using this VnfProfile and the VNF instance(s) created using other VnfProfile(s) in the same group.

    - Each AffinityOrAntiAffinityGroup is defined by its group id and specifies either affinity or anti-affinity with its scope (NFVI_NODE, NFVI_POP, NETWORK_LINK_AND_NODE, etc.).
    See ETSI GS NFV-IFA 014 [i.14], clause 6.3.5 and ETSI GS NFV-IFA 011 [i.11], clause 7.1.8.12.
    The AffinityOrAntiAffinityGroup information element describes the affinity or anti-affinity relationship applicable between the VNF instances created using different VnfProfiles, the Virtual Link instances created using different VlProfiles or the nested NS instances created using different NsProfiles; that is, it describes the affinity/anti-affinity between elements of a network service.

    - During the Grant VNF Lifecycle Operation operation, the VNFM sends the PlacementConstraint so the NFVO can consider the affinity/anti-affinity information as described in ETSI GS NFV-IFA 007 [i.7], clauses 6.3.2 and 8.3.6. In addition, the VNFM can specify in the grant request the fallbackBestEffort attribute to allow resource assignments by the NFVO where the affinity/anti-affinity rules are not fully satisfied. In the referenced ETSI NFV documents, no information is given on the origin (or source) of the fallbackBestEffort. See recommendation Mtenant.affinitygrp.05 in clause 6.1.6 to add attributes/parameters about fallbackBestEffort.

- AffinityOrAntiAffinityGroup for the constituents of a VNF maps to TOSCA policy with type tosca.policies.nfv.AffinityRule or tosca.policies.nfv.AntiAffinityRule as specified in ETSI GS NFV-SOL 001 [i.9], clause 6.10.10. It allows the scope values nfvi_node, zone, zone_group, nfvi_pop, network_link_and_node.

3) At the level of the constituents of the VNF, rules are defined for the relation between instances created from the same VDU and rules for the internal virtual links:

   - VDU specifies AffinityOrAntiAffinityGroup and localAffinityOrAntiAffinityRule in the VduProfile (see ETSI GS NFV-IFA 011 [i.11], clause 7.1.8.3).
     The localAffinityOrAntiAffinityRule applies between the virtualisation containers (e.g. virtual machines) to be created based on this VDU.
     The affinityOrAntiAffinityGroupId in the VduProfile references the affinity or anti-affinity group(s) the VDU belongs to.

   - Virtual Links specify AffinityOrAntiAffinityGroup and localAffinityOrAntiAffinityRule in the VirtualLinkProfile
     (see ETSI GS NFV-IFA 014 [i.14], clause 6.3.4 and ETSI GS NFV-IFA 011 [i.11], clause 7.1.8.3, the description differs slightly between the two specifications).
     The localAffinityOrAntiAffinityRule applies between VLs instantiated from the referenced VLD.
     The affinityOrAntiAffinityGroupId in the VirtualLinkProfile references an affinity or anti-affinity group which expresses affinity or anti-affinity relationship between the VL(s) using this VirtualLinkProfile and the VL(s) using other VirtualLinkProfile(s) in the same group.

   - Inside a VNF, the LocalAffinityOrAntiAffinityRule is defined without a group id, and in the same way it specifies either affinity or anti-affinity with its scope (unlike the previous IE, it lists NFVI_POP, ZONE, ZONE_GROUP, NFVI_NODE, etc.).
     See ETSI GS NFV-IFA 014 [i.14], clause 6.3.8 and ETSI GS NFV-IFA 011 [i.11], clause 7.1.8.11. The description differs slightly between the two specifications and in IFA011 adds an additional attribute for nfviMaintenanceGroupInfo.
     The LocalAffinityOrAntiAffinityRule information element specifies affinity or anti-affinity rules applicable to VNFs or VLs instantiated from the same VNFD or VLD. Therefore it cannot be referenced from another VNF.

   - ETSI GS NFV-SOL 001 [i.9] does not differentiate between the VNF-level affinity/anti-affinity between the same or different VNFDs.

4) VIM provides capabilities to specify the affinity or anti-affinity during resource allocation, which are described in ETSI GS NFV-IFA 005 [i.5] for the use of NFVO and in ETSI GS NFV-IFA 006 [i.6] for the use of VNFM:

   - The VIM provides operations to dynamically create virtualised resource affinity-or-anti-affinity constraints groups:

     - For compute resources the operation is described in ETSI GS NFV-IFA 005 [i.5], clause 7.3.1.9 and ETSI GS NFV-IFA 006 [i.6], clause 7.3.1.9.

     - For network resources the operation is described in ETSI GS NFV-IFA 005 [i.5], clause 7.4.1.6 and ETSI GS NFV-IFA 006 [i.6], clause 7.4.1.6.

     - For storage resources the operation is described in ETSI GS NFV-IFA 005 [i.5], clause 7.5.1.9 and ETSI GS NFV-IFA 006 [i.6], clause 7.5.1.9.

   - The affinity-or-anti-affinity constraints groups are stored in an AffinityOrAntiAffinityConstraint information element and can be referenced either by a group id or by a list of resources (AffinityOrAntiAffinityResourceList see ETSI GS NFV-IFA 005 [i.5], clause 8.4.8.3 and ETSI GS NFV-IFA 006 [i.6], clause 8.4.8.3). The attributes of this IE are: type (affinity or anti-affinity), scope (various values applying either to compute, network or storage resources) and either group id or reference to a list of resources (see ETSI GS NFV-IFA 005 [i.5], clause 8.4.8.2 and ETSI GS NFV-IFA 006 [i.6], clause 8.4.8.2).

   - During resource allocation, the affinityOrAntiAffinityConstraints for a resource are specified using references to the groups previously created.

    - For compute resources see ETSI GS NFV-IFA 005 [i.5], clause 7.3.1.2 and ETSI GS NFV-IFA 006 [i.6], clause 7.3.1.2.
    - For network resources see ETSI GS NFV-IFA 005 [i.5], clause 7.4.1.2 and ETSI GS NFV-IFA 006 [i.6], clause 7.4.1.2.
    - For storage resources see ETSI GS NFV-IFA 005 [i.5], clause 7.5.1.2 and ETSI GS NFV-IFA 006 [i.6], clause 7.5.1.2.

- During resource migration, new affinityOrAntiAffinityConstraints for a resource are specified again using references to the groups previously created:
    - For compute resources see ETSI GS NFV-IFA 005 [i.5], clause 7.3.1.8 and ETSI GS NFV-IFA 006 [i.6], clause 7.3.1.8.
    - For storage resources see ETSI GS NFV-IFA 005 [i.5], clause 7.5.1.8 and ETSI GS NFV-IFA 006 [i.6], clause 7.5.1.8.
- During resource reservation, the affinityOrAntiAffinityConstraints for a resource are specified again using references to the groups previously created:
    - For compute resources see ETSI GS NFV-IFA 005 [i.5], clause 7.8.1.2 and ETSI GS NFV-IFA 006 [i.6], clause 7.8.1.2.
    - For network resources see ETSI GS NFV-IFA 005 [i.5], clause 7.8.2.2 and ETSI GS NFV-IFA 006 [i.6], clause 7.8.2.2.
    - For storage resources see ETSI GS NFV-IFA 005 [i.5], clause 7.8.3.2 and ETSI GS NFV-IFA 006 [i.6], clause 7.8.3.2.
    - Note that in case of resource reservation, there are separate input parameters for affinityConstraint and antiAffinityConstraint, both specifying an AffinityOrAntiAffinityConstraint IE which has this type also included as an attribute of type enum.
- ETSI GS NFV-SOL 014 [i.20] adds the affinity/anti-affinity constraints to the resource groups:
    - for compute resources, only NFVI-PoP and NFVI-Node are possible scope;
    - for network resources, virtual-switch, router, physical-NIC, physical-network, NFVI-Node are possible; and
    - for storage resource, only the value NFVI-Node is possible.

This concept enables the NFVO or VNFM to use proper resource groups with specified affinity/anti-affinity when allocating resources by calling VIM. According to ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6], there is an operation to create the resource groups.

The above concepts cover only affinity/anti-affinity within the scope of an NS and its constituents. Solution #1.4, clause 6.1.5 proposes a way to allow similar concept based on anti-affinity groups to isolate network services, making it suitable for multi-tenancy.

6.1.3 Solution Proposal #1.2: Affinity-or-anti-affinity constraints group identification

The VIM provides operations to dynamically create virtualised resource affinity-or-anti-affinity constraints groups (see ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6]. NFVO and VNFM can use these operations to create the virtualised resource affinity-or-anti-affinity constraints groups necessary to reflect the affinity or anti-affinity defined in the NSD and VNFD as shown in solution proposal #1.1. The VIM assigns a group id when the affinity-or-anti-affinity constraints group is created. Currently there is no indication about uniqueness of the group ids.

When the NFVO or VNFM specify further the affinity or anti-affinity, this group id is used. NFVO and VNFM can share the information about the group ids generated during LCM operations. Therefore the group id is defined uniquely in the scope of a VIM.

On the other side, the group id, once created can be used in any resource request and is not protected to be used only in the scope of a certain VNF, NS or tenant. Thus affinity-or-anti-affinity constraints groups can be used in an attack (e.g. intrusion attack by using an affinity group of another NS). Protection against such attacks is out of scope of the present document.

6.1.4 Solution Proposal #1.3: Infrastructure resource groups

Besides affinity or anti-affinity groups, the VIM defines infrastructure resource groups. The resourceGroupId can be used in addition to affinityOrAntiAffinityConstraints during resource LCM requests.

The infrastructure resource groups provide a means for a logical grouping of virtual resources assigned to a tenant within an Infrastructure Domain.

ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6] do not specify the necessary operations for the management of resource groups for infrastructure tenants (e.g. creation of an infrastructure resource group, etc.), as explicitly stated in clause 7.1 of ETSI GS NFV-IFA 005 [i.5] and clause 7.1 of ETSI GS NFV-IFA 006 [i.6].

