ETSI GR NFV-EVE 019 V5.1.1 (2023-10)

Network Functions Virtualisation (NFV) Release 5; Architectural Framework; Report on VNF generic OAM functions

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
RGR/NFV-EVE019ed511

Keywords
architecture, automation, configuration,
management, MANO, NFV, OAM

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

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 analyses and defines the type of OAM functions for VNFs that can be generalized and be provided as a "generic function" supporting e.g. the provisioning, connectivity, configuration and monitoring of VNFs on a virtualised platform. The present document also describes possible solutions to realize such generic OAM functions.

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.

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 not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI GR NFV 003: "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

[i.2] ETSI GR NFV-IFA 029: "Network Functions Virtualisation (NFV) Release 3; Architecture; Report on the Enhancements of the NFV architecture towards "Cloud-native" and "PaaS"".

[i.3] ETSI GS NFV-IFA 027: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Performance Measurements Specification".

[i.4] ETSI GS NFV-SOL 016: "Network Functions Virtualisation (NFV) Release 3; Protocols and Data Models; NFV-MANO procedures specification".

[i.5] ETSI GS NFV-IFA 008: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Ve-Vnfm reference point - Interface and Information Model Specification".

[i.6] ETSI GS NFV-IFA 031: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Requirements and interfaces specification for management of NFV-MANO".

[i.7] ETSI GS NFV-IFA 010: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Functional requirements specification".

[i.8] ETSI GS NFV-IFA 009: "Network Functions Virtualisation (NFV); Management and Orchestration; Report on Architectural Options".

[i.9] 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.10] ETSI GS NFV-IFA 007: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Or-Vnfm reference point - Interface and Information Model Specification".

[i.11] ETSI GR NFV-IFA 041 (V4.1.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Report on enabling autonomous management in NFV-MANO".

[i.12] ETSI GR NFV-IFA 037 (V4.1.1): "Network Functions Virtualisation (NFV) Release 4; Architectural Framework; Report on further NFV support for 5G".

[i.13] ETSI GR NFV-IFA 046 (V5.1.1): "Network Functions Virtualisation (NFV) Release 5; Architectural Framework; Report on NFV support for virtualisation of RAN".

[i.14] ETSI TS 123 501: "3GPP; TSG Services and System Aspects; System architecture for the 5G System (5GS); Stage 2 (Release 17)".

[i.15] ETSI GS NFV-IFA 011 (V4.4.2): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; VNF Descriptor and Packaging Specification".

[i.16] ETSI GS NFV-IFA 047 (V4.4.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Management data analytics Service Interface and Information Model specification".

[i.17] ETSI GS NFV-IFA 048 (V4.4.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Policy Information Model Specification".

[i.18] ETSI GS NFV-IFA 050 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Intent Management Service Interface and Intent Information Model Specification".

[i.19] ETSI TS 123 288: "5G; Architecture enhancements for 5G System (5GS) to support network data analytics services (Release 17).

[i.20] ETSI TS 128 104: "5G; Management and orchestration; Management Data Analytics (MDA) (Release 17).

[i.21] ETSI GR NFV-IFA 023 (V3.1.1): "Network Functions Virtualisation (NFV); Management and Orchestration; Report on Policy Management in MANO; Release 3".

[i.22] ETSI GS NFV-SOL 012 (V4.3.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; RESTful protocols specification for the Policy Management Interface".

[i.23] ETSI GR NFV-IFA 028 (V3.1.1): "Network Functions Virtualisation (NFV) Release 3; Management and Orchestration;Report on architecture options to support multiple administrative domains".

[i.24] ETSI GS NFV-SOL 005 (V4.3.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; RESTful protocols specification for the Os-Ma-nfvo Reference Point".

[i.25] ETSI GS NFV-IFA 049 (V4.4.1): "Network Functions Virtualisation (NFV) Release 4; Architectural Framework; VNF generic OAM functions specification".

[i.26] ETSI GR NFV-IFA 035: "Network Functions Virtualisation (NFV) Release 5; Architectural Framework; Report on network connectivity integration and operationalization for NFV".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GR NFV 003 [i.1] and the following apply:

VNF generic OAM function: function that provides in a generic form OAM capabilities applicable to any kind of VNFs, NFV-MANO FBs and functions and NFVI

NOTE 1: These functions aim at easing the provisioning, connectivity, configuration and monitoring of one or more entities (e.g.VNFs).

NOTE 2: The kinds of VNF concern to diverse VNF implementation approaches and diverse network functionality and services provided by the VNFs.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GR NFV 003 [i.1] and the following apply:

CE	Customer Edge
DNS	Domain Name System
L2VPN	Layer 2 VPN
NTP	Network Time Protocol
NWDAF	Network Data Analytics Function
MDA	Management Data Analytics
MDAF	Management Data Analytics Function
PAP	Policy Administration Point
PE	Provider Edge
PF	Policy Function
PIM	Physical Infrastructure Manager

4 Use Cases

4.1 Overview

This clause provides a list of use cases related to functionality that would benefit from VNF generic OAM functions. The use cases are grouped into two categories, namely use cases related to the lifecycle of VNF generic OAM functions and use cases related to type of VNF generic OAM functions. In all the use cases, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine.

4.2 Use cases related to LCM of VNF generic OAM functions

4.2.1 Use case: Instantiation of VNFs using VNF generic OAM functions managed by NFV-MANO

4.2.1.1 Introduction

This use case is about the instantiation of VNFs that their implementation uses one or more VNF generic OAM functions. That way, the Operator is provided with a more generic way to configure, manage/operate, and monitor the VNFs it operates. Also, the management of the VNFs will be more focused on services management, while some of the VNF generic OAM functions will deal with the underlying resources, host, and network. In this use case it is assumed that NFV-MANO is responsible to manage the lifecycle of the VNF generic OAM functions, e.g. instantiate a VNF generic OAM function when required by a VNF instance or terminate a VNF generic OAM function when it is no longer being used by any VNF instance. It is also assumed that the VNF generic OAM function can be consumed by multiple consumer instances at the same time.

During the instantiation of the VNFs, NFV-MANO needs to check the availability of the VNF generic OAM functions.

Different options are possible (not an exhaustive list):

a) The required VNF generic OAM function supports all required functionalities. It is already instantiated in the system, can be shared and is operational.

b) The VNF generic OAM function supports all required functionalities, is not yet instantiated in the system or the instantiated function cannot be shared.

c) The VNF generic OAM function is available but some all of the required functionalities are missing and the VNF instantiation will fail.

The user story related to this use case is the following:

> An Operator can instantiate VNFs that are using VNF generic OAM functions and has a generic way to configure, manage/operate, and monitor the different network functions in the Operator's environment and can focus on "services management".

4.2.1.2 Actors and roles

Table 4.2.1.2-1 describes the use case actors and roles.

Table 4.2.1.2-1: Instantiation of VNFs using VNF generic OAM functions managed by NFV-MANO, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system including the NFV-MANO functional entities, the VNFs, the VNF generic OAM functions, and the underlying infrastructure.
2	OSS/BSS	The entity that receives request from the Operator to instantiate the VNF.
3	NFV-MANO	The entity instantiating and managing the VNF.

4.2.1.3 Trigger

Table 4.2.1.3-1 describes the use case trigger.

Table 4.2.1.3-1: Instantiation of VNFs using VNF generic OAM functions managed by NFV-MANO, trigger

Trigger	Description
Operator is requesting the instantiation of the VNF.	

4.2.1.4 Pre-conditions

Table 4.2.1.4-1 describes the pre-conditions of this use case.

Table 4.2.1.4-1: Instantiation of VNFs using VNF generic OAM functions managed by NFV-MANO, pre-conditions

#	Pre-condition	Description
1	NFV-MANO environment is operational	
2	VNF Package is onboarded	The VNF implementation supports the use of one or more VNF generic OAM functions.
3	VNF generic OAM functions are available	All VNF generic OAM functions required by the VNF are available. Some of them may already be instantiated and are operational.
4	NFV-MANO knows which VNF generic OAM functions are required by the VNF and need to be instantiated	

4.2.1.5 Post-conditions

Table 4.2.1.5-1 describes the post-conditions of this use case.

Table 4.2.1.5-1: Instantiation of VNFs using VNF generic OAM functions managed by NFV-MANO, post-conditions

#	Post-condition	Description
1	VNF is instantiated	VNF is instantiated and is using VNF generic OAM functions.

4.2.1.6 Operational Flows

Table 4.2.1.6-1 describes the base flow of this use case.

Table 4.2.1.6-1: Instantiation of VNFs using VNF generic OAM functions managed by NFV-MANO, base flow

#	Actor/Role	Description
Begins When	Operator -> OSS/BSS -> NFV-MANO	Operator is triggering the instantiation of the VNF to the NFVO via OSS/BSS.
1	NFV-MANO	NFV-MANO is checking the availability of the VNF generic OAM functions required to run the VNF function.
2	NFV-MANO	If a VNF generic OAM function is not yet instantiated, the VNF generic OAM function will be instantiated by NFV-MANO.
3	NFV-MANO	Once all required VNF generic OAM functions are instantiated and ready to be used, NFV-MANO will instantiate the requested VNF. A detailed flow for the "Instantiate VNF instance" is described as part of the "NS instantiate procedure" in clause 5.2 of ETSI GS NFV-SOL 016 [i.4].
Ends When	NFV-MANO -> OSS/BSS -> Operator	NFV-MANO returns an "operation completed" notification to the Operator via OSS/BSS.

4.2.2 Use case: Termination of VNF instances using VNF generic OAM functions managed by NFV-MANO

4.2.2.1 Introduction

This use case is about the termination of VNF instances that their implementation uses one or more VNF generic OAM functions. In this use case it is assumed that NFV-MANO is responsible to manage the lifecycle of the VNF generic OAM functions, e.g. terminate or scale in a VNF generic OAM function when it is no longer being used by one or more VNF instances.

The user stories related to this use case are:

- An Operator can terminate the VNF instances that are using VNF generic OAM functions, so that the virtualised resources associated to the VNFs and associated VNF generic OAM functions can be released.
- An Operator can terminate the VNF instances without having to handle the lifecycle of the VNF generic OAM functions.

4.2.2.2 Actors and roles

Table 4.2.2.2-1 describes the use case actors and roles.

Table 4.2.2.2-1: Termination of VNF instances using VNF generic OAM functions managed by NFV-MANO, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system including the NFV-MANO functional entities, the VNFs, the VNF generic OAM functions, and the underlying infrastructure.
2	OSS/BSS	The entity that receives request from the Operator to terminate the VNF.
3	NFV-MANO	The entity managing and terminating the VNF instance.