Currently no isolation requirements can be defined for infrastructure resource groups. Therefore, they do not provide all necessary means to define isolation between tenants. See clause 6.1.6 for recommendations to enable that.

The infrastructure resource group ids, once created can be used in any resource request and are not protected to be used only in the scope of a certain VNF, NS or tenant. Thus infrastructure resource groups can be used in an attack. Protection against such attacks is out of scope of the present document.

Recommendation Mtenant.resourcegrp.03 suggests to study how tenant information from Os-Ma-nfvo or Or-Or reference point can be mapped to infrastructure resource groups. As it is studied in key issue #2, especially in solution proposal #2.2, the tenant can be identified by the client used in the OAuth mechanism in the access token as defined in ETSI GS NFV-SOL 013 [i.31] and ETSI GS NFV-SEC 022 [i.34].

6.1.5 Solution Proposal #1.4: Use nested NSs to express tenant isolation

As illustrated in Use Case #4, clause 5.5 and also in Use Case #7, Figure 5.8.2.2-2, the isolation needs of network services can be expressed via anti-affinity rules in a nesting NS. Thus users that want to isolate network services, can create such a nesting NS for the sole purpose of creating the anti-affinity constraints for NSs.

If this is used for multi-tenancy, such a nesting NS can only be created by an independent NFV-MANO user, so privacy of the tenants is provided. Changes of isolation constraints can be done using update NS operations.

The solution proposal provides a good means to express isolation constraints, but several aspects of multi-tenancy are not addressed:

- Management isolation cannot be achieved with this mechanism.
- Shared use of NSs cannot be defined.
- Usage is not very practical since an independent owner of the nesting NS is in control of defining the isolation constraints.

No further recommendations are derived from this solution proposal.

6.1.6 Recommendations

The recommendations related to affinity-or-antiaffinity groups are listed in Table 6.1.6-1.

Table 6.1.6-1: Recommendations related to affinity-or-antiaffinity groups

Identifier	Recommendation description	Comments and/or traceability
Mtenant.affinitygrp.01	Align the description of AffinityOrAntiAffinityGroup information element between stage 2 documents, e.g. ETSI GS NFV-IFA 014 [i.14] and ETSI GS NFV-IFA 011 [i.11].	Solution Proposal #1.1
Mtenant.affinitygrp.02	Align the possible scope values between AffinityOrAntiAffinityGroup information elements and LocalAffinityOrAntiAffinityRule information elements in stage 2 documents, e.g. ETSI GS NFV-IFA 014 [i.14] and ETSI GS NFV-IFA 011 [i.11].	Solution Proposal #1.3
Mtenant.affinitygrp.03	Align the description of AffinityOrAntiAffinityGroup information element between stage 3 documents, e.g. ETSI GS NFV-SOL 001 [i.9] and ETSI GS NFV-SOL 014 [i.20].	Solution Proposal #1.1
Mtenant.affinitygrp.04	Align the usage of affinity/anti-affinity between stage 2 and stage 3 specifications.	Solution Proposal #1.1
Mtenant.affinitygrp.05	Analyse and provide a way to specify the attribute fallbackBestEffort	Solution Proposal #1.1
Mtenant.affinitygrp.06	Align the description of affinity/anti-affinity during resource allocation with the description of affinity/anti-affinity during resource reservation.	Solution Proposal #1.1
Mtenant.affinitygrp.07	Analyse whether VNFM needs to provide operations to manage affinity-or-anti-affinity groups for VNFs/VNF instances.	Solution Proposal #1.1
Mtenant.affinitygrp.08	Specify that an affinity-or-anti-affinity group id needs to be unique within a VIM.	Solution Proposal #1.2

The recommendations related to resource groups are listed in Table 6.1.6-2.

Table 6.1.6-2: Recommendations related to infrastructure resource groups

Identifier	Recommendation description	Comments and/or traceability
Mtenant.resourcegrp.01	Specify the requirements and operations for the management of infrastructure resource groups. See note 2 in clause 7.1 of ETSI GS NFV-IFA 005 [i.5].	Solution Proposal #1.3
Mtenant.resourcegrp.02	Specify the isolation expectations for infrastructure resource groups.	Solution Proposal #1.3
Mtenant.resourcegrp.03	Revisit specification of Os-Ma-nfvo and Or-Vnfm to specify how tenant information from Os-Ma-nfvo or Or-Or reference points can be mapped to infrastructure resource groups. See note.	Solution Proposal #1.3
Mtenant.resourcegrp.04	Specify infrastructure resource group when creating CIS clusters via the CCM. Extend the modelling of CIS clusters to include infrastructure resource groups	Solution Proposal #1.3
Mtenant.resourcegrp.05	Specify that an infrastructure resource group id needs to be unique within a VIM.	Solution Proposal #1.3
Mtenant.resourcegrp.06	Revisit use of resource groups between ETSI GS NFV-IFA 005 [i.5] and ETSI GS NFV-IFA 006 [i.6] which is not completely consistent.	See Use Case #1, clause 5.2.4
Mtenant.resourcegrp.07	Revisit interfaces of Or-Vi and Vi-Vnfm reference points to add attributes and parameters related to resource groups where missing.	See Use Case #1, clause 5.2.4
NOTE: As studied in key issue #2, especially in solution proposal #2.2, the tenant can be identified by the client used in the Oauth mechanism in the access token as defined in ETSI GS NFV-SOL 013 [i.31] and ETSI GS NFV-SEC 022 [i.34].		

6.2 Key Issue #2: Tenant Identification

6.2.1 Description

Use cases #1, #6, #7, #8 do not directly expect that NFV-MANO is aware of the tenant that uses a resource or entity. The isolation could be managed without identifying tenants. However, the remaining Use Cases can best be implemented using a tenant identification, or management access to resources or entities is restricted based on the tenants.

ETSI GS NFV-IFA 010 [i.10] provides a set of requirements for multi-tenancy which include:

- Requirements on NFVO for tenant management and tenant aware LCM operations in clause 6.14.
- Requirements on VNFM for tenant management and associating resources to tenants in clause 7.10.
- Requirements on VIM for tenant management and associating resources to tenants, as well as tenant specific use of software images in clause 8.7.
- Requirements on VIM for tenant awareness during resources reservation in clause 8.2.2.
- Requirements on VIM for quota management per consumer (e.g. tenant) in clause 8.2.9.
- General requirement to provide the identification of an appropriate tenant (infrastructure tenant, VNF tenant or NS tenant) when performing an operation.

However, NFV-MANO APIs currently do not specify operations and parameters in support of these capabilities.

As illustrated in Use Case #9 (clause 5.10), it is expected to identify on the Os-Ma-nfvo and the Ve-Vnfm interfaces, the tenant during a LCM operation of a VNF instance. Clause 5.10.6 describes this recommendation for the License Management of the VNF instances.

The information of the tenant/owner of the VNF instances is missing in the VNF LCM operation on Ve-Vnfm and Os-Ma-nfvo interfaces, described in ETSI GS NFV-IFA 008 [i.8] and ETSI GS NFV-IFA 013 [i.13] specifications.

The following solutions are proposed:

- Solution Proposal #2.1 proposes to add tenant management as already indicated in ETSI GS NFV-IFA 010 [i.10].
- Solution Proposal #2.2 illustrates how existing authorization mechanism can be extended to protect against management by another tenant.
- Solution Proposal #2.3 proposes to add information of the tenant/owner of the VNF instance (e.g. the service provider Id), e.g. as initial requester of the NS instantiation in which the VNF is included.
- Solution Proposal #2.4 describes other aspects of management isolation, including the capability to filter the LCM notification related to a VNF instance according to a specific tenant and providing tenant information to notifications.
- Solution Proposal #2.5 describes how authorization and authentication can be used to provide different levels of privilege to tenants.

6.2.2 Solution Proposal #2.1: Tenant management

As described above, ETSI GS NFV-IFA 010 [i.10] already provides a set of requirements for tenant management. However, the solution for those requirements is not specified at the reference points. This solution provides the necessary management for tenant information:

- An interface on Os-Ma-nfvo reference point is expected to provide capability to create, read, update, delete tenant information in the NFV-MANO system. The information stored about a tenant includes an identification and the respective isolation expectations. The tenant identification can be assigned by NFVO or provided to the NFVO.
- Other interfaces on the Os-Ma-nfvo reference point include parameter(s) for the tenant information.
- Also Or-Vnfm and other reference points include parameter(s) for the tenant information.
- Runtime information elements of resources describe the ownership of the resources.
- The owner of a resource can invoke all operations related to the resource (e.g. LCM operations, queries, subscriptions, etc.), other tenants can be restricted to a subset of operations or no access at all. See also key issue #2, especially solution proposal #2.2.

- There can be additional privileges to invoke operations related to the resources owned by other tenants. For details on privileges see key issue #2, especially solution proposal #2.3.

Some more information about capabilities of tenant management can be found in clause 5.3 of ETSI GR ZSM 010 [i.22], which partly goes beyond the requirements applicable for NFV and documented in ETSI GS NFV-IFA 010 [i.10].

6.2.3 Solution Proposal #2.2: Tenant authorization

NFV-MANO uses authorization and authentication mechanisms based on Oauth 2.0 as specified in ETSI GS NFV-SOL 013 [i.31], clause 8. This protects all NFV-MANO APIs to only be used by authorized consumers (tenants). In case of multi-tenancy, consumers are authorized not only to use specific APIs, but access control is necessary on the level of managed resources. Oauth 2.0 provides scope values to define authorization for associated permissions for a client. Scope values can be used to specify the interfaces a client has permission to use, and also can be used to specify the resources, a client has permission to access. This can be used to protect resources against access by clients without the necessary permission.

ETSI GS NFV-SOL 005 [i.32], Annex F, specifies the allowed authorization scope values for the Os-Ma-nfvo reference point, to set permissions for different operations of NS LCM and VNF Package Management interfaces. Permissions for certain NSs or VNF packages so far are not foreseen. Similarly, ETSI GS NFV-SOL 002 [i.33], Annex F, specifies the allowed authorization scope values for the VNF LCM interface on the Ve-Vnfm reference point, and ETSI GS NFV-SOL 003 [i.29], Annex G for the VNF LCM interface and VNF package management interface on the Or-Vnfm reference point.

ETSI GS NFV-SEC 022 [i.34], Annex A, provides information about the use of access tokens in the industry. It explains e.g. how OpenStack Keystone uses authorization scopes to define the authorization of a tenant for a project scope.

Therefore, this solution proposes to enhance the definition of authorization scope values on NFV-MANO interfaces to:

- protect all NFV-MANO interfaces, e.g. FCAPS;
- specify how authorization scope values are set to define resources, e.g. VNF instances or infrastructure resource groups.

6.2.4 Solution Proposal #2.3: Ownership

This solution proposes to introduce ownership for resources. When a resource (e.g. VNF package, VNF instance, infrastructure resource) is created, the caller of the respective operation is set as owner for the resource. One possibility to identify the tenant as owner is provided by the identification of the client used in the Oauth mechanism in the access token as defined in ETSI GS NFV-SOL 013 [i.31] and ETSI GS NFV-SEC 022 [i.34]. The owner automatically has all access rights for a resource within his privileges.

Additionally, the owner can be allowed to set permissions for another tenant or transfer ownership to another tenant. See key issue 4.

For example, the owner of VNF instances can be added as attribute to the vnfInstance data type.

An external consumer of the Ve-Vnfm interface could subscribe to the vnfIdentifierCreationNotification or to VnfLcmOperationOccurrenceNotification, then query the vnfInfo to get the vnfInstance data structure of the corresponding VNF Instance, where the tenant information will be present. For a better efficiency, the vnfInstance data structure could be included in the VnfInstanceSubscriptionFilter, to be able to filter on VNF instances of a certain tenant.

An external consumer of the Os-Ma-nfvo interface could subscribe to the NsChangeNotification to be notified about changes related to a VNF component of an NS, and then query information on the NS instance to get the NsInstance data, and within the latter, the VnfInstance data to get the tenant information.

As an optimization, the tenant information can be included in appropriate notifications and result parameters, see clause 6.3.10, recommendation Mtenant.sharing.01.

6.2.5 Solution Proposal #2.4: Management isolation

This solution proposes to use authorization and authentication as described in the solution #2.2. As is specified in ETSI GS NFV-SOL 013 [i.31] and ETSI GS NFV-SEC 022 [i.34], consumers of NFV-MANO APIs request authorization tokens to get authorized to use NFV-MANO APIs. In this solution proposal, it is emphasized that the authorization server provides different access tokens to consumers, using specific authorization scope values, thereby making sure that a tenant cannot access resources (e.g. VNFs, infrastructure) of another tenant.

In particular, this mechanism also makes sure that subscription to notifications is only possible for resources the user has authorization to subscribe to. However, additional functionality is necessary to enable wildcard subscriptions, e.g. a tenant can subscribe to all VNFs that he owns. See recommendation Mtenant.ownership.05 in Table 6.2.7-3.

Management isolation in addition to access control covers fault management and performance management.

Fault management is important for reliability of any system. In case of multi-tenancy, it is not only important that failures of resources of one tenant do not affect another tenant, but also that information about failures is provided only to affected tenants, which is supported by the subscription and filtering for notifications as described above.

Performance management for multi-tenancy needs to isolate performance information between the tenants which is also mainly supported by the same mechanisms for authorizing subscription and query operations. But in addition to performance measurements and other performance information, tenant management can provide means to define performance or resource quota for the tenants. See key issue #6 for quota for tenants.

6.2.6 Solution Proposal #2.5: Levels of permission

This solution proposes to use authorization and authentication as described in the solution #2.2. As is specified in ETSI GS NFV-SOL 013 [i.31] and ETSI GS NFV-SEC 022 [i.34], consumers of NFV-MANO APIs request authorization tokens to get authorized to use NFV-MANO APIs. In this solution proposal, it is emphasized that the authorization scope values currently defined in ETSI GS NFV-SOL 005 [i.32], Annex F, ETSI GS NFV-SOL 002 [i.33], Annex F, and ETSI GS NFV-SEC 022 [i.34], Annex A, enable defining specific types of access, e.g. separate privilege for each operation on an interface, and read-only access. This covers all scenarios except update NS operations. Updating some attributes could be protected by special permissions, e.g. update type ADD_VNF or CREATE_SNAPSHOT could be allowed for a user who is not allowed to initiate CHANGE_VNFPKG (see Table 6.5.2.12-1 in ETSI GS NFV-SOL 005 [i.32] for the list of update types). A recommendation is added to revisit the possible authorization scope values in the light of multi-tenancy (Mtenant.tenantauth.06 in Table 6.2.7-2).

6.2.7 Recommendations

The recommendations related to tenant identification and management are listed in Table 6.2.7-1.

Table 6.2.7-1: Recommendations related to tenant identification and management

Identifier	Recommendation description	Comments and/or traceability
Mtenant.tenantmgmt.01	Specify an interface on the Os-Ma-nfvo reference point in ETSI GS NFV-IFA 013 [i.13] according to requirement Nfvo.Mtm.001 in ETSI GS NFV-IFA 010 [i.10], clause 6.14, with the capability to create, read, update, delete tenants.	Solution Proposal #2.1
Mtenant.tenantmgmt.02	Provide the capability to specify for a tenant its expectation to isolate resources. See note 1.	Solution Proposal #2.1
Mtenant.tenantmgmt.03	Revisit Os-Ma-nfvo interfaces to specify the tenant information where needed.	Solution Proposal #2.1
Mtenant.tenantmgmt.04	Revisit other reference points and service interfaces to specify the tenant information where needed.	Solution Proposal #2.1
Mtenant.tenantmgmt.05	Revisit attributes of runtime information elements to specify the owner of a resource. Resources can be NS, VNF or other constituent of an NS or compute, storage and network resources.	Solution Proposal #2.3
Mtenant.tenantmgmt.06	Add requirements to verify tenancy information (ownership or a privilege) before executing an operation. See note 2.	Solution Proposal #2.4
Mtenant.tenantmgmt.07	Add requirements to allocate resources for tenants according to their isolation needs. See note 3.	Use case #8, clause 5.9.4
NOTE 1: It is recommended that the expectations for isolation can also include a fallbackBestEffort option. NOTE 2: This includes the verification that only the owner or a tenant with proper privilege is able to subscribe to notifications. NOTE 3: In case of container based VNFs this includes the selection of appropriate CIS clusters and namespaces.		

The recommendations related to tenant authorization are listed in Table 6.2.7-2.

Table 6.2.7-2: Recommendations related to tenant authorization

Identifier	Recommendation description	Comments and/or traceability
Mtenant.tenantauth.01	Provide stage 2 specification, e.g. specific requirements for the authorization of tenants for all operations and resources. See note 1.	Solution Proposal #2.2
Mtenant.tenantauth.02	Revisit the specification of authorization scope values, so they allow specifying the scope of resources a tenant is allowed to access. See note 1.	Solution Proposal #2.2
Mtenant.tenantauth.03	Specify the authorization scope values for all NFV-MANO interfaces on all reference points and all service interfaces. See note 2.	Solution Proposal #2.2
Mtenant.tenantauth.04	Define the relation of tenant management, ownership of resources and authorization scope values.	Solution Proposal #2.2
Mtenant.tenantauth.05	Provide explicit requirements for the notification mechanism to enable subscription within the permitted scope of a tenant.	Solution Proposal #2.4
Mtenant.tenantauth.06	Revisit possible authorization scope values in the light of multi-tenancy.	Solution Proposal #2.5
NOTE 1: Resources in this context can be logical resources or infrastructure. NOTE 2: Interfaces can be on reference points or service interfaces.		

The recommendations related to ownership are listed in Table 6.2.7-3.

Table 6.2.7-3: Recommendations related to ownership

Identifier	Recommendation description	Comments and/or traceability
Mtenant.ownership.01	Define ownership for Network services and their constituents (including VNFs, VNF packages, descriptors, infrastructure resources, etc.)	Solution Proposal #2.3
Mtenant.ownership.02	As default, during resource creation, the initial requestor is set the owner for a resource.	Solution Proposal #2.3
Mtenant.ownership.03	Include information about ownership in operation results and notifications where appropriate.	Solution Proposal #2.3
Mtenant.ownership.04	Define which operations can result in a change of ownership.	Solution Proposal #2.3
Mtenant.ownership.05	Add the capability to subscribe to notifications according to the owner of the resource and according to privileges.	Solution Proposal #2.3
Mtenant.ownership.06	Add the capability to filter notifications according to the owner of the resource.	Solution Proposal #2.3

6.3 Key issue #3: Sharing of artifacts

6.3.1 Description

Use case #9 in clause 5.10.2 illustrates the need to share VNF packages, NSDs or other artifacts.

When two tenants want to use the same VNF package or NSD file archive, there are two different approaches:

- Each tenant can take the VNF package or NSD and onboard it to the NFV system. Thus the VNF package or NSD is onboarded twice. See solution #3.1.
- The artifact is onboarded once and can be used by multiple tenants. See solution #3.2.

As an option, in the first approach (first bullet above) a central catalogue or repository of VNF packages or of NSD file archives can be part of the solution. See solution #3.5.

ETSI GS NFV-SOL 005 [i.32] describes that some artifacts of a VNF can be provided either as part of the VNF package or from an external source. See solution #3.6 for details.

Container images can be provided via the Container Image Repository (CIR) as described in ETSI GS NFV-IFA 040 [i.19]. See solution #3.7 for multi-tenancy aspects of the CIR.