4.2.2.3 Trigger

Table 4.2.2.3-1 describes the use case trigger.

Table 4.2.2.3-1: Termination of VNF instances using VNF generic OAM functions managed by NFV-MANO, trigger

Trigger	Description
Operator is requesting the termination of the VNF instance.	

4.2.2.4 Pre-conditions

Table 4.2.2.4-1 describes the pre-conditions of this use case.

Table 4.2.2.4-1: Termination of VNF instances using VNF generic OAM functions managed by NFV-MANO, pre-conditions

#	Pre-condition	Description
1	NFV-MANO environment is operational.	
2	NFV-MANO knows which lifecycle operations are required to be performed on the VNF generic OAM functions (e.g. terminate, scale in) after terminating the VNF instance.	

4.2.2.5 Post-conditions

Table 4.2.2.5-1 describes the post-conditions of this use case.

Table 4.2.2.5-1: Termination of VNF instances using VNF generic OAM functions managed by NFV-MANO, post-conditions

#	Post-condition	Description
1	VNF instance is terminated.	
2	Lifecycle operations on the VNF generic OAM functions (e.g. terminate, scale in) are successfully completed.	

4.2.2.6 Operational Flows

Table 4.2.2.6-1 describes the base flow of this use case.

Table 4.2.2.6-1: Termination of VNF instances using VNF generic OAM functions managed by NFV-MANO, base flow

#	Actor/Role	Description
Begins When	Operator -> OSS/BSS -> NFV-MANO	Operator is triggering the termination of the VNF instance to the NFVO via OSS/BSS.
1	NFV-MANO	NFV-MANO terminates the VNF instance. A detailed flow for the "Terminate VNF instance" is described as part of the "NS termination procedure" in clause 5.3 of ETSI GS NFV-SOL 016 [i.4].
2	NFV-MANO	Once the termination of the VNF instance has been successfully executed, NFV-MANO is performing the lifecycle operations on the VNF generic OAM functions (e.g. terminate, scale in). See note.
Ends When	NFV-MANO -> OSS/BSS -> Operator	NFV-MANO returns an "operation completed" notification to the Operator via OSS/BSS.
NOTE: The allowed lifecycle operations depend on the type of the VNF generic OAM function. For example, the Log aggregator function as described in clause 4.3.1 below cannot be terminated. The reason is that this function might stay available for keeping log files accessible for consumption by users such as the Operator.		

4.2.3 Use case: Lifecycle management of VNF generic OAM function managed by NFV-MANO

4.2.3.1 Introduction

The goal of this use case is to describe a generic use case "LCM of VNF generic OAM function managed by NFV-MANO". The use case shows a complete lifecycle management of a VNF generic OAM function that is instantiated, scaled, and terminated based on the demand of an exemplary NS.

The user story related to this use case is:

- An Operator can deploy VNFs that are using VNF generic OAM functions, and has a generic way to configure, manage/operate, and monitor the different network functions in the Operator's environment and can focus on "services management".

The following clauses describe the use case of "LCM of VNF generic OAM function managed by NFV-MANO" which is related to above user story. This use case assumes that the Operator is managing the VNF generic OAM function using existing mechanisms of NFV-MANO for the lifecycle management of these functions. In addition, for the purpose of the present use case, the VNF generic OAM function is assumed to be scalable.

4.2.3.2 Actors and roles

Table 4.2.3.2-1 describes the use case actors and roles.

Table 4.2.3.2-1: Lifecycle management of VNF generic OAM function managed by NFV-MANO, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system including the NFV-MANO functional entities, the VNFs, the VNF generic OAM functions, and the underlying infrastructure.
2	OSS/BSS	The entity that receives request from the Operator to perform various NS LCM operations.
3	VNF A, VNF B	Instances of two different VNFs A and B that both require the same VNF generic OAM function.
4	NS A	In this simplified use case, NS A consists of VNF A and VNF B.
5	VNF generic OAM function	A VNF generic OAM function.
6	NFV-MANO	Management and network orchestration framework including NFVO, VNFM(s), and VIM.

4.2.3.3 Trigger

Table 4.2.3.3-1 describes the use case trigger.

Table 4.2.3.3-1: Lifecycle management of VNF generic OAM function managed by NFV-MANO, trigger

Trigger	Description
Operator is deploying NS A that is requiring the VNF generic OAM function.	

4.2.3.4 Pre-conditions

Table 4.2.3.4-1 describes the pre-conditions of this use case.

Table 4.2.3.4-1: Lifecycle management of VNF generic OAM function managed by NFV-MANO, pre-conditions

#	Pre-condition	Description
1	NS A Descriptor and VNF Packages of the VNFs A and B to be instantiated are on-boarded.	
2	The VNF generic OAM function is ready to be instantiated.	
3	The VNF generic OAM function supports multiple consumer instances at the same time.	

4.2.3.5 Post-conditions

Table 4.2.3.5-1 describes the post-conditions of this use case.

Table 4.2.3.5-1: Lifecycle management of VNF generic OAM function managed by NFV-MANO, post-conditions

#	Post-condition	Description
1	NS A was successfully instantiated, then scaled and at the end of the use case terminated.	
2	The VNF generic OAM function was instantiated and if not needed anymore e.g. by other NSs, terminated. See note.	
NOTE: The allowed lifecycle operations depend on the type of the VNF generic OAM function. For example, the Log aggregator function as described in clause 4.3.1 below cannot be terminated because this function might stay available for keeping log files accessible for consumption by users such as the Operator.		

4.2.3.6 Operational Flows

Table 4.2.3.6-1 describes the base flow of this use case.

Table 4.2.3.6-1: Lifecycle management of VNF generic OAM function managed by NFV-MANO, base flow

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS -> NFV-MANO	The Operator requests the instantiation of NS A via OSS/BSS.
1	NFV-MANO	As part of the instantiation process of VNF A (as a constituent of NS A), NFV-MANO determines from the VNF Package of VNF A that VNF A requires the VNF generic OAM function.
2	NFV-MANO	As the VNF generic OAM function is not yet instantiated, the VNF generic OAM function will be instantiated reusing LCM functionality provided by NFV-MANO. Once the required VNF generic OAM function is instantiated and ready to be used, NFV-MANO will instantiate the requested VNF A. This may include start of the health monitoring between the VNF A and the VNF generic OAM function. See also clause 4.2.1.
3	NFV-MANO	As part of the instantiation process of VNF B, NFV-MANO determines from the VNF Package of VNF B that VNF B requires the VNF generic OAM function.
4	NFV-MANO	As the VNF generic OAM function is already instantiated and used, NFV-MANO will instantiate the requested VNF B. This may include changes to some configuration of the VNF generic OAM function and/or start of the health monitoring between the VNF B and the VNF generic OAM function.
5	NFV-MANO -> OSS/BSS -> Operator	NFV-MANO notifies the completion of the instantiation operation to the Operator via OSS/BSS.
6	Operator -> OSS/BSS -> NFV-MANO	The operator requests to scale out the NS A by adding additional instances of VNF B via OSS/BSS.
7	NFV-MANO <-> VNF generic OAM function	NFV-MANO is informing the VNF generic OAM function about the planned scale out of VNF B.

#	Actor/Role	Description
8	NFV-MANO <-> VNF generic OAM function	NFV-MANO instantiates additional instances of VNF B. Due to increased demand the VNF generic OAM function needs to address and based on scaling policies defined in the NS A Descriptor, NFV-MANO performs the scaling up/out of the VNF generic OAM function.
9	NFV-MANO -> OSS/BSS -> Operator	NFV-MANO notifies the completion of the scaling out operation to the Operator via OSS/BSS.
10	Operator -> OSS/BSS -> NFV-MANO	The Operator requests to scale in the NS A by terminating VNF A via OSS/BSS.
11	NFV-MANO <-> VNF generic OAM function	NFV-MANO is informing the VNF generic OAM function about the planned termination of VNF A. This may include changes to some configuration of the VNF generic OAM function and/or stop any health monitoring between the VNF instance(s) to be terminated and the VNF generic OAM function.
12	NFV-MANO <-> VNF generic OAM function	NFV-MANO terminates the VNF A. Due to the decreased demand the VNF generic OAM function needs to address and based on scaling policies defined in the NS A Descriptor, NFV-MANO performs the scale down/in of the VNF generic OAM function. See note.
13	NFV-MANO -> OSS/BSS -> Operator	NFV-MANO notifies the completion of the scaling in operation to the Operator via OSS/BSS.
14	Operator -> OSS/BSS -> NFV-MANO	The Operator requests the termination of NS A via OSS/BSS.
15	NFV-MANO	NFV-MANO is informing the VNF generic OAM function about the planned termination of all instances of VNF B. This may include changes to some configuration of the VNF generic OAM function and/or stop any health monitoring between the VNF instance(s) to be terminated and the VNF generic OAM function.
16	NFV-MANO	NFV-MANO terminates the NS A, including all instances of VNF B. See also clause 4.2.2.
17	NFV-MANO	Once the termination of all instances of VNF B has been successfully executed, NFV-MANO determines whether the VNF generic OAM function is still used by any other VNF instance. If it is not needed anymore, NFV-MANO terminates the VNF generic OAM function.
18	NFV-MANO -> OSS/BSS -> Operator	NFV-MANO notifies the completion of the termination operation to the Operator via OSS/BSS.
Ends when	NFV-MANO	NFV-MANO has terminated all instances of the VNF generic OAM function and all resources that had been associated with the VNF generic OAM function have been freed.
NOTE:	This use case flow does not preclude if a vertical or horizontal scaling of the VNF generic OAM function is applied to compensate for the increased/decreased demand (e.g. number of connected VNF instances) and the use case does not make any assumption that a VNF generic OAM function needs to support both vertical and horizontal scaling.	

4.3 Use cases related to type of VNF generic OAM functions

4.3.1 Use case: Log aggregator function

4.3.1.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine. It is also assumed that the VNFC instances (network,application, etc.) and the underlying infrastructure generate log messages which can be accessed by authorized entities.

The goal of this use case is to describe a generic Log aggregator function used by Operators to retrieve log records that have been forwarded to and stored by this function. For example, this would allow Operators to troubleshoot issues associated to a terminated VNFC instance once the logs have been processed by this Log aggregator function.

User stories related to this use case are (not an exhaustive list):

- An Operator or NFV-MANO can retrieve log records from a Log aggregator function and can troubleshoot issues associated with a terminated VNFC instance.