6.3.2 Solution Proposal #3.1: Multiple onboarding of a VNF package

ETSI GS NFV-IFA 011 [i.11] specifies in clause 7.1.2.2 that the vnfdId is a unique value and is also used as the unique identifier of the VNF package that contains this VNFD. This is consistent with the descriptor_id property defined in ETSI GS NFV-SOL 001 [i.9]. As a consequence of the globally unique vnfdId being used as package identifier, it does not seem possible for a tenant A to onboard a VNF package if a package with the same vnfdId has already been onboarded by another tenant B - even in case the tenant A has no visibility of the package onboarded by tenant B.

NOTE: As described in Use Case #9, clause 5.10.2, it can happen in multi-tenancy environments that two tenants want to onboard the same VNF package independently from each other. Tenants expect to maintain their own packages (e.g. for upgrades) and instantiate VNFs from them. Tenants are not aware whether another tenant onboards the same VNF package.

ETSI GS NFV-SOL 005 [i.32] introduces vnfPkgId as the identification of the VNF package. The vnfPkgId is allocated by the NFVO, when the individual VNF package resource is created. This allows multiple tenants to onboard the same VNF package (with same vnfdid). According to the VNF Package on-boarding procedure in ETSI GS NFV-SOL 016 [i.47], clause 5.1.3, at the time of creating the vnfPkgId (during CreateVnfPkgInfoRequest), the NFVO is not aware of the VNFD of the VNF package that is to be uploaded.

ETSI GS NFV-SOL 003 [i.29], clause 10 defines two resource sub-trees with identical structure, which only differ in the identifier per "Individual VNF package" resource. VNF packages can be identified by the vnfPkgId or by the vnfdId. In ETSI GS NFV-SOL 003 [i.29] it is assumed that for any given vnfdId value, there is at most one associated vnfPkgId value in the whole resource tree visible to the VNFM. This is important, since during LCM, a VNF is always referenced by the vnfdId only.

In clause 10.2 of ETSI GS NFV-SOL 003 [i.29], the "vnf_packages" subtree is deprecated, but in ETSI GS NFV-SOL 005 [i.32] and ETSI GS NFV-SOL 016 [i.47] the "vnf_packages" subtree is used for the onboarding. A recommendation is provided in clause 6.3.10 to propose better alignment.

The present solution proposes to use the "vnf_packages" subtree during the onboarding of VNF packages as described in ETSI GS NFV-SOL 005 [i.32] and ETSI GS NFV-SOL 016 [i.47]. This allows onboarding of multiple packages with the same vnfdId. However, some constraints are relevant:

- Packages with the same vnfdId can be assumed identical.
- In case of signed packages, the packages with same vnfdId can look different.
- In case of identical packages, the NFVO can decide to store packages only once.
- During subsequent LCM operations, the packages are identified by the vnfdId contained in them. Therefore, during LCM operations only one package with the specified vnfdId can be visible for a consumer.
- During LCM operations, the NFVO can provide different URIs to VNFM, to allow the same VNFM manage VNF instance with the same vnfdId, but coming from different VNF packages.

6.3.3 Solution Proposal #3.2: Multiple tenants using the same onboarded VNF package

In this solution proposal, the tenants make sure that the VNF package is on-boarded only once and receive information and permission to use the VNF package for instantiation (see key issue #4). This can be achieved in different ways:

1) The tenants are aware of each other and share the necessary information by means outside of NFV-MANO.

2) There is a separate function (e.g. a broker) that synchronizes between tenants, on-boards VNF packages if not yet on-boarded by another tenant, provides tenants with information and if necessary privileges. This function can be inside or outside of NFV-MANO. It is not described in the present document.

6.3.4 Solution Proposal #3.3: Multiple tenants onboarding the same VNF package using different vnfdId

This solution proposal tries to overcome the issues illustrated in Solution Proposal #3.1 by using different vnfdId even when the rest of the VNF packages is identical. In the case that multiple tenants onboard their own copy of the same VNF package, it can be guaranteed outside of NFV-MANO, that each copy of a VNF package contains a different vnfdId. But according to ETSI GS NFV-IFA 011 [i.11], the vnfdId is an attribute of the VNFD (same as the descriptor_id property defined in ETSI GS NFV-SOL 001 [i.9]), it is stored inside the VNF package. If vnfdId is different, the package is different.

If multiple tenants want to onboard their own copies of the same VNF independently, each VNF Package of the same VNF contains a different globally unique vnfdId.

NFV-MANO, can use information like vnfdExtInvariantId, vnfProvider, vnfProductName, vnfSoftwareVersion (see clause 7.1.2 of ETSI GS NFV-IFA 011 [i.11]) to determine whether VNF packages contain the same VNF.

6.3.5 Solution Proposal #3.4: Multiple onboarding of an NSD

ETSI GS NFV-IFA 014 [i.14] specifies in clause 6.2.2.2 that the nsdId is globally unique and is also used to identify the NSD information element.

ETSI GS NFV-SOL 005 [i.32] clarifies that the "Create NS identifier" operation uses the nsdId as input, and in the output result, the operation provides the NsInstance.

Thus, an NSD can only be used once to create an NSD resource and multiple onboarding of a network service are rejected. Multiple tenants using the same NSD can do that if they make sure that the NSD is on-boarded only once and they receive information and permission to use the NSD. This can be achieved in different ways:

1) The tenants are aware of each other and share the necessary information by means outside of NFV-MANO.

2) There is a separate function (e.g. a broker) that synchronizes between tenants, on-boards NSD file archives if not yet on-boarded by another tenant, provides tenants with information and if necessary privileges. This function can be inside or outside of NFV-MANO. It is not described in the present document.

6.3.6 Solution Proposal #3.5: Catalogue for NFV artifacts

Solution proposals above can be supported by a central catalogue or repository shared between tenants. This would avoid multiple onboarding when a VNF package or NSD is used by multiple tenants. Such catalogue or repository is therefore recommended, but no requirements on NFV-MANO are derived in the present document.

6.3.7 Solution Proposal #3.6: External locations of VNF artifacts

According to ETSI GS NFV-SOL 005 [i.32] some artifacts of a VNF can be provided from a source external to the VNF package. This is possible in case of VnfPackageSoftwareImageInfo (see clause 9.5.3.2 of ETSI GS NFV-SOL 005 [i.32]), VnfPackageArtifactInfo (clause 9.5.3.3), VnfcSnapshotImageInfo (clause 11.5.3.2) and SnapshotPkgArtifactInfo (clause 11.5.3.3).

In this case, such external locations are used for retrieval of artifacts. The data model in ETSI GS NFV-SOL 005 [i.32] only specifies a URI where to retrieve the artifact. It is expected that access control is in place to retrieve this artifact in the context of multitenancy.

6.3.8 Solution Proposal #3.7: Multi-tenancy of the CIR

The CIR can be considered a location of artifacts, similar to clause 6.3.5. However, it is part of NFV-MANO and as such can fulfil the same access control as any other NFV-MANO component. Recommendations in clause 6.2.7 apply also for the CIR, e.g. Mtenant.tenantmgmt.04 and Mtenant.tenantmgmt.06.

6.3.9 Solution Proposal #3.8: License management in case of shared artifacts

As described in clause 5.10.6 and clause 5.7 of ETSI GR NFV-IFA 034 [i.17], the VNF-LM uses either the Os-Ma-nfvo or the Ve-Vnfm reference points for the management of licenses of VNF instances during the LCM of the VNF instances and during VNF package management. VNF-LM can use tenant information contained in notifications to identify the tenant and check for proper licensing for a tenant. See recommendation Mtenant.sharing.01.

See also key issue #4 about management of privileges. VNF-LM can similarly use tenant information contained in notifications to identify the tenant and check for proper licensing when privileges are managed.

6.3.10 Recommendations

The recommendations related to sharing artifacts are listed in Table 6.3.10-1.

Table 6.3.10-1: Recommendations related to sharing artifacts

Identifier	Recommendation description	Comments and/or traceability
Mtenant.sharing.01	It is recommended to specify a requirement to include tenant information in notifications and result parameters of operations where appropriate.	Solution Proposal #3.8
Mtenant.sharing.02	It is recommended to revisit the description in ETSI GS NFV-SOL 003 [i.29], ETSI GS NFV-SOL 005 [i.32] and ETSI GS NFV-SOL 016 [i.47] about the handling of the "vnf_packages" subtree in the context of Multitenancy, to better clarify the interworking between NFVO and VNFM.	Solution Proposal #3.1

6.4 Key issue #4: Management of Privileges

6.4.1 Description

As specified in ETSI GS NFV-SOL 013 [i.31] and ETSI GS NFV-SEC 022 [i.34], consumers of NFV-MANO APIs request authorization tokens to get authorized to use NFV-MANO APIs. The authorization server has knowledge about the privileges of tenants to use NFV-MANO APIs on certain managed objects.

However, how these privileges are managed is for future study and are not covered in the present document.

6.4.2 Recommendations

The recommendations related to sharing artifacts are listed in Table 6.4.2-1.

Table 6.4.2-1: Recommendations related to privilege management

Identifier	Recommendation description	Comments and/or traceability
Mtenant.privilegemgmt.01	It is recommended to specify requirements and interfaces for the management of privileges of tenants to use NFV-MANO APIs on certain managed objects.	

6.5 Key issue #5: Sharing virtual storage and PaaS services

6.5.1 Description

As illustrated in Use Case #9, clause 5.10, NFV support for 5G, as studied in ETSI GR NFV-IFA 037 [i.46], leverages shared virtual storage and PaaS services. While ETSI GR NFV-IFA 037 [i.46] analyses general aspects of sharing, the present key issue adds multi-tenancy aspects.

6.5.2 Solution Proposal #5.1: Sharing virtual storage

This solution proposal is based on the recommendations 5gnfv.desc.004 and 5gnfv.desc.005 about sharing virtualised storage resources in ETSI GR NFV-IFA 037 [i.46]. These recommendations introduce attributes in VNFD and NSD about sharing of virtualised storage resources. In the same way, attributes can indicate whether sharing with another tenant can be allowed. When the sharing of the resources is established, e.g. the tenants share the address or connection information, management privileges as discussed in key issue #2 can control the permission to establish the sharing. Additional control of the storage access depends on the capability of the storage system and is out of scope of the present document. As indicated in the Annex on multi-tenancy in Anuket project (clause A.3), Anuket Reference Model for Cloud Infrastructure (RM) [i.38] provides some related information.