- An Operator or NFV-MANO can retrieve log records from a Log aggregator function and does not have to collect logs from many different VNF instances and VNF-associated logs from different NFV-MANO functional entities.
- An Operator can manage the log information in a Log aggregator function, e.g. filter the type of logs, select a log level, etc.

Figure 4.3.1.1-1 illustrates one example of the relationship of the different actors involved in this use case.

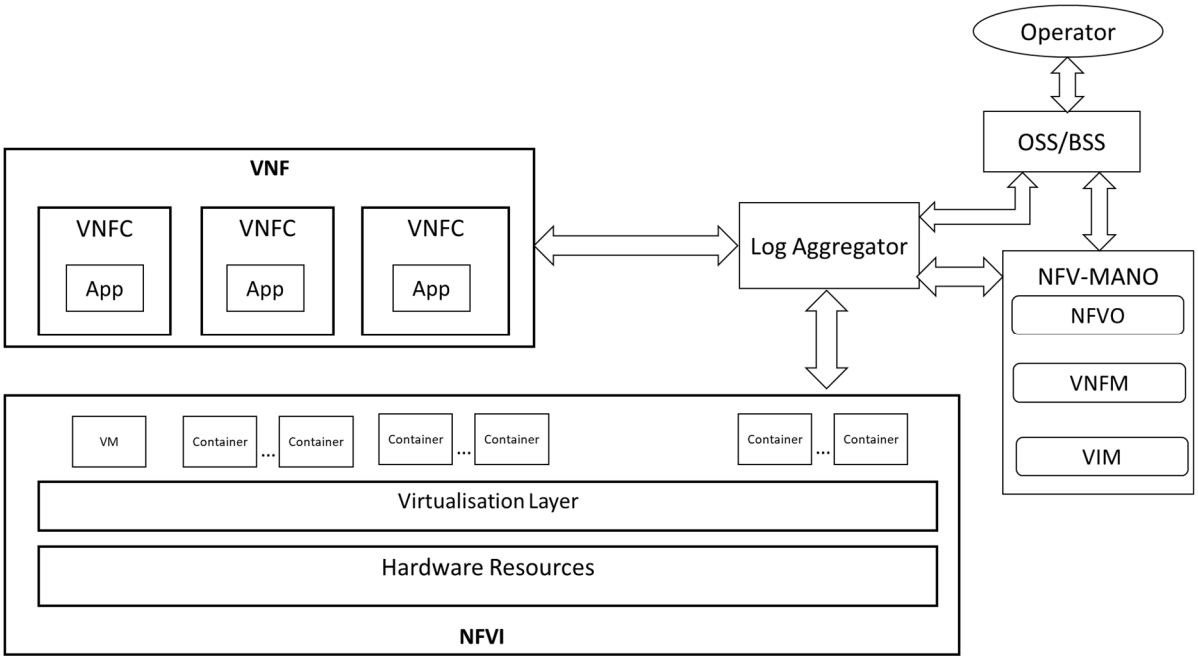

Figure 4.3.1.1-1: Example of relationship between Operator, OSS/BSS, Log aggregator, VNF, NFVI and NFV-MANO

The following clauses describe the use case of "Log aggregator function" which is related to the above user stories. In the following descriptions, references to VNFs/VNFCs logging exposure towards the VNF Log aggregator function implicitly and/or explicitly indicate logging information about the VNF application as well.

4.3.1.2 Actors and roles

Table 4.3.1.2-1 describes the use case actors and roles.

Table 4.3.1.2-1: Log aggregator function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO and the Log aggregator function.
2	OSS/BSS	The entity that receives the request from the Operator to trigger the retrieval of log records.
3	VNF/VNFC/NFVI /NFV-MANO	The entities that provide and forward logs (e.g. application, container, database access logs) related to a VNF to the Log aggregator.
4	Log aggregator	The entity that stores and processes the logs and exposes interfaces towards the Operator.

4.3.1.3 Trigger

Table 4.3.1.3-1 describes the use case trigger.

Table 4.3.1.3-1: Log aggregator function, trigger

Trigger	Description
Operator is requesting via OSS/BSS the retrieval of certain logs associated to a given VNF/VNFC instance (including but not limited to terminated instance).	

4.3.1.4 Pre-conditions

Table 4.3.1.4-1 describes the pre-conditions of this use case.

Table 4.3.1.4-1: Log aggregator function, pre-conditions

#	Pre-condition	Description
1	Log aggregator function is instantiated and configured to collect and store the logs	
2	VNF/VNFC instances, NFVI and NFV-MANO are operational	
3	NFVI, NFV-MANO and or VNF/VNFC instance forward new log entries to the Log aggregator function	

4.3.1.5 Post-conditions

Table 4.3.1.5-1 describes the post-conditions of this use case.

Table 4.3.1.5-1: Log aggregator function, post-conditions

#	Post-condition	Description
1	Operator has all information requested, e.g. to troubleshoot the given VNF instance.	Logging information can be accessed even after the actual entity which had created the Logs is no longer available. The information returned contains only the Logs matching the filter specified in the request.

4.3.1.6 Operational Flows

Table 4.3.1.6-1 describes the base flow of this use case.

Table 4.3.1.6-1: Log aggregator function, base flow

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS -> Log aggregator	The Operator requests the retrieval of the logs to the Log aggregator function via OSS/BSS. The request contains a filter to select the requested logs. The filter can include information about the entities of interest, the type of logs (e.g. performance reports), the requested log level (e.g. the minimum severity in case of logs of type "alarm"), the requested time window (e.g. logs from the last 60 minutes), etc.
1	Log aggregator	The Log aggregator function selects the logs to be returned based on the filter provided in the request message.
Ends when	Log aggregator -> OSS/BSS -> Operator	The Log aggregator function returns the requested information to the Operator via OSS/BSS.

4.3.2 Use case: Log analyser function

4.3.2.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine. It is also assumed that there is a Log aggregator function (see use case in clause 4.3.1) providing logs collected from the VNFC instances (network, application, etc.) and the underlying infrastructure.

The goal of this use case is to describe a generic "Log analyser function" so that Operators can be notified by the Log analyser function when for example a log record matches a given pattern. The Log analyser function aims at analysing any type of log entry and can be also configured to send notifications based on for example statistical processing, or threshold crossing. In addition, due to its system wide view, the Log analyser function can for example identify inconsistency issues that cannot be identified by simply collecting logs from single VNF/VNFC instances.

User stories related to this use case are (not an exhaustive list):

- An Operator or NFV-MANO can get notifications from the Log analyser function, like a threshold being crossed or a log record matches a given pattern and can take appropriate actions where needed.
- An Operator or NFV-MANO can manage the logs analyser function, like configuring the analyser function, setting thresholds and allowing-list/blocklist patterns, configuring the severity level of the notifications, etc.

The following clauses describe the use case of "Log analyser function" which is related to above user stories. In the following descriptions, references to VNFs/VNFCs logging exposure towards the VNF log aggregator function implicitly and/or explicitly indicate logging information about the VNF application as well.

4.3.2.2 Actors and roles

Table 4.3.2.2-1 describes the use case actors and roles.

Table 4.3.2.2-1: Log analyser function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO, the Log aggregator and Log analyser function.
2	OSS/BSS	The entity that receives the request from the Operator to configure the Log analyser function and receive notifications about log analysis resutls.
3	VNF/VNFC/NFVI /NFV-MANO	The entities that provide and forward logs (e.g. application, container, database access logs) related to the VNF to the Log aggregator.
4	Log aggregator	The entity that stores and processes the logs from the VNF/VNFC/NFVI/NFV-MANO and exposes interfaces towards a consumer (e.g. the Log analyser) . See use case in clause 4.3.1.
5	Log analyser	The entity that analyses the logs provided by the Log aggregator and that can notify the Operator through OSS/BSS.
6	Notification manager	The entity that routes notifications (e.g. alerts) sent by client applications to the Operator. See use case in clause 4.3.7.

4.3.2.3 Trigger

Table 4.3.2.3-1 describes the use case trigger.

Table 4.3.2.3-1: Log analyser function, trigger

Trigger	Description
Operator is requesting via OSS/BSS the configuration of the Log analyser function.	

4.3.2.4 Pre-conditions

Table 4.3.2.4-1 describes the pre-conditions of this use case.

Table 4.3.2.4-1: Log analyser function, pre-conditions

#	Pre-condition	Description
1	Log aggregator function is instantiated and configured to collect logs from the VNF/VNFC/NFVI/NFV-MANO instances.	
2	Log entries are available at the Log aggregator function and can be retrieved by the Log analyser function.	
3	Log analyser function is instantiated.	

4.3.2.5 Post-conditions

Table 4.3.2.5-1 describes the post-conditions of this use case.

Table 4.3.2.5-1: Log analyser function, post-conditions

#	Post-condition	Description
1	Operator has received a notification with information related to an event observed by the Log analyser function.	

4.3.2.6 Operational Flows

Table 4.3.2.6-1 describes the base flow of this use case.

Table 4.3.2.6-1: Log analyser function, base flow

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS -> Log analyser	The Operator configures via OSS/BSS the Log analyser to retrieve data from the Log aggregator and on how to process them.
1	Log analyser <-> Log aggregator	The Log analyser retrieves data from the Log aggregator.
2	Log analyser	The Log analyser processes and evaluates the data collected. For example, it detects events like the frequency of a log message pattern crossed a threshold for a given time period.
Ends when	Log analyser -> (Notification manager) -> OSS/BSS -> Operator	The Log analyser issues an alert notification with information related to the event to the Operator via OSS/BSS. Whether this notification is sent directly or via a Notification manager is out of scope of this use case.

4.3.3 Use case: Traffic enforcer function

4.3.3.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine. It is also assumed that the underlying platform provides capabilities to control how incoming and outgoing traffic is transported to the VNFC instances.

The goal of this use case is to describe a generic "traffic enforcer function" so that Operators can perform maintenance tasks on multiple VNFC instances by blocking the traffic for those VNFC instances. This would for example allow Operators to isolate problematic VNFC instances, reroute the traffic to other VNFC instances and avoid further negative service impacts.

User stories related to this use case are (not an exhaustive list):

- An Operator can request the traffic enforcer function to perform the required traffic isolation on problematic VNFC instances and does not have to be directly involved in the actual process.
- An Operator can request the traffic enforcer function to perform the required traffic rerouting of a VNFC instance due to performance issues experienced, so that the load on the VNF instance can be reduced.

Figure 4.3.3.1-1 illustrates one example of the relationship of the different actors involved in this use case whereby external traffic interfaces of VNFC 21 and VNFC 24 instances are being blocked by the Traffic enforcer and traffic is rerouted from VNFC 21 to VNFC 22 instance.