6.5.3 Solution Proposal #5.2: Sharing PaaS services

This solution proposal is based on the VNFD and NSD specification evolved from recommendations 5gnfv.desc.001 in ETSI GR NFV-IFA 037 [i.46]. As specified in clause 7.1.21.2 of ETSI GS NFV-IFA 011 [i.11], a PaasServiceRequest enables the capability of a VNFD to describe the information and requirements by the VNF in terms of PaaS Services (such as VNF Common/Dedicated Services) that the VNF needs for its operation. In the PaasServiceRequest, the attribute "usageFormat" describes the intended usage format of the PaaS Service, including the cases of "COMMON", which is when a PaaS Service is to be used as a VNF Common Service, or "UNDEFINED", in which case the usage is determined by the management and orchestration system or some operational policies.

Furthermore, as specified in clause 17.3 of ETSI GS NFV-IFA 010 [i.10], the PaaS Services Repository (PSR) function has the capability to inventory the PaaS Service instances that have been deployed. The PSR keeps also information about the association of the deployed PaaS Services instances to the PaaS Services consumers, such as one or more VNF instances or NS instances. This will provide information to a PSR consumer about which PaaS Service instances are shared among VNF/NS instances.

In the present solution, it is expected that authorization and access control to the PaaS Service is present to avoid impacts on data isolation and access to the same PaaS Service instance by multiple consumers, such as VNF instances. As per the latest VNF lifecycle operation granting interface specification from ETSI GS NFV-IFA 007 [i.7], PaaS Service requests can be indicated in the granting request and be granted as PaaS assets in the output, similarly to handling any other kind of virtualized and containerized resource for a VNF instance. This means that the NFVO also plays a role in the granting of the usage of the PaaS Services to the VNF instances. If the same PaaS Service instance is used, and thus shared, by multiple VNF instances, the "paasServiceId" will refer to the same identifier value. Depending on the kind of PaaS Service, and the mechanism to access it by the VNF instance, the "paasServiceHandle" might have different values.

6.5.4 Recommendations

The recommendations related to sharing virtual storage and PaaS services are listed in Table 6.5.4-1.

Table 6.5.4-1: Recommendations related to sharing virtual storage and PaaS services

Identifier	Recommendation description	Comments and/or traceability
Mtenant.sharestorage.01	It is recommended to specify a requirement for the VNFD to support defining whether the VNF can use shared virtualised storage resources with VNFs of another tenant. See note.	Solution proposal #5.1
Mtenant.sharestorage.02	It is recommended to specify a requirement for the NSD to support defining which VNFs are expected to use virtualised storage resources that are to be shared among VNFs of another tenant. See note.	Solution proposal #5.1
NOTE: The recommendation is based on recommendation 5gnfv.desc.004 / 5gnfv.desc.005 in ETSI GR NFV-IFA 037 [i.46], clause 6.4.		

6.6 Key issue #6: Quota management for tenants

6.6.1 Description

In multi-tenancy system it is beneficial to restrict usage of resources by tenants to certain limits. The solution proposals in this clause propose to introduce quota per tenant.

ETSI GS NFV-IFA 010 [i.10] includes already quota management to prevent excessive resource consumption in the VIM by a given consumer of a virtualised resource. This is set in relation to the resource capacity, e.g. when multiple VNFs share a physical resource. For container based VNFs quota can be defined using namespaces.

In case of multi-tenancy systems, quota can also be managed per tenant. These can include quota for infrastructure resource usage as well as quota to onboard or instantiate NFV entities such as VNFs, VNF instances, VNF packages, NSs, or also the number of NFV-MANO operations. Use case #6 in clause 5.7 illustrates a special case using management level agreements (MLA) between a MANO-T and MANO-P system.

6.6.2 Solution Proposal #6.1: Tenant quota management

This solution proposal introduces quota management per tenant. As part of the tenant management defined in solution proposal #2.1 in clause 6.2.2 limits for the resource usage and NFV-MANO usage can be defined. The NFVO can map the tenant quota to the resources used by the tenant and can enforce the limits using existing capabilities of quota management as described in ETSI GS NFV-IFA 010 [i.10] for the various NFV functional blocks.

6.6.3 Recommendations

The recommendations related to tenant quota management are listed in Table 6.6.3-1.

Table 6.6.3-1: Recommendations related to tenant quota management

Identifier	Recommendation description	Comments and/or traceability
Mtenant.quota.01	Specify requirements and interfaces for tenant quota management.	Solution proposal #6.1
Mtenant.quota.02	Specify requirements to enforce quota for usage of infrastructure resources by tenants.	Solution proposal #6.1
Mtenant.quota.03	Specify requirements to enforce quota for NFV-MANO operations per tenant.	Solution proposal #6.1

6.7 Key Issue # 7 - Management of MLA

6.7.1 Description

Use case #6 (see clause 5.7) enables the provision of MANO-T systems for respective tenants in order to grant them management autonomy. An MLA is negotiated with the tenants that determines the operational and functional bounds of the MANO-T systems. The MLA template essentially identifies the operational bounds of a MANO-T by specifying the interfaces and operations that a MANO-T is allowed to execute. Once instantiated, the operations of individual MANO-T system instances are monitored and enforced by the MANO-P system in compliance with the respective MLA. It is noted that a MANO-T is considered as a managed object of a MANO-P system. Therefore, the interfaces and operations defined for the LCM of NS instances can potentially be leveraged towards the LCM of MANO-T systems, albeit with necessary extensions to ensure enforcement of LCM operations as per the MLA.

The existing NFV-MANO APIs lack support for the realization of MANO-T systems and the existing interfaces, operations, information elements need to be investigated for supporting MLA templates and for the management of the MANO-T instances. In view of this the following solutions are proposed:

- Solution Proposal #7.1 proposes solution options for specifying MLA parameters.
- Solution Proposal #7.2 proposes solution for the management of the MANO-T system and its compliance with the MLA.

6.7.2 Solution Proposal #7.1: Specifying MLA Parameters

In order to specify MLA parameters for determining the functional and operational bounds of a MANO-T system, two solution options are proposed.

Option #MLAtemplate.1: Specifying MLA Parameters within NSD

Since a MANO-T is considered as a managed object of MANO-P and its instantiation process is similar to that of an NS instance (see clause 5.7.2.2.3), therefore this solution option proposes to specify MLA parameters by leveraging the existing NSD templates as specified in ETSI GS NFV-SOL 001 [i.9] and extend it with relevant parameters, such as tenant id, the id of the MANO-T system and its functional blocks (NFVO-T, VNFM-T, VIM-T), the relevant permissions, etc.

Option #MLAtemplate.2: Specifying a separate MLA template

This solution proposes to define a separate MLA template for each instance of the MANO-T system. The MLA template specifies all the necessary parameters indicating the permissions and functional/operational bounds of the MANO-T system. The MANO-T system will then ensure compliance of the on-boarded NSD files with the MLA template.

6.7.3 Solution Proposal #7.2: MLA Compliance

Clause 5.7.2.4 provides an example of a base flow where a MANO-T system ensures compliance when an LCM operation is required for an NS that exceeds the MLA bounds. In such a situation, the NFVO-T informs the NFVO-P of such an event and a decision whether to allow or disallow such an operation is reached. This implies a need to have relevant interface(s) to enable the exchange of relevant information between NFVO-T and NFVO-P to reach a suitable decision. Two solution options are proposed to realize the relevant information exchange between the NFVO-T and NFVO-P.

Option #MLAinterface.1: Specifying new reference point between NFVO-T and NFVO-P

This solution option proposes a new reference point between the NFVO-T and NFVO-P, over which required interfaces and operations are specified which will enable the NFVO-T and NFVO-P to exchange relevant information to collaborate on LCM operations decisions, and also exchange reports, notifications, etc.

Option #MLAinterface.2: Leveraging existing reference points for the inter-communication between NFVO-T and NFVO-P

Since the MANO-T is considered as a managed object of MANO-P system, therefore this solution option proposes to use the existing MANO reference points. In this regard there are two possible options:

- ***Option #MLAinterface.2.1:*** This option proposes to utilize the Ve-Vnfm reference point by leveraging the interfaces and information elements specified in ETSI GS NFV-IFA 008 [i.8] and/or specifying new interfaces to enable the intercommunication and notifications between the MANO-T and MANO-P systems, and their FBs respectively, for compliance purposes.

- ***Option #MLAinterface.2.2:*** Since a tenant can be associated to an administrative domain, this option proposes to leverage the interfaces and information elements specified for the Or-Or reference point as specified in ETSI GS NFV-IFA 30 [i.16], and/or specifying new interfaces over this reference point to enable the intercommunication and notifications between the MANO-T and MANO-P systems, and their FBs respectively, for compliance purposes.

6.7.4 Recommendations

The recommendations related to the management of MLA templates are listed in Table 6.7.4-1.

Table 6.7.4-1: Recommendations related to managing MLA templates

Identifier	Recommendation description	Comments and/or traceability
Mtenant.mla.01a	Revisit ETSI GS NFV-IFA 014 [i.14] and specify an MLA parameter information element that contains parameters relevant to an MLA negotiated between the tenant and the MANO-P.	Solution Proposal #7.1 option #MLAtemplate.1. See note.
Mtenant.mla.01b	Specify a new template, the MLA template, that contains the necessary parameters indicating the permissions and functional/operational bounds of the MANO-T.	Solution Proposal #7.1 option #MLAtemplate.2. See note.
Mtenant.mla.02	Specify interfaces on the Os-Ma-nfvo reference point in ETSI GS NFV-IFA 013 [i.13] for enabling the triggering, negotiating, updating, deleting of an MLA template for the NMT.	See clause 5.7.2.2
Mtenant.mla.03	Specify a notify operation on the Os-Ma-nfvo reference point in ETSI GS NFV-IFA 013 [i.13] for supporting MLA related notifications between NFVO-T and NMT.	See example flows in clause 5.7.2.
Mtenant.mla.04a	Specify a new reference point between NFVO-T and NFVO-P for the exchange of relevant information on LCM operations decisions, and also exchange reports, notifications, etc.	Solution Proposal #7.2 option #MLAinterface.1. See note.
Mtenant.mla.04b	Revisit ETSI GS NFV-IFA 008 [i.8] to specify interfaces on the Ve-Vnfm reference point for the intercommunication and notifications between the MANO-T and MANO-P.	Solution Proposal #7.2 option #MLAinterface.2.1. See note.
Mtenant.mla.04c	Revisit ETSI GS NFV-IFA 030 [i.16] to specify interfaces on the Or-Or reference point for the intercommunication and notifications between the MANO-T and MANO-P.	Solution Proposal #7.2 option #MLAinterface.2.2. See note.
NOTE: The decision on which solution option and related recommendation to adopt, can be made during the normative phase. Options are deemed to be exclusive (i.e. either one or the other, but not both) within the respective solution proposal set.		

7 Recommendations

The present document derives recommendations from the Use Cases and key issues, related to the following topics:

- Affinity-or-antiaffinity groups, see Table 6.1.6-1
- Infrastructure resource groups, see Table 6.1.6-2
- Tenant management and identification of tenants, see Table 6.2.7-1
- Tenant authorization, see Table 6.2.7-2
- Ownership, see Table 6.2.7-3
- Sharing of artifacts, see Table 6.3.10-1
- Management of privileges, see Table 6.4.2-1
- Sharing storage and PaaS services, see Table 6.5.4-1
- Tenant quota management, see Table 6.6.3-1
- Management of MLA templates, see Table 6.7.4-1

In many cases these recommendations can be implemented independently from each other, thus the specification can be provided during several NFV releases or release drops.

8 Conclusions

The present document has studied various Use Cases and deployment scenarios for multi-tenancy in NFV deployments supporting multi-tenancy. It was found that many aspects of multi-tenancy are already provided in early NFV releases. However, several aspects were found missing in the present standardization specifications for NFV-MANO. Recommendations have been provided to close those gaps.

Annex A: Multi-Tenancy in Open Source

A.1 Multi-Tenancy in OpenStack®

A.1.1 General

The OpenStack® concepts supporting multi-tenancy in NFV-MANO context mainly apply to the VIM, but can be considered also for the higher level NFV-MANO tenancy concepts.

NOTE: The OpenStack® Word Mark and OpenStack Logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. ETSI is not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

As described in OpenStack® Operation Guide [i.24], in OpenStack®, "a group of users is referred to as a project or tenant". The terms "project" and "tenant" are interchangeable and used in parallel for historical reasons.

Projects/tenants "own" resources.

OpenStack® provides a user management, and users can only be created having one or several projects. The user management allows to protect management of resources to the users associated with the project that is using the resources. Users can also be grouped and role based access to the projects is provided this way.

Thus OpenStack® implements tenancy concepts supporting both, resource isolation and management isolation.

A.1.2 Feedback from Tacker

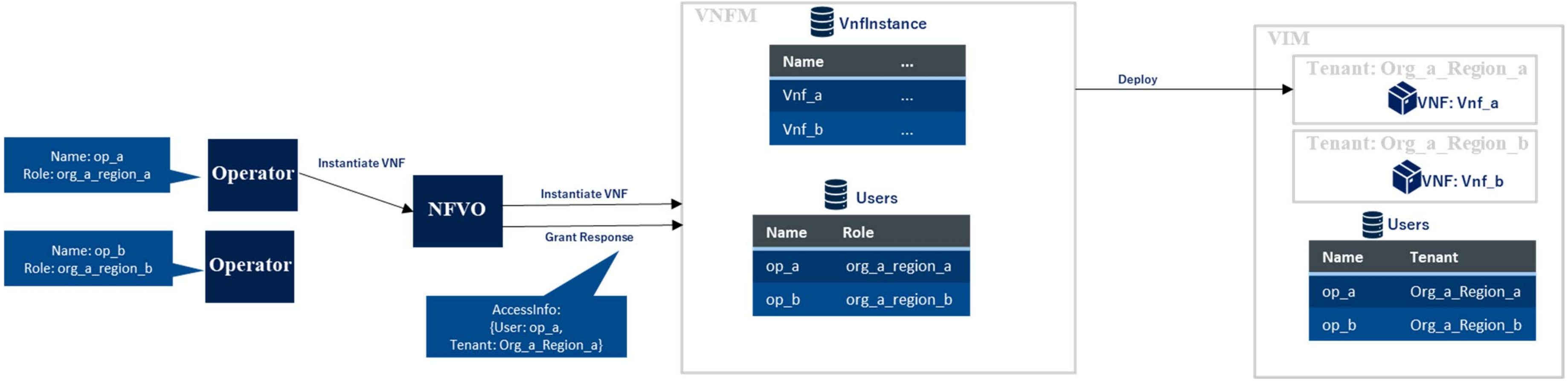
When providing feedback to NFV-SOL working group, the Tacker project also included a request to add attributes to store tenant information of VNF instances.

Tenants of VNFM and tenants of VIM are not always in a 1:1 relationship.

There can be cases where a VNFM does not have tenants, but a VIM has two tenants.

User credentials used in API calls for VNFM can be different from credentials in AccessInfo of grant responses.

Figure A.1.2-1 illustrates the attributes used during grant.



Name	...
Vnf_a	...
Vnf_b	...

Name	Role
op_a	org_a_region_a
op_b	org_a_region_b

Name	Tenant
op_a	Org_a_Region_a
op_b	Org_a_Region_b

Figure A.1.2-1: Use new attribute in grant

Problem: VNFM cannot perform the access control on VnfInstance with tenants of VIM.
Operators can see the information that they are not expected to see e.g. *op_a* is only allowed to access resources in tenant: *Org_a_Region_a*, however, the responses sent from VNFM also include resources in tenant: *Org_a_Region_b*.

Figure A.1.2-2 illustrates that the response discloses information to the wrong tenant.

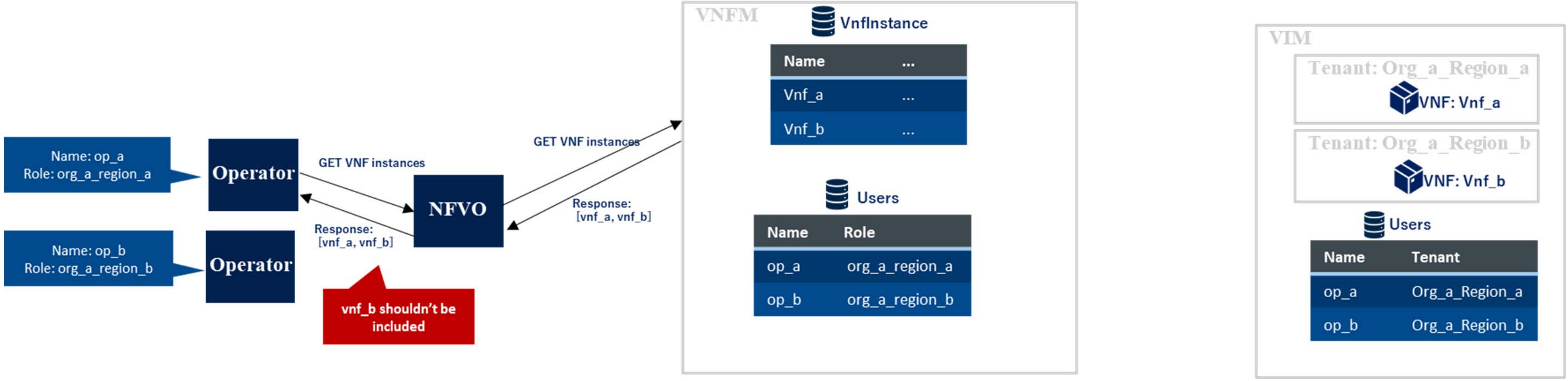


Figure A.1.2-2: Illustrate response to wrong tenant

Tacker proposes candidate attributes to store tenant information of the VNF in the VIM. Excerpts from ETSI GS NFV-SOL 004 [i.35] with normative provisions are quoted:

- **metadata:** Metadata that the VNF provider foresees are expected to be declared in the VNFD. "The VNFM shall accept requests to write metadata that are not declared in the VNFD", see ETSI GS NFV-SOL 003 [i.29].

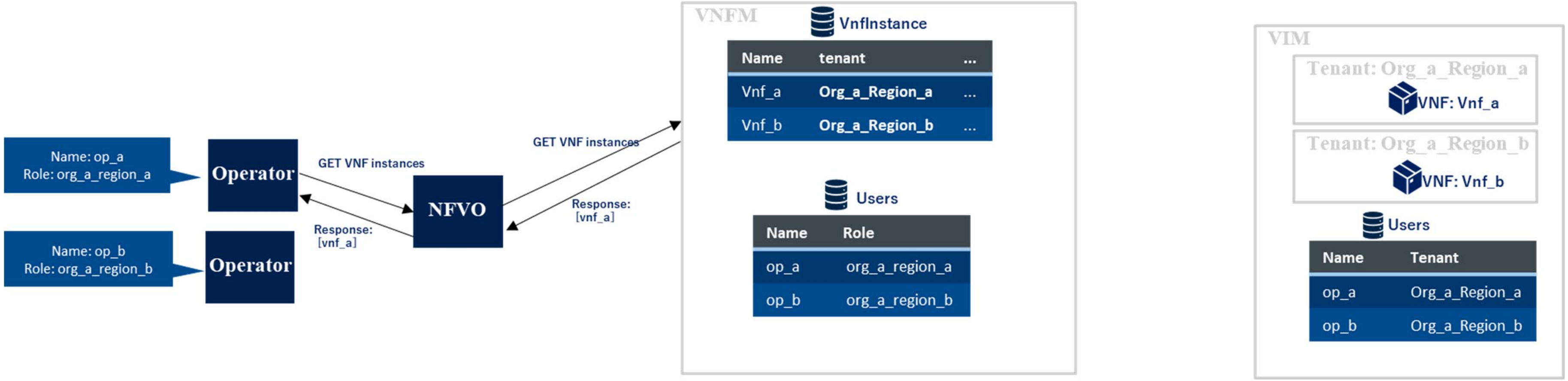
- **extensions:** Additional VNF-specific attributes that affect the lifecycle management of this VNF instance. ... "All extensions that are allowed for the VNF are declared in the VNFD." ... "The VNFM shall reject requests to write extension attributes that are not declared in the VNFD", see ETSI GS NFV-SOL 003 [i.29].