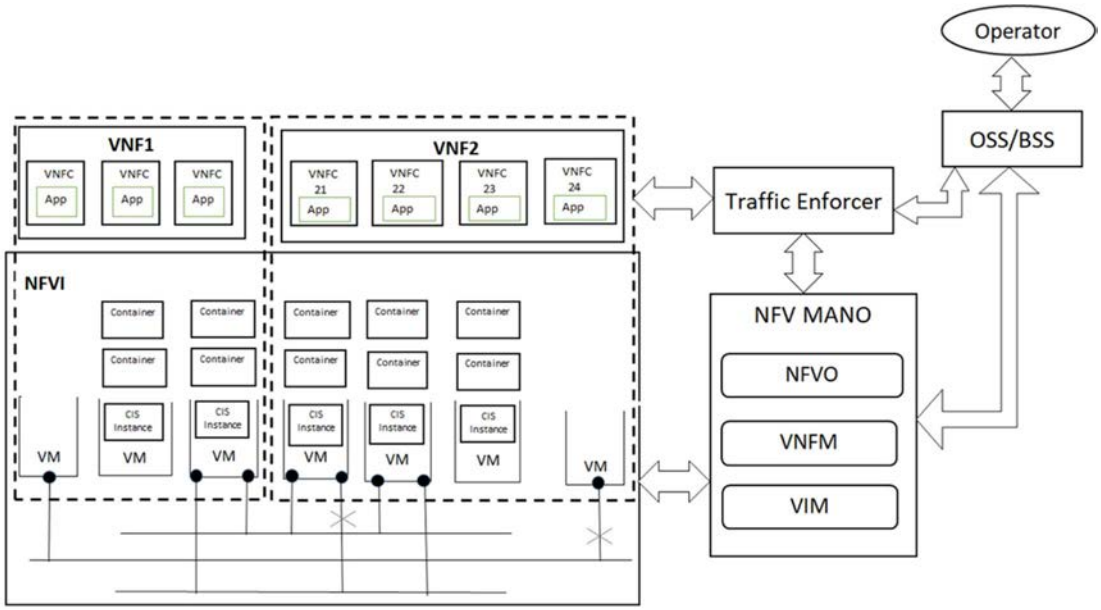

Figure 4.3.3.1-1: Example of relationship between Operator, OSS/BSS, Traffic Enforcer, and NFV-MANO

The following clauses describe the use case of "Traffic enforcer function" which is related to above user stories.

4.3.3.2 Actors and roles

Table 4.3.3.2-1 describes the use case actors and roles.

Table 4.3.3.2-1: Traffic enforcer function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO and the Traffic enforcer function.
2	OSS/BSS	The entity that receives the request from the Operator to trigger blocking and rerouting of traffic on selected VNFC instances.
3	Traffic enforcer	The entity that blocks and reroutes the traffic of VNFC instances via NFV-MANO based on Operator requests.
4	NFV-MANO	The entity (e.g. Container Infrastructure Service Management (CISM) [i.2] for containerized workloads) that manages the workload whose traffic is to be isolated and rerouted.

4.3.3.3 Trigger

Table 4.3.3.3-1 describes the use case trigger.

Table 4.3.3.3-1: Traffic enforcer function, trigger

Trigger	Description
Operator is requesting the blocking and rerouting of traffic on selected VNFC instances.	

4.3.3.4 Pre-conditions

Table 4.3.3.4-1 describes the pre-conditions of this use case.

Table 4.3.3.4-1: Traffic enforcer function, pre-conditions

#	Pre-condition	Description
1	Traffic enforcer function is instantiated and configured.	
2	NFV-MANO expose interfaces to Traffic enforcer function, which allow workload traffic to be isolated.	

4.3.3.5 Post-conditions

Table 4.3.3.5-1 describes the post-conditions of this use case.

Table 4.3.3.5-1: Traffic enforcer function, post-conditions

#	Post-condition	Description
1	Operator has received response to blocking and rerouting request from Traffic enforcer function.	
2	The affected VNFC instances have been successfully blocked, e.g. have been isolated and traffic is re-routed.	

4.3.3.6 Operational Flows

Table 4.3.3.6-1 describes the base flow of this use case.

Table 4.3.3.6-1: Traffic enforcer function, base flow

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS -> Traffic enforcer	The Traffic enforcer function receives and processes the request from the Operator via OSS/BSS and decides the order of actions to be performed on the VNFC instances.
1	Traffic enforcer <-> NFV-MANO	The Traffic enforcer function performs the required traffic blocking and rerouting operations on the VNFC instances by consuming the interfaces exposed by NFV-MANO and reroutes the traffic.
Ends when	Traffic enforcer -> OSS/BSS -> Operator	The Traffic enforcer function returns the result of the traffic blocking and rerouting request to the Operator via OSS/BSS.
NOTE: The "order of actions" does not exclude that certain actions are executed in parallel.		

4.3.4 Use case: VNF metrics aggregator function

4.3.4.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine. It is also assumed that the VNFC instances expose VNF-specific metrics, such as number of packets sent (network related metrics) and number of active sessions (application related metrics), which can be accessed by authorized entities.

The goal of this use case is to describe a generic "VNF metrics aggregator function" so that Operators can monitor the health and performance of multiple VNF/VNFC instances. The VNF metrics aggregator function, due to its system wide view, can for example identify performance issues that cannot be identified by monitoring single VNF/VNFC instances.

NOTE: While the "VNF metrics aggregator function" is processing metrics (i.e. "raw data") produced by the VNF/VNFC instances, ETSI GS NFV-IFA 027 [i.3] specifies measurements which are aggregated and exposed by the different NFV-MANO entities (VIM, VNFM, NFVO) based on the performance metrics collected at the NFVI.

User stories related to this use case are (not an exhaustive list):

- An Operator can retrieve information from the VNF metrics aggregator function and does not have to collect metrics from many different VNF instances and VNF-associated metrics from NFV-MANO functional entities.
- An Operator can receive aggregated information from the VNF metrics aggregator function and can get pre-processed/aggregated information showing the overall system performance compared to the performance of single VNF/VNFC instance(s).
- An Operator can manage the VNF metrics aggregator function, like filtering the type of metrics, configuring how metrics are aggregated, etc.

The following clauses describe the use case of "VNF metrics aggregator function" which is related to above user stories. References to VNFs/VNFCs metrics exposure implicitly and/or explicitly indicate metrics exposure related to the VNF application as well.

For this use case no assumption is considered regarding:

- the format of the data collected by the VNF metrics aggregator function (e.g. raw data, data series, file format, serialization format, etc.);
- the mechanism used to convey data between the data source and the VNF metrics aggregator; and
- the mechanism used by the VNF metrics aggregator function to store/retrieve data internally (e.g. storage type, storing mechanism, etc.).

4.3.4.2 Actors and roles

Table 4.3.4.2-1 describes the use case actors and roles.

Table 4.3.4.2-1: VNF metrics aggregator function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO and the VNF metrics aggregator function.
2	OSS/BSS	The entity that receives the request from the Operator to configure the VNF metrics aggregator function.
3	VNF/VNFC/NFV-MANO /NFVI	The entities that expose VNF-specific metrics (e.g. number of active sessions, packets sent) or VNF-associated metrics from NFV-MANO to the VNF metrics aggregator.
4	VNF metrics aggregator	The entity that collects the metrics from the VNF/VNFC/NFV-MANO/NFVI and exposes interfaces towards the Operator via OSS/BSS.

4.3.4.3 Trigger

Table 4.3.4.3-1 describes the use case trigger.

Table 4.3.4.3-1: VNF metrics aggregator function, trigger

Trigger	Description
Operator is requesting via OSS/BSS the configuration of the VNF metrics aggregator function and the collection of metrics.	

4.3.4.4 Pre-conditions

Table 4.3.4.4-1 describes the pre-conditions of this use case.

Table 4.3.4.4-1: VNF metrics aggregator function, pre-conditions

#	Pre-condition	Description
1	VNF metrics aggregator function is instantiated.	
2	NFV-MANO, NFVI and VNF/VNFC instances generate metrics that can be collected by the VNF metrics aggregator.	

4.3.4.5 Post-conditions

Table 4.3.4.5-1 describes the post-conditions of this use case.

Table 4.3.4.5-1: VNF metrics aggregator function, post-conditions

#	Post-condition	Description
1	Operator has access to all information required, e.g. to take countermeasures when high load on NFV-MANO interfaces or VNF/VNFC instances have been detected.	

4.3.4.6 Operational Flows

Table 4.3.4.6-1 describes the base flow of this use case.

Table 4.3.4.6-1: VNF metrics aggregator function, base flow

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS -> VNF metrics aggregator	The Operator configures via OSS/BSS the VNF metrics aggregator function, like defining the entities to be monitored, setting thresholds on certain metrics, etc.
1	VNF metrics aggregator <-> VNF/VNFC/NFV-MANO/NFVI	The VNF metrics aggregator collects metrics from the monitored targets and aggregates the information.
2	Operator <-> VNF metrics aggregator	The Operator retrieves individual or aggregated information from the VNF metrics aggregator including historical metrics data (e.g. past 24 hours) that is available from this function.
Ends when	Operator	The Operator has received the relevant information.

4.3.5 Use case: VNF metrics analyser function

4.3.5.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine. It is also assumed that there is a VNF metrics aggregator function (see use case in clause 4.3.4) providing metrics collected from the VNFC instances, such as number of packets sent (network related metrics) and number of active sessions (application related metrics) and VNF-associated metrics from different NFV-MANO functional entities.

The goal of this use case is to describe a generic "VNF metrics analyser function" so that Operators can monitor the health and performance of the system and individual VNF/VNFC instances. The VNF metrics analyser function can be configured to send notifications based on e.g. statistical processing, abnormal behaviour detection, or threshold crossing. In addition, due to its system wide view, the VNF metrics analyser function is able to, for example identify performance issues that cannot be identified by monitoring single VNF/VNFC instances.

User stories related to this use case are (not an exhaustive list):

- An Operator can retrieve processed information from the VNF metrics analyser function on the performance of individual instances, group of instances, and the overall system and can monitor the system.
- An Operator can get notifications from the VNF metrics analyser function, like abnormal behaviour detection or a threshold being crossed and can take appropriate actions where needed.

- An Operator can manage the VNF metrics analyser function, like configuring the analysis function, setting thresholds, configuring the severity level of the notifications, etc.

The following clauses describe the use case of "VNF metrics analyser function" which is related to above user stories. In the following descriptions, references to VNFs/VNFCs metrics exposure towards the VNF metrics aggregator function implicitly and/or explicitly indicate metrics information about the VNF application as well.

4.3.5.2 Actors and roles

Table 4.3.5.2-1 describes the use case actors and roles.

Table 4.3.5.2-1: VNF metrics analyser function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO, the VNF metrics aggregator and analyser functions.
2	OSS/BSS	The entity that receives the request from the Operator to configure the VNF metrics analyser function.
3	VNF/VNFC/NFV-MANO	The entities that expose VNF-specific metrics (e.g. number of active sessions, packets sent) or VNF-associated metrics from NFV-MANO via an interface to the VNF metrics aggregator.
4	VNF metrics aggregator	The entity that collects the metrics from the VNF/VNFC/NFV-MANO and exposes interfaces towards the Operator. See use case in clause 4.3.4.
5	VNF metrics analyser	The entity that analyses the metrics provided by the VNF metrics aggregator and that can notify the Operator via OSS/BSS.
6	Notification manager	The entity that handles (e.g. groups, deduplicate, routes) notifications (e.g. alerts) sent by client applications and routes them to the Operator. See use case in clause 4.3.7.

4.3.5.3 Trigger

Table 4.3.5.3-1 describes the use case trigger.

Table 4.3.5.3-1: VNF metrics analyser function, trigger

Trigger	Description
Operator is requesting via OSS/BSS the configuration of the VNF metrics analyser function.	

4.3.5.4 Pre-conditions

Table 4.3.5.4-1 describes the pre-conditions of this use case.

Table 4.3.5.4-1: VNF metrics analyser function, pre-conditions

#	Pre-condition	Description
1	VNF metrics aggregator function is instantiated and configured to collect metrics from the VNF/VNFC instances and VNF-associated metrics from NFV-MANO.	
2	VNF metrics analyser function is instantiated.	

4.3.5.5 Post-conditions

Table 4.3.5.5-1 describes the post-conditions of this use case.

Table 4.3.5.5-1: VNF metrics analyser function, post-conditions

#	Post-condition	Description
1	Operator has received a notification with information related to an event observed by the VNF metrics analyser function. The Operator also has access to related metrics required to troubleshoot the cause of the event.	

4.3.5.6 Operational Flows

Table 4.3.5.6-1 describes the base flow of this use case.

Table 4.3.5.6-1: VNF metrics analyser function, base flow

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS -> VNF metrics analyser	The Operator configures via OSS/BSS the VNF metrics analyser to retrieve data from the VNF metrics aggregator and how to process it.
1	VNF metrics analyser <-> VNF metrics aggregator	The VNF metrics analyser retrieves data from the VNF metrics aggregator.
2	VNF metrics analyser	The VNF metrics analyser processes and evaluates the data collected. For example, it detects an event, e.g. when the load on multiple VNF/VNFC is above a given threshold for a given time period, analyses the data for abnormal behaviour, correlates events to identify likely culprits, identifies possible performance issues affecting multiple instances, tries to identify possible silent failures, etc.
3	VNF metrics analyser -> (Notification manager) -> OSS/BSS -> Operator	The Analyser issues an alert notification with information related to the event to the Operator via OSS/BSS. Whether this notification is sent directly or via a Notification manager is out of scope of this use case.
4	Operator <-> VNF metrics analyser, Operator <-> VNF metrics aggregator	The Operator collects additional information from the VNF metrics analyser and/or VNF metrics aggregator.
Ends when	Operator	The Operator has received the relevant information.

4.3.6 Use case: Time function

4.3.6.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine.

The goal of this use case is to describe a generic "Time function" that is ensuring the time synchronicity of multiple VNFs and their VNFC with the master time server in the Operator's network. In case of any issues detected, the "Time function" raises an alert and may also take appropriate actions, e.g. set the correct time in the VNFC/host. The function will also provide logs for troubleshooting.

User stories related to this use case are (not an exhaustive list):

- An Operator can rely on the Time function, to ensure that the system time of all VNFs and their components is synchronized, i.e. the time skew is kept within a certain boundary.
- An Operator can use the Time function to configure the time protocol(s) used in the system, e.g. to configure the time source and does not have to configure each host/VNFC separately.
- An Operator can get alerts and logs from the Time function, and can troubleshoot and take appropriate actions where needed. This includes, e.g. logs collected from the slaves/clients about time skew observed, corrective actions, and alerts if one of more of the master time server nodes are not reachable. The logs can be collected and retrieved via corresponding responsible functions such as the "Log aggregator function".

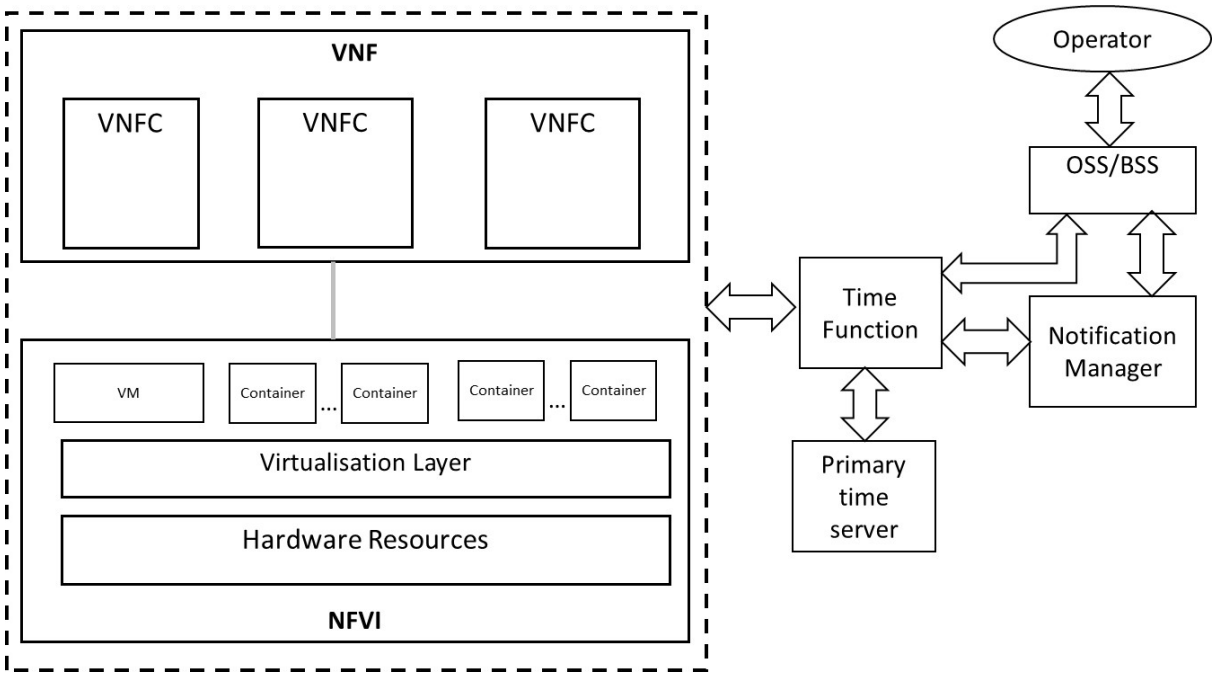

Figure 4.3.6.1-1: Example of relationship between entities involved in the Time function use case

The following clauses describe the use case of "Time function" which is related to above user stories.

4.3.6.2 Actors and roles

Table 4.3.6.2-1 describes the use case actors and roles.

Table 4.3.6.2-1: Time function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled the Notification manager, Time function.
2	OSS/BSS	The entity that receives the request from the Operator to configure the Time function.
3	Primary time server(s)	(Group of) primary/master time server(s) running in the Operator network. EXAMPLE: A stratum 1 NTP server.
4	Time function	Time function, providing/managing a secondary/intermediate time server(s). The secondary time servers are syncing with the primary/master time server(s). EXAMPLE: The secondary time servers can be realized by a stratum 2 NTP server.
5	VNFC/host	VNF component or host system running a time client. The VNFC/host will sync with secondary/intermediate time server(s).
6	Notification manager	The entity that handles (e.g. groups, deduplicate, routes) notifications (e.g. alerts) sent by client applications and routes them to the Operator. See use case in clause 4.3.7.

4.3.6.3 Time synchronization (base flow #1)

4.3.6.3.1 Introduction

In this base flow, the objective is to provision to the hosts/VNFCs a common secondary time synchronization server provided by a Time function that is in turn synchronized with primary time servers(s).

4.3.6.3.2 Trigger

Table 4.3.6.3.2-1 describes the use case trigger.

Table 4.3.6.3.2-1: Time function, trigger for base flow #1

Trigger	Description
Operator is requesting via OSS/BSS the configuration of the Time function.	

4.3.6.3.3 Pre-conditions

Table 4.3.6.3.3-1 describes the pre-conditions of this use case for base flow #1.

Table 4.3.6.3.3-1: Time function, pre-conditions for base flow #1

#	Pre-condition	Description
1	Notification manager is instantiated and configured.	
2	Time function is instantiated.	

4.3.6.3.4 Post-conditions

Table 4.3.6.3.4-1 describes the post-conditions of this use case for base flow #1.

Table 4.3.6.3.4-1: Time function, post-conditions for base flow #1

#	Post-condition	Description
1	VNFC/hosts are in time sync with the secondary time server.	

4.3.6.3.5 Operational Flow #1

Table 4.3.6.3.5-1 describes the base flow #1 of this use case.

Table 4.3.6.3.5-1: Time function, base flow #1

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS -> Time function	The Operator configures via OSS/BSS the Time function, e.g. information about the Primary time server.
1	Time function	The Time function synchronizes its internal time with the Primary time server.
2	Time function -> VNFC/host	Each VNFC/host synchronizes its internal time with the configured Time function. This step runs continuously, unless otherwise stated.
Ends when	VNFC/host	VNFCs/hosts are in time sync with the secondary time server within a certain acceptable skew.

4.3.6.4 Time re-synchronization after drift (base flow #2)

4.3.6.4.1 Introduction

In this base flow, the objective is to ensure that the Time function and the hosts/VNFCs become again time synchronized after a drift in time sync has been detected.

4.3.6.4.2 Trigger

Table 4.3.6.4.2-1 describes the use case trigger.

Table 4.3.6.4.2-1: Time function, trigger for base flow #2

Trigger	Description
One of the secondary time servers has drifted by more than an Operator-configured threshold of seconds.	

4.3.6.4.3 Pre-conditions

Table 4.3.6.4.3-1 describes the pre-conditions of this use case for base flow #2.