- **vnfConfigurableProperties:** Configurable properties referred in these attributes are declared in the VNFD. ... "The VNFM shall reject requests to write configurable properties that are not declared in the VNFD", see ETSI GS NFV-SOL 003 [i.29].

Although these attributes are assumed to be configured in VNFD, tenant information is not determined by VNFD.

Suggestion: Define appropriate attributes to store tenant information for VNF instances to determine which VNF a user can access.

Figure A.1.2-3 illustrates the use of attributes to convey tenant information.



Name	tenant	...
Vnf_a	Org_a_Region_a	...
Vnf_b	Org_a_Region_b	...

Name	Role
op_a	org_a_region_a
op_b	org_a_region_b

Name	Tenant
op_a	Org_a_Region_a
op_b	Org_a_Region_b

Figure A.1.2-3: Illustrate use of attributes for tenant information

NOTE: Key issue #2 provides several aspects for solving the issue illustrated by Tacker, see clause 6.2:

- As part of solution #2.1, additional attributes are proposed.
- As part of solution #2.3, it is proposed to define ownership for VNFs.
- As part of solution #2.2, operators need to authorize themselves to access a VNF.
  The response of GET VNF Instances operation in Figure A.1.2-3 then only contains VNFs the operator A is authorized to read and subscribe to notifications.

A.2 Multi-Tenancy in Kubernetes®

A.2.1 Introduction

Kubernetes® does not have direct Use Cases and concepts for tenants per se. However, it can still provide indirect support for multi-tenancy with the help of its native features and concepts, such as, namespaces, Role Based Access Control (RBAC), resource quotas, etc.

Multi-tenancy concepts in Kubernetes® can be categorized into two categories: "soft" multi-tenancy, requiring weak isolation, and "hard" multi-tenancy requiring strong isolation. These are rather abstract terms hinting at the level of trust and isolation between the tenants sharing the Kubernetes® cluster. Use cases where tenants do not trust each other or require stronger isolation, e.g. multiple customers sharing a Kubernetes® cluster, multiple NMTs sharing the same NFVI layer, etc. require hard multi-tenancy. In more extreme cases, another option can be to use separate clusters for each tenant.

NOTE: For more details on multi-tenancy support in Kubernetes®, refer to the official Kubernetes® documentation on the topic [i.41].

A.2.2 Isolation considerations for tenants

A.2.2.1 Same cluster

A.2.2.1.1 General

Kubernetes® offers isolation of various forms for tenants sharing the same Kubernetes® cluster. For stronger isolation requirements, different clusters for each tenant can be used.

There can be two types of isolations considered within the same Kubernetes® cluster, namely 'Control Plane isolation' and 'Data Plane isolation'.

NOTE: For more details on different Kubernetes® objects mentioned in the following clauses, refer to the official Kubernetes® documentation [i.42].

A.2.2.1.2 Control plane isolation

Isolation on the Control plane level refers to isolating Kubernetes® resources between tenants, such as Deployments, Services, Config Maps, etc. This is done by means of creating separate namespaces for each tenant. Furthermore, Kubernetes® objects with namespace scope, such as Roles and RoleBindings can be used to implement RBAC and ensure that tenants cannot access or modify Kubernetes® resources belonging to other tenants. Resource quotas, namespace-scoped objects, can ensure that tenants do not exceed their allocated share of cluster resources, like compute, memory, storage, etc. thereby minimizing the 'noisy neighbour' effect.

An alternate to namespace-based isolation is the concept of virtual control plane per each tenant. This approach further isolates the cluster's Kubernetes® API server by assigning each tenant their dedicated control plane components, e.g. Kubernetes® API server, etc. store and controller manager. The Kubernetes® Cluster API project [i.43] is among some implementations providing this kind of isolation. Another implementation, vCluster [i.44] creates virtual Kubernetes® clusters on top of the underlying host cluster to offer virtual control plane for each tenant.

A.2.2.1.3 Data plane isolation

Data plane isolation refers to isolating tenants' pods and workloads running on the shared Kubernetes® cluster. By configuring appropriate Network Policies, isolation on network communication level can be ensured between pods belonging to different tenants. For storage isolation, separate StorageClass per tenant can be used to assign PersistentVolumes (a cluster wide object by design) to individual tenants claiming storage resources using their PersistentVolumeClaims (namespace-scoped object). 'Container sandboxing' concept can be used to further isolate pods belonging to different tenants that are sharing the same host OS. This refers to running containers inside a VM or userspace kernel to avoid any security issues like, container breakouts. Node-based isolation can be another technique to isolate data plane workloads among multiple tenants. A node or a set of nodes can be dedicated to tenants for running their workloads/pods in a shared Kubernetes® cluster.

NOTE: Implementation of isolation rules specified in Network policies is dependent on the underlying Container Networking Interface (CNI) plugin being able to support the implementation of network policies.

A.2.2.2 Multiple clusters

To avoid potential security issues and offer strict isolation among tenants, multi-cluster solutions can be considered in Kubernetes®, with each tenant having their own cluster. Dedicated hardware can also be considered for each tenant to offer increased security and isolation. Kubernetes® Cluster API [i.43] can be used to provision and manage clusters for different tenants.

A.2.3 Relation with multi-tenancy in NFV

NFV-MANO multi-tenancy Use Cases related to containerized workloads (e.g. Use Case #8) are described in clause 5 of the present document.

In NFV-MANO framework, CISM is the entity responsible for managing containerized workloads and can be partially mapped to Kubernetes®. For example, OS container management service provided by CISM function can be profiled to Kubernetes® API [i.45]. Additionally, CCM is the entity responsible for management of CIS clusters and its service interfaces can be profiled to Kubernetes® Cluster API [i.43].

Use case #8, described in clause 5 of the present document, deals with deployment of containerized workloads belonging to different NMTs, with varying isolation needs. See the relevant Use Case for more details.

Table A.2.3-1 provides the mapping between different deployment scenarios for containerized VNFs in NFV with varying levels of isolation and their corresponding types of isolation supported in Kubernetes®.

Table A.2.3-1: Mapping between NFV deployment scenarios and the corresponding type of isolation supported in Kubernetes®

NFV multi-tenancy deployment scenario	Type of isolation in Kubernetes®	Cluster	Description
Deployment of containerized VNFs with namespace isolation	Control Plane isolation (namespace based)	Same cluster	CISM is made aware of the affinity and anti-affinity levels that can be applied within its scope of responsibility (within CIS cluster) and can apply affinity and anti-affinity on the level of namespaces.
Deployment of containerized VNFs with isolation based on CIS cluster nodes	Data Plane isolation (node level)	Same cluster	CISM is made aware of the affinity and anti-affinity levels that can be applied within its scope of responsibility (within CIS cluster) and can apply affinity and anti-affinity on the level of CIS cluster nodes.
Deployment of containerized VNFs with isolation based on both namespaces and CIS cluster nodes	Control Plane isolation (namespace based) Data Plane isolation (node level)	Same cluster	CISM is made aware of the affinity and anti-affinity levels that can be applied within its scope of responsibility (within CIS cluster) and can apply affinity and anti-affinity on the level of namespaces and CIS cluster nodes.
Deployment of containerized VNFs with isolation based on CIS clusters	Multi-cluster isolation	Multiple clusters	NFVO selects the right CIS cluster based on affinity/anti-affinity constraints for CIS cluster level or NFVI related levels in the NSD and VNFD during the instantiation workflow. CCM can be used to manage multiple CIS clusters.

A.3 Multi-Tenancy in Anuket

Linux® Foundation Anuket project [i.37] provides reference model and reference architectures for virtualized and cloud native network functions, which specify requirements with respect to multi-tenancy.

The principles and general requirements as found in the Reference Model for Cloud Infrastructure (RM) [i.38] specify requirements about isolation and independent management including details describing isolation of compute, storage or network resources and providing information on shared storage and network for multiple tenants. Acceleration infrastructure supporting tenant specific programming is also considered. Anuket sets requirements on tenant management, isolation, security as well as the management of resource quota per tenant.

The Reference Architecture for OpenStack based cloud infrastructure (RA1) [i.39] describes in detail how above requirements can be fulfilled in an OpenStack based system. In the same way, the Reference Architecture for Kubernetes® based cloud infrastructure (RA2) [i.40] describes the high-level system components and their interactions, taking the goals and requirements and mapping them to Kubernetes (and related) components. As Kubernetes® does not support strict isolation RA2 recommends to use separate Kubernetes Clusters for the deployment where hard multi-tenancy requirements apply. The use of namespaces for multi-tenancy is only recommended in cases where there are no hard multi-tenancy requirements (multiple development teams in the same organization).