Table 4.3.6.4.3-1: Time function, pre-conditions for base flow #2

#	Pre-condition	Description
1	Time function and notification manager are instantiated and configured.	
2	Primary time server is available and in-sync with a reference clock.	

4.3.6.4.4 Post-conditions

Table 4.3.6.4.4-1 describes the post-conditions of this use case for base flow #2.

Table 4.3.6.4.4-1: Time function, post-conditions for base flow #2

#	Post-condition	Description
1	The secondary time server is again within the allowed boundaries against the primary time server(s).	
2	The Operator has received a notification from the time function related to a secondary time server being out of sync, as well as a notification that the situation is resolved. The Operator can also access logs related to the situation in order to troubleshoot the problem, e.g. via the "Log aggregator function".	

4.3.6.4.5 Operational Flow #2

Table 4.3.6.4.5-1 describes the base flow #2 of this use case.

Table 4.3.6.4.5-1: Time function, base flow #2

#	Actor/Role	Description
Begins when	Time function	The Time function detects that one of the secondary time servers is out of sync by more than an allowed threshold of seconds.
1	Time function -> (Notification manager) -> Operator	The Time function sends a notification to the Operator. See notes 1 and 3.
2	Time function	The Time function will set the correct time in the secondary time server. See notes 2 and 3.
3	Time function -> VNFC/host	Each VNFC/host continues synchronizing its internal time with the configured Time function. In case a VNFC/host detects after the resetting of the time that its own time information is out of sync by more than an allowed threshold of seconds, the Time function will take the appropriate measures to set the correct time in the VNFC/host. See notes 2 and 3.
4	Time function -> (Notification manager) -> Operator	After the issue has been resolved, the Time function sends a notification to the Operator. See notes 1 and 3.
Ends when	Operator	The Operator has received notifications about the issue and can access additional logs that are available from the Time function, e.g. via the "Log aggregator function" (see also note 3). All components are again in sync with the primary time server.
NOTE 1: Whether alerts are sent directly or via a Notification manager is out of scope of this use case. NOTE 2: If setting the time is not possible, or any other situation occurs that the Time function cannot handle, the issue would be escalated to the Operator. NOTE 3: All events related to notifications, out-of-sync, re-synchronization, etc. are logged. For this, the Log aggregator can be leveraged as described in clause 4.3.1.		

4.3.7 Use case: Notification manager function

4.3.7.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine.

The goal of this use case is to describe a generic "Notification manager function" that is handling notifications, sent by other VNF generic OAM functions, such as alerts. The function manages the notifications, e.g. deduplication, grouping, and routing the notifications to the intended receiver, e.g. NFV-MANO, the Operator.

User stories related to this use case are (not an exhaustive list):

- An Operator or NFV-MANO can subscribe to retrieve notifications, like alerts, from the notification manager function can receive immediate notifications, e.g. in case of performance issues or faults observed.
- An Operator or NFV-MANO can subscribe to retrieve notifications from the Notification manager function and does not have to subscribe to many different entities and avoid duplicate notifications.
- An Operator or NFV-MANO can use the Notification manager function and can manage the notifications interesting to receive. This includes, inhibiting or silencing notifications.

The following clauses describe the use case of "Notification manager function" which is related to above user stories.

4.3.7.2 Actors and roles

Table 4.3.7.2-1 describes the use case actors and roles.

Table 4.3.7.2-1: Notification manager function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO, the VNF generic OAM and Notification manager functions.
2	OSS/BSS	The entity that receives the request from the Operator to configure the Notification manager.
3	VNF generic OAM function	A VNF generic OAM function. EXAMPLE: VNF metrics analyser function. See use case in clause 4.3.5.
4	Notification manager	The entity that handles (e.g. groups, deduplicate, routes) notifications (e.g. alerts) sent by the VNF generic OAM function and routes them to NFV-MANO and the Operator.
5	NFV-MANO	The entities that consume the notifications sent by the Notification Manager.

4.3.7.3 Trigger

Table 4.3.7.3-1 describes the use case trigger.

Table 4.3.7.3-1: Notification manager function, trigger

Trigger	Description
Operator is requesting via OSS/BSS the configuration of the Notification manager function.	

4.3.7.4 Pre-conditions

Table 4.3.7.4-1 describes the pre-conditions of this use case.

Table 4.3.7.4-1: Notification manager function, pre-conditions

#	Pre-condition	Description
1	The VNF generic OAM function is instantiated and configured.	
2	Notification manager function is instantiated.	
3	NFV-MANO environment is operational.	
4	NFV-MANO and the Operator are subscribed to receive notifications from the Notification manager.	

4.3.7.5 Post-conditions

Table 4.3.7.5-1 describes the post-conditions of this use case.

Table 4.3.7.5-1: Notification manager function, post-conditions

#	Post-condition	Description
1	NFV-MANO and the Operator have received a notification with information about an event observed by the VNF generic OAM function.	

4.3.7.6 Operational Flows

Table 4.3.7.6-1 describes the base flow of this use case.

Table 4.3.7.6-1: Notification manager function, base flow

#	Actor/Role	Description
Begins when	Operator <-> OSS/BSS <-> Notification manager	The Operator configures via OSS/BSS the Notification manager, e.g. information about the VNF generic OAM function.
1	VNF generic OAM function -> Notification manager	The VNF generic OAM function detects an event and sends a notification to the Notification manager.
2	Notification manager -> NFV-MANO, Operator	The Notification manager forwards the notification to NFV-MANO and the Operator.
Ends when	NFV-MANO, Operator	The Operator and NFV-MANO have received the relevant information.

4.3.8 Use case: Network configuration manager function

4.3.8.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine. As part of this functionality the Network configuration manager function can be used by NFV-MANO to establish the configurations described in the VLD and CPD declarative descriptors of the VNF and other configuration provided at runtime (e.g. via interfaces).

The goal of this use case is to describe a generic "Network configuration manager function" that is handling the configuration of the external connectivity of VNFs/VNFCs.

In an example use case flow, the Network configuration manager function sets configuration information to disconnect one or more external CPs of one VNF instance (VNF 1) from one PNF instance (PNF 1) and reconnects the CP(s) to another PNF instance (PNF 2) as illustrated in Figure 4.3.8.1-1. This enables applying various types of network configuration on VNF/VNFC instance in a consistent manner.

NOTE: It is out of the scope of the present document to specify the types of network configuration that are pushed from OSS/BSS or NFV-MANO towards the Network Configuration Manager.

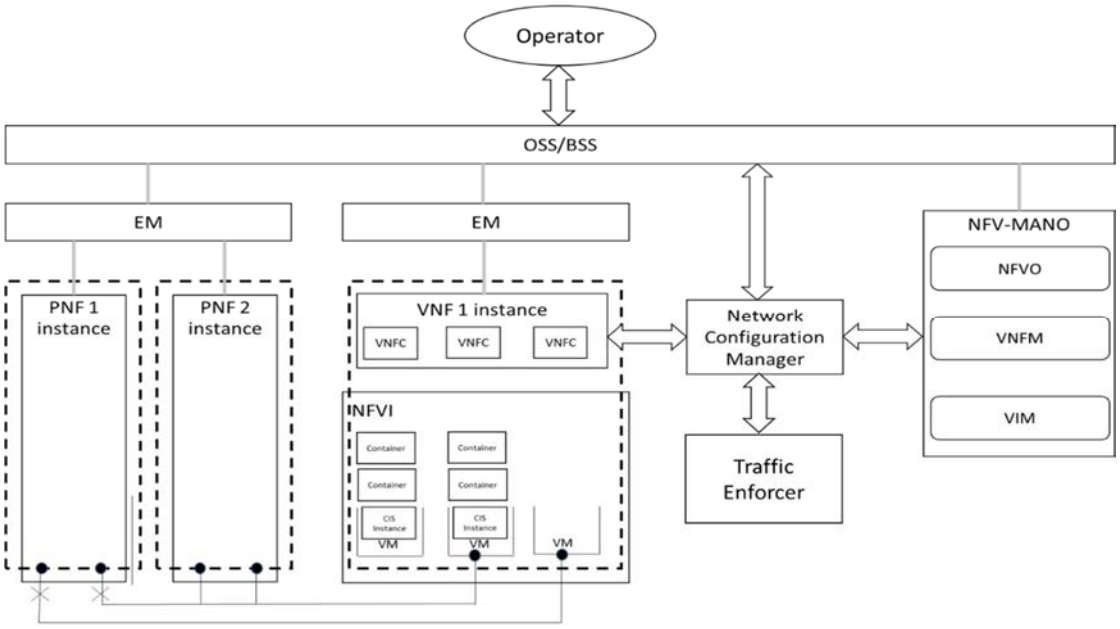

Figure 4.3.8.1-1: Example of relationship between entities involved in the network configuration manager use case

User stories related to this use case are:

- An Operator can request the Network configuration manager function to configure in a consistent manner both the connectivity and the applications of the VNF/VNFC instances.
- An Operator can change the configuration related to connection points of VNF instances in order to preserve the communication of VNF/VNFC instances when maintaining networks or VNF instances, e.g. hardware replacement.

The following clauses describe the use case of "Network configuration manager function" which is related to above user stories.

4.3.8.2 Actors and roles

Table 4.3.8.2-1 describes the use case actors and roles.

Table 4.3.8.2-1: Network configuration manager function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO and the Network configuration manager.
2	OSS/BSS	The entity that receives request from the Operator to change external VNF connectivity.
3	NFV-MANO	The entities that receive request from the OSS/BSS to change external VNF connectivity and manage the addition/deletion of external link ports of the VNF instance.
4	Network configuration manager	The entity that manages the distribution of the target network configuration to the VNF/VNFC(s).
5	Traffic enforcer function	The entity that blocks the traffic of VNFC instances.
6	VNF/VNFC instance	The entities that are configured by the Network Configuration Manager.

4.3.8.3 Trigger

Table 4.3.8.3-1 describes the use case trigger.

Table 4.3.8.3-1: Network configuration manager function, trigger

Trigger	Description
Operator detects a changing service condition which requires network configuration change of a VNF instance.	

4.3.8.4 Pre-conditions

Table 4.3.8.4-1 describes the pre-conditions of this use case.

Table 4.3.8.4-1: Network configuration manager function, pre-conditions

#	Pre-condition	Description
1	Network configuration manager is instantiated to configure VNF/VNFC instances.	
2	NFV-MANO environment is operational.	
3	VNF instance is connected with another VNF/PNF instance via an old Virtual Link (VL).	
4	Traffic enforcer function is instantiated.	
5	Target Virtual Link (VL) is available but not yet configured to be used.	

4.3.8.5 Post-conditions

Table 4.3.8.5-1 describes the post-conditions of this use case.

Table 4.3.8.5-1: Network configuration manager function, post-conditions

#	Post-condition	Description
1	The VNF instance connects to a different VNF/PNF instance via the target VL.	
2	Operator can monitor normality of service for new network configuration and VL after changing service condition.	

4.3.8.6 Operational Flows

Table 4.3.8.6-1 describes the base flow of this use case. In this base flow, a change external VNF connectivity request is shown as an example of a network configuration applied by this VNF generic OAM function.

Table 4.3.8.6-1: Network configuration manager function, base flow

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS	Operator requests to OSS/BSS to change the network configuration of a particular VNF instance.
1	OSS/BSS -> NFV-MANO	OSS/BSS sends NS update request to NFV-MANO to perform external VNF connectivity change.
2	OSS/BSS -> Network configuration manager	If needed, OSS/BSS provides to the Network configuration manager additional network configuration to be applied, e.g. DNS, routing table of load balancer.
3	NFV-MANO -> Network configuration manager	NFV-MANO sends request with new network configuration related to VNF/VNFC instances, e.g. segment identifier, IP address to Network configuration manager.
4	Network configuration manager -> VNF/VNFC instance	Network configuration manager configures VNF/VNFC instance and related network entities. See note.
5 (optional)	Network configuration manager -> Traffic enforcer function	Network configuration manager requests Traffic enforcer function to block traffic towards old VL.
6 (optional)	Traffic enforcer function -> Network Configuration Manager	Traffic enforcer function notifies the completion of traffic blocking to Network Configuration manager.
7	Network configuration manager -> NFV-MANO or OSS/BSS	Network configuration manager notifies to NFV-MANO or OSS/BSS about completion of network configuration.
8	NFV-MANO	NFV-MANO changes connectivity by deleting link port of old VL and adding link port of target VL.
9	NFV-MANO -> OSS/BSS -> Operator	NFV-MANO notifies to Operator via the OSS/BSS about completion of the NS update operation.
Ends when	Operator	Operator has received all relevant information.
NOTE: Network configuration manager needs to be able to recognize that requests originating according to step 2 and step 3 need to be applied consistently.		

4.3.9 Use case: Upgrade VNF function

4.3.9.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine.

The goal of this use case is to describe a generic "Upgrade VNF function" that is handling in coordination with NFV-MANO the software upgrade of VNF/VNFC instances in order to run with new software version and configuration. The "Upgrade VNF function" is exposing interfaces towards VNF/VNFC instances, NFV-MANO and other VNF generic OAM functions to enable and automate the actual upgrade process.

The Upgrade VNF function leverages and complements the existing NFV-MANO functionality regarding "Change current VNF Package". NFV-MANO is responsible for e.g. handling the resource fulfilment and changes of the VNF, while the Upgrade VNF function performs the upgrade coordination actions for the VNF itself, e.g. reusing or defining new LCM coordination actions. In other words, the Upgrade VNF function assists in a generic way with upgrade tasks on the VNF side (e.g. application configuration on new VNFC instances, applying new database schema, assisting on the control of traffic of VNFC to be upgraded), while NFV-MANO ensures that the VNF virtualised resources are (temporarily) added or removed and software images are distributed to the underlying platform (e.g. CIR for OS container images).

User stories related to this use case are:

- An Operator can request the Upgrade VNF function to provide new service by upgrading to a new software version and adding network connectivity to new type of VNF instance, e.g. update software of VNF/VNFC, import new service name, import new certificate for other VNF in load balancer, setting configuration of CP in load balancer, so that complexity on the Operator side can be reduced when providing a new service.
- An Operator can request the Upgrade VNF function to enable an additional virtual resource of a VNFC instance after VNF upgrading, e.g. enable adding CPU or memory, or adding or extending volume to use by extending the storage size, to allow OAM functionalities (currently implemented as part of the VNF) be moved to the upgrade VNF function to prepare for enabling cloud-native VNFs.

- An Operator can request Upgrade VNF function to coordinate updating VNFs to run with new software, e.g. reference to software images (VM or OS container images), database schema change, application configuration files so that the Operator can be relieved of such coordination tasks.

The following clauses describe the use case of "Upgrade VNF function" which is related to above user stories.

4.3.9.2 Actors and roles

Table 4.3.9.2-1 describes the use case actors and roles.

Table 4.3.9.2-1: Upgrade VNF function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates NFV-MANO and the Upgrade VNF function.
2	OSS/BSS	The entity that receives the request from the Operator to upgrade VNF instances.
3	Upgrade VNF function	The entity that provides configuration to VNF/VNFC instances and coordinates other application specific tasks during the upgrade.
4	VNF/VNFC instances	The entities that are upgraded by the upgrade VNF function in coordination with NFV-MANO.
5	Traffic enforcer function	The entity that isolates the VNFCs to be upgraded during the upgrade process.
6	Network configuration manager function	The entity that changes connectivity of the VNF/VNFC instances during the upgrade process.
7	NFV-MANO	The entities that perform and apply the change current VNF Package operation.

4.3.9.3 Trigger

Table 4.3.9.3-1 describes the use case trigger.

Table 4.3.9.3-1: Upgrade VNF function, trigger

Trigger	Description
OSS/BSS receives a request from the Operator to upgrade VNF/VNFC instance(s).	

4.3.9.4 Pre-conditions

Table 4.3.9.4-1 describes the pre-conditions of this use case.

Table 4.3.9.4-1: Upgrade VNF function, pre-conditions

#	Pre-condition	Description
1	Upgrade VNF function is instantiated.	
2	Traffic enforcer function is instantiated.	
3	Network configuration manager function is instantiated.	
4	NFV-MANO environment is operational.	
5	VNF/VNFC instances run with an old version.	
6	A new VNF Package is available with the new version but is not yet onboarded.	

4.3.9.5 Post-conditions

Table 4.3.9.5-1 describes the post-conditions of this use case.

Table 4.3.9.5-1: Upgrade VNF function, post-conditions

#	Post-condition	Description
1	The VNF/VNFC instances run with new version.	
2	NFV-MANO manages VNF/VNFC instances with new version.	
3	Operator can monitor normality of service after this upgrade process.	

4.3.9.6 Operational Flows

Table 4.3.9.6-1 describes the base flow of this use case.

Table 4.3.9.6-1: Upgrade VNF function, base flow

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS	Operator requests OSS/BSS to update/upgrade VNF/VNFC instances.
1	OSS/BSS <-> NFV-MANO	OSS/BSS onboards a VNF Package with new file(s). The VNF Package might include the reference to software images (VM or OS container images), and some configuration files.
2	OSS/BSS -> NFV-MANO	OSS/BSS sends request to change current VNF Package to NFV-MANO as described as part of the "Update NS operation" in clause 7.3.5 of ETSI GS NFV-IFA 013 [i.9].
3	NFV-MANO	NFV-MANO starts executing the change current VNF Package operation as described as part of the "Change current VNF package operation" in clause 7.2.23 of ETSI GS NFV-IFA 007 [i.10]. NFV-MANO processes the onboarded VNF Package (including the target VNFD). The interactions of changing current VNF Package, granting and resource changes take place as described in steps 2 to 11 in the procedure documented in clause B.3.2 of ETSI GS NFV-IFA 007 [i.10]. Looping for the various modification processes can start as illustrated before step 11 of the referred procedure.
4	NFV-MANO <-> Upgrade VNF function	If the modification process includes coordination actions of the VNFM with the VNF or EM (as supported by the Upgrade VNF function), this is triggered according to the steps 12 of the procedure in clause B.3.2 of ETSI GS NFV-IFA 007 [i.10].
5	NFV-MANO <-> Upgrade VNF function	The processing of coordination actions and changes starts as indicated in step 13 of the procedure in clause B.3.2 of ETSI GS NFV-IFA 007 [i.10]. If necessary, as part of the present coordination action, relevant files which are needed to assist with upgrade tasks on the VNF side and that are to be processed during the modification process and coordination actions (e.g. new database schema, additional executables for the VNF) are provisioned to the Upgrade VNF function. See note 1. The following steps 6 to 9 describe the actions that the VNF upgrade function can perform as part of the triggered coordination action and are thus regarded as inner sub-steps of the step 13 of the procedure in clause B.3.2 of ETSI GS NFV-IFA 007 [i.10].
6	Upgrade VNF function <-> Traffic enforcer function	Upgrade VNF function sends request to Traffic enforcer function to isolate particular VNFC instances. See note 2.
7	Upgrade VNF function <-> VNFC instances	Upgrade VNF function installs new file(s) to the VNFC instances that had been isolated in step 6 and/or configures resources related to the VNF, e.g. configures a database.
8	Upgrade VNF function <-> Traffic enforcer function	Upgrade VNF function sends a request to Traffic enforcer function to activate the VNFC instances that had received the new file(s) and/or configuration in step 7.
9	Upgrade VNF function <-> VNFC instances	Upgrade VNF function observes the normality of VNFC instances running with new file for certain amount of time.
		Repeat steps 6 to 9 of the present flow for the remaining VNFC instances, if any.
10	Upgrade VNF function -> NFV-MANO	Upgrade VNF function notifies back to NFV-MANO about the completion of the coordination action following step 14 of the procedure in clause B.3.2 of ETSI GS NFV-IFA 007 [i.10].
		Repeat steps 4 to 10 of the present flow as needed depending on the modification process and as indicated by the loop of steps 11 to 17 in the procedure in clause B.3.2 of ETSI GS NFV-IFA 007 [i.10].

#	Actor/Role	Description
11	NFV-MANO -> OSS/BSS	NFV-MANO performs the remaining steps of change current VNF package operation and notifies back to OSS/BSS about the result of the operation. If there was abnormality observed in steps 4 to 9 of the present flow, appropriate information is provided to OSS/BSS to support the decision making about whether a roll back should be initiated.
12 (optional)	OSS/BSS or Upgrade VNF function <-> Network configuration manager	OSS/BSS or Upgrade VNF function sends request to Network configuration manager to add network configuration, if new VNF Package which OSS/BSS had onboarded required to add a new network in step 1.
13 (optional)	Network configuration manager -> OSS/BSS	Network configuration manager notifies to OSS/BSS about the completion of the network configuration request.
14	OSS/BSS -> Operator	OSS/BSS notifies to Operator about the completion of the update/upgrade request.
Ends when	Operator	Operator has received the relevant information.
NOTE 1: The provisioning of the necessary files to the VNF upgrade function could be either done by the VNFM "pushing" the files to the VNF upgrade function, or the VNF upgrade function "pulling" them from the VNFM. NOTE 2: The number of VNFCs to be isolated will be limited to ensure that the service is kept running normally, e.g. keep up a certain redundancy of the VNFC instances.		