Annex B: Change History

Date	Version	Information about changes
June 2019	0.0.1	Provide initial skeleton
October 2019	0.0.2	NFVEVE(19)000071r2 EVE018 Initial text for overview NFVEVE(19)000073r4 EVE018 Draft first Use Case NFVEVE(19)000084r1 EVE018 Use Case Two users with own NFVO on shared NFVI Remove list of contributors and other editorial changes
December 2019	0.0.3	NFVEVE(19)000076r3 EVE018 Draft first Use Case: Description Related editorial changes NFVEVE(19)000104r2 EVE018 Use case on network slice NFVEVE(19)000109r2 EVE018: Use case on nested NS
February 2020	0.0.4	NFVEVE(20)000004 EVE018 Some fixes for consistency NFVEVE(20)000005 EVE018 Variant of UC#1 NFVEVE(20)000006r2 EVE018 Details of UC#1 NFVEVE(20)000007r1 EVE018 Some addition to UC#4 NFVEVE(20)000011 EVE018 Use Case from 3GPP Multitenancy Study
March 2020	0.0.5	NFVEVE(20)000024r1 - EVE018 – Use Case on Tenant Operated MANO System NFVEVE(20)000039 - EVE018 Align UC#2 to UC#1 NFVEVE(20)000040 - EVE018 Add resource groups for VNFM in UC2
April 2020	0.0.6	NFVEVE(20)000056 - EVE018 Variants of UC#1 and UC#2 with Multiple Sites NFVEVE(29)000057 - EVE018 Variant of UC#4 with nested and other NS by same tenant NFVEVE(20)000058 - EVE018 Analysis clause for UC#1
May 2020	0.0.7	NFVEVE(20)000067 EVE018 Analysis clause for UC#2 NFVEVE(20)000068 EVE018 Detailed User Story for UC#3 NFVEVE(20)000075 EVE018 Variant for UC#3 network slice subnet with multiple NS NFVEVE(20)000076 EVE018 Analysis of UC#3
September 2020	0.0.8	NFVEVE(20)000087 EVE018 Use case Different levels of isolation NFVEVE(20)000127r1 EVE018 Detailed User Story UC#4 Nested NS NFVEVE(20)000128r1 EVE018 Analysis for UC#4 Nested NS NFVEVE(20)000131r1 EVE018 Align to improvements in UC#4 NFVEVE(20)000132 EVE018 Add key issues to skeleton NFVEVE(20)000133 EVE018 Add Use Case to share entities between SPs
November 2020	0.0.9	NFVEVE(20)000152 EVE018 Remove some Editors Notes NFVEVE(20)000153 EVE018 Improve Overview NFVEVE(20)000154 EVE018 Motivation for Use Case #8 containerized VNFs NFVEVE(20)000155 EVE018 Details and analysis for Use Case #5 Some editorial corrections
December 2020	0.0.10	NFVEVE(20)000168r1 EVE018 Start details on Use Case #7 NFVEVE(20)000170 EVE018 Clause 5.7.2.2 Usage of anti-affinity groups NFVEVE(20)000171 EVE018 Clause 6.1 Details on Resource Groups
May 2021	0.0.11	NFVEVE(21)000037 EVE018 Clause 5.7.2.3 Flow for Use Case #7 NFVEVE(21)000038 EVE018 Some alignments NFVEVE(21)000039 EVE018 Add more key issues to skeleton
July 2021	0.0.12	NFVEVE(21)000050r2 EVE018 Tenant Identification NFVEVE(21)000055 EVE018 Annex on Multi-Tenancy in OpenStack NFVEVE(21)000056 EVE018 Scope and editorial NFVEVE(21)000059 EVE018 More content for Key Issue #2 Some rapporteur's actions
July 2021	0.0.13	NFVEVE(21)000068r3 EVE018 Correct clause 5.7.2.2 Usage of anti-affinity groups NFVEVE(21)000070r1 EVE018 Clause 5.7.3 Variants of Use Case #7 NFVEVE(21)000071 EVE018 Clause 5.7.3 Analysis for Use Case #7
October 2021	0.0.14	NFVEVE(21)000105r3 EVE018 Clause 4 Overview additions NFVEVE(21)000106r1 EVE018 Clause 5 Introduction to Use Cases NFVEVE(21)000112r1 EVE018 Clause 5.5.3 Add Variant of ENISA Rapporteur's actions to renumber clauses 5.x
December 2021	0.0.15	NFVEVE(21)000135r1 EVE018 Fix normative language NFVEVE(21)000113r1 EVE018 More content for Use Case #9 NFVEVE(21)000079 EVE018 - Use Case 6 Summary
January 2022	0.0.16	NFVEVE(21)000136r2 EVE018 Clause 5.10.2 Details for Use Case #9 shared entities NFVEVE(21)000143r1 EVE018 a few small changes

Date	Version	Information about changes
March 2022	0.0.17	NFVEVE(21)000117r4 EVE018 - Use Case 6 Summary Extension and Abbreviations NFVEVE(22)000021r1 EVE018 Clause 6.1 Describe Key Issue #1 Some editorial corrections (numbering)
July 2022	0.0.18	NFVEVE(22)000080 EVE018 Add note on kata containers NFVEVE(22)000081r2 EVE018 Add content to Use Case #8 NFVEVE(22)000123 EVE018 Correct Diagram 5.9.1-1 NFVEVE(22)000124r1 EVE018 Structure for Use Case 10
October 2022	0.0.19	NFVEVE(21)000096r6 EVE018 - Use Case 6 Actors, Pre-Post Conditions and Description Editorial Changes: Formatting, numbering of notes.
October 2022	0.0.20	NFVEVE(22)000184r1 EVE018 Editors Note in clause 5.9.3 NFVEVE(22)000185 EVE018 Add note to clause 5.4.3.1 NFVEVE(22)000186r1 EVE018 Editors note in clause 5.3 NFVEVE(22)000187 EVE018 Editors note and clarification in clause 5.4.1 NFVEVE(22)000188 EVE018 Editors note in clause 5.10.3.2.3
January 2023	0.0.21	NFVEVE(22)000223r3 EVE018 – Use Case 6 MANO-T LCM Operation Description NFVEVE(23)000014r1 EVE018 Use case 9 Motivation Sharing Packages
March 2023	0.0.22	NFVEVE(23)000017 Use case 9 add special case on shared storage, approved. NFVEVE(23)000018 Use case 9 add details on sharing NSD or package, approved NFVEVE(23)000019r2 Use case 9 add clarifications in clause 5.10.3.1, Expect a revision. approved NFVEVE(23)000048r1 EVE018 - Use Case 6 MANO-T LCM Operation Description Outside MLA Bounds, approved NFVEVE(23)000049 EVE018 - Use Case 6: Editorial edits and Removing ENs, approved
August 2023	0.0.23	NFVEVE(23)000139r1 EVE018 update clause 6.1 key issue#1 NFVEVE(23)000140r1 EVE018 update clause 4 overview NFVEVE(23)000144r3 EVE018 updates for Use Case #9 (sharing) NFVEVE(23)000160r1 EVE018 Solution 1&2 in key issue #1
September 2023	0.0.24	NFVEVE(23)000145r1 EVE018 Include Tacker feedback to Annex NFVEVE(23)000161r2 EVE018 Solution 1.3 in key issue #1 NFVEVE(23)000168r1 EVE018 - UC#6 MANO-T Deployment Variant NFVEVE(23)000169r3 EVE018 - UC#6 Analysis NFVEVE(23)000170r3 EVE018 key issue #2
October 2023	0.0.25	NFVEVE(23)000193r2 EVE018 Solution 2.2 on authorization NFVEVE(23)000194 EVE018 Change recommendations from Use Cases to reference clause 6 NFVEVE(23)000195 EVE018 update references from Use Cases to key issues NFVEVE(23)000196r1 EVE018 Resolve Editors notes and fix references in Use Case #8 NFVEVE(23)000197 EVE018 Resolve Editors note in clause 5.10 NFVEVE(23)000200r2 EVE018 Annex on Multi-Tenancy in Anuket NFVEVE(23)000201r2 EVE018 Reduce Use Case #9 NFVEVE(23)000202 EVE018 Correct references and resolve Editors Note in Use Case #7
November 2023	0.0.26	NFVEVE(23)000203r2 EVE018 Clause 6.2 More on tenant management NFVEVE(23)000204 EVE018 Two small additions NFVEVE(23)000205r2 EVE018 Add two key issues NFVEVE(23)000208 EVE018 Resolve Editors Notes in clause 5.10.1
November 2023	0.0.27	NFVEVE(23)000209 EVE018 Remove three Editors Notes NFVEVE(23)000210r3 EVE018 Multi-tenancy in Kubernetes NFVEVE(23)000214r1 EVE018 several small changes and corrections NFVEVE(23)000215 EVE018 Remaining Editors notes in clause 6.1
November 2023	0.0.28	NFVEVE(23)000220r1 EVE018 Editors note on shared storage NFVEVE(23)000221 EVE018 Resolve Editors note on license management NFVEVE(23)000222 EVE018 Three changes

Date	Version	Information about changes
November 2023	0.0.29	NFVEVE(23)000216r2 EVE018 Key issue #3 on sharing VNF package NFVEVE(23)000226r1 EVE018 some more small topics NFVEVE(23)000227 EVE018 Add key issue on quota management NFVEVE(23)000230 EVE018 Feedback from SEC
December 2023	0.0.30	NFVEVE(23)000233r2 EVE018 More on VNF package onboarding in key issue #3 NFVEVE(23)000234 EVE018 add a missing recommendation NFVEVE(23)000235r2 EVE018 other solution proposals for key issue#3
January 2024	0.0.31	NFVEVE(24)000001r1 EVE018 Update figures after discussion with SEC NFVEVE(24)000002r1 EVE018 Remaining changes
February 2024	0.1.0	NFVEVE(24)000003r5 EVE018 - UC#6 Clause 6 Key issue description and solution proposals NFVEVE(24)000010r1 EVE018 Multiple clauses PaaS Services sharing NFVEVE(24)000015r3 EVE018 - UC#6 - Recommendations on MLA management NFVEVE(24)000020r1 EVE018 Conclusion clauses and final editorial changes NFVEVE(24)000021r1 EVE018 - UC#6 Editorial Inputs
March 2024	0.2.0	NFVEVE(24)000049r1 EVE018 Editorial review NFVEVE(24)000050r1 EVE018 Multiple clauses Technical review NFVEVE(24)000053r1 EVE018 Address comments on references clause

History

Document history		
V5.1.1	May 2024	Publication