4.3.10 Use case: VNF configuration manager function

4.3.10.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine.

The goal of this use case is to describe a generic "VNF configuration manager function" that is handling changes to the configuration of a VNF/VNFC.

In an example use case flow, the VNF configuration manager function sets configuration information to one or more VNF/VNFC instances. This enables applying various types of configuration on VNF/VNFC instance in a consistent manner.

The VNF/VNFC configuration that can be set via this VNF configuration manager, is configuration that is typically set by the Operator or an administrator, e.g. NFV-MANO-related configurations, certain application-related thresholds, etc.

NOTE 1: It is out of the scope of the present document to specify the specific types of configuration that can be pushed from OSS/BSS to the VNF/VNFC instances via the VNF Configuration Manager.

User stories related to this use case are:

- An Operator can request the VNF configuration manager function to configure in a consistent manner both the VNF/VNFC instances as seen from NFV-MANO as well as the application related configuration of the VNF/VNFC instances, and does not have to be involved in the distribution of the target configuration setting as well as the preparation (e.g. create backup) and postprocessing actions (e.g. verify/test the new configuration).
- NFV-MANO can query for configuration records stored by the VNF configuration manager and reapply the same configuration to new VNF/VNFC instances.

NOTE 2: The VNF configuration manager does not understand the semantics of the configuration that are pushed to the VNF/VNFC instances.

The following clauses describe the use case of "VNF configuration manager function" which is related to above user stories.

4.3.10.2 Actors and roles

Table 4.3.10.2-1 describes the use case actors and roles.

Table 4.3.10.2-1: VNF configuration manager function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO and the VNF configuration manager.
2	OSS/BSS	The entity that receives request from the Operator to change the configuration of the VNF/VNFC(s).
3	VNF configuration manager	The entity that manages the distribution of the target configuration to the VNF/VNFC(s). This can include preparation (e.g. create backup) and postprocessing actions (e.g. verify/test the new configuration).
4	VNF/VNFC instances	The entities that are configured by the VNF configuration manager.

4.3.10.3 Trigger

Table 4.3.10.3-1 describes the use case trigger.

Table 4.3.10.3-1: VNF configuration manager function, trigger

Trigger	Description
VNF configuration manager receives from the Operator a request to change the configuration of a particular VNF/VNFC instance.	

4.3.10.4 Pre-conditions

Table 4.3.10.4-1 describes the pre-conditions of this use case.

Table 4.3.10.4-1: VNF configuration manager function, pre-conditions

#	Pre-condition	Description
1	VNF configuration manager is instantiated to configure VNF/VNFC instances.	
2	NFV-MANO environment is operational.	
3	Target configuration to be distributed to the VNF/VNFC instances is available.	

4.3.10.5 Post-conditions

Table 4.3.10.5-1 describes the post-conditions of this use case.

Table 4.3.10.5-1: VNF configuration manager function, post-conditions

#	Post-condition	Description
1	The VNF/VNFC instances have successfully been configured with the target configuration, and the system condition has been checked after the modification of the configuration.	
2	The VNF configuration manager has stored a copy of the current VNF configuration and the previous VNF configuration.	A copy of the current VNF configuration is stored in order to e.g. re-apply the configuration to the instance in the future or for troubleshooting purposes. A copy of the previous VNF configuration is stored, e.g. for troubleshooting or to go back to the previous configuration in case setting the new configuration leads to an issue. The VNF configuration may also store more historical VNF configuration records for troubleshooting purposes.
3	Operator can monitor normality of service with the new configuration.	

4.3.10.6 Operational Flows

Table 4.3.10.6-1 describes the base flow of this use case.

Table 4.3.10.6-1: VNF configuration manager function, base flow

#	Actor/Role	Description
Begins when	Operator -> OSS/BSS	Operator has identified a new configuration to be pushed to selected VNF/VNFC instance(s) and requests to OSS/BSS to change the configuration of those one or more VNF/VNFC instance(s).
1	OSS/BSS -> VNF configuration manager	OSS/BSS sends a request with the new VNF/VNFC configuration to the VNF configuration manager. The request includes the new configuration data as well as the target VNF/VNFC instance(s). The request can also include information to be used to verify/test the new configuration. The request can further include multiple configuration data along with information about which information should be set to which of the VNF/VNFC instances (e.g. based on a filter).
2	VNF configuration manager	VNF configuration manager checks the system condition (e.g. state of the VNF, ongoing VNF LCM operation, etc.) based on the information provided by the OSS/BSS to ensure that the system and the VNF/VNFC instances are ready to receive the new configuration.
3	VNF configuration manager <-> VNF/VNFC instances	VNF configuration manager reads the current configuration of the target VNF/VNFC instances, e.g. to create a backup of the configuration data, to check if the configuration is already up-to-date, etc. To check if the configuration is already up-to-date, the VNF configuration manager can compare (string comparison) the set of parameter and values (e.g. key value pairs) read from the current configuration with those that are expected to be configured. The VNF configuration manager also verifies if the target configuration is writeable, or whether the configuration is read-only.
4	VNF configuration manager -> VNF/VNFC instances	VNF configuration manager applies the target configuration to the target VNF/VNFC instance(s).
5	VNF configuration manager <-> VNF/VNFC instances	VNF configuration manager checks the configuration after the change in order to ensure that the configuration has been correctly set. Checking the configuration can be performed as indicated in step 3 with string comparison of parameter-value sets.

#	Actor/Role	Description
6	VNF configuration manager <-> VNF/VNFC instances	VNF configuration manager tests/verifies the modified VNF/VNFC instances based on the information/instructions provided by the OSS/BSS in the request.
7	VNF configuration manager -> OSS/BSS	VNF configuration manager notifies to OSS/BSS about the completion of the VNF configuration request.
8	OSS/BSS -> Operator	OSS/BSS notifies to Operator about the completion of the change VNF configuration request.
Ends when	Operator	Operator has received the relevant information.

4.3.11 Use case: VNF testing manager

4.3.11.1 Introduction

In the present use case, it is assumed that the VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers [i.2] or a virtual machine.

The goal of this clause is to describe a generic "VNF testing manager". The VNF testing manager can be used to support multilayer testing functionalities up to the application layer related to VNF and NFV-MANO operations:

- An operator can request the VNF testing manager to support connectivity testing between VNFC instances or between VNFC instances and other entities (e.g. storage devices, DNS, gateway systems, etc.). Examples of connectivity testing operations are configuring and triggering a connectivity test and collection of connectivity testing results. Besides reachability tests, connectivity tests can be also related to QoS measurements (e.g. link load and congestion levels, packet loss, etc.).

When testing is performed in a virtualized environment, multiple different dimensions can be considered. For example:

- Connectivity testing considering the underlay network and/or the overlay network.
- Different sending and receiving entities like VNF-to-VNF, VNFC-to-gateway, Node-to-Node, etc.
- Different virtualization technologies (e.g. VM-based, container-based).

In the case of connectivity testing, the VNF testing manager is the entity which is used to configure the test, start the test and also collect the result of the connectivity test.

The following assumptions are considered regarding the operation of the VNF generic OAM function VNF testing manager. The VNF testing manager:

- exposes a testing management interface in the northbound which can be consumed by authorized consumers (NFV-MANO, OSS, etc.);
- depending on the test the VNF testing manager, can receive relevant instructions (e.g. configuration of the test, test report collection etc.);
- interacts with all the appropriate entities management systems to support the actual configuration of the test, execution of the test and results retrieval;
- dynamically identifies endpoints used for the test (e.g. IP addresses, Ports, FQDN, URI or other);
- translates a received (sub-)test specification to actions (e.g. which tools to use and which configuration) for the test;
- can cover testing for one or more layers of the protocol stack; and
- as a VNF generic OAM function, it can interact with NFV-MANO entities, OSS, NFVI, VNFs and other functions (e.g. MDAF) to support the execution of the test.

See clause 4.4.1.4 for a use case description related to Intra-NFVI-Pop connectivity testing and clause 4.4.1.5 for a use case description related to Inter-NFVI-Pop connectivity testing.

4.3.12 Use case: Policy Management for VNF Generic OAM Functions

4.3.12.1 Introduction

Besides MDA and Intent management, automation management in ETSI NFV is closely related to policy management. ETSI GR NFV-IFA 023 [i.21] describes use cases related to policy management, provides examples of mapping PF and PAP to NFVMANO FBs and categorizes the types of NFV-MANO policy. A PAP defines policies, and a PF evaluates policies and executes the policy decisions (policy enforcement). An information model for NFV-MANO policies is specified in ETSI GS NFV-IFA 048 [i.17]. In NFV-MANO, each FB (e.g. NFVO) exposes a Policy management interface. In ETSI GS NFV-SOL 012 [i.22] RESTful protocol and data models fulfilling the requirements related to policy managements interfaces are specified. ETSI GS NFV-SOL 012 [i.22] regards a policy as an artefact.

In this use case a VNF generic OAM function called Policy Agent (PA) is exposing a Policy management interface and can receive policies for itself but also for any VNF and any other VNF Generic OAM function. In both cases this single VNF generic OAM function is responsible to perform the relevant PF operations.

Regarding the operations of the Policy Agent, the following assumptions are considered:

- The Policy Agent is exposing a policy management interface in the northbound.
- No restriction is imposed on the entity which is acting as a PAP and can interact with the Policy Agent to convey the policy.
- The Policy Agent is acting as a PF for policies targeting itself.
- The Policy Agent is acting as a PF for policies targeting other VNF generic OAM functions. Based on the policy enforcement the appropriate interactions are made with other VNF Generic OAM functions.
- The Policy Agent can act as a PF for policies targeting VNFs consuming the generic OAM functions.
- The Policy Agent can directly interact with OSS, NFV-MANO, VNF generic OAM functions, VNFs and other functions (e.g. MDAF).
- The present use case precludes that other VNF generic OAM functions expose a Policy management interface.

In the example scenario, the Policy Agent is used to enforce a policy related to the Traffic Enforcer VNF generic OAM function, see Figure 4.3.12.1-1. The operator is acting as a PAP and the Policy Agent as a PF. Lines in Figure 4.3.12.1-1 represent points of interaction assumed in the use case. The interactions between NFV-MANO and OSS etc. are not depicted for the sake of simplicity.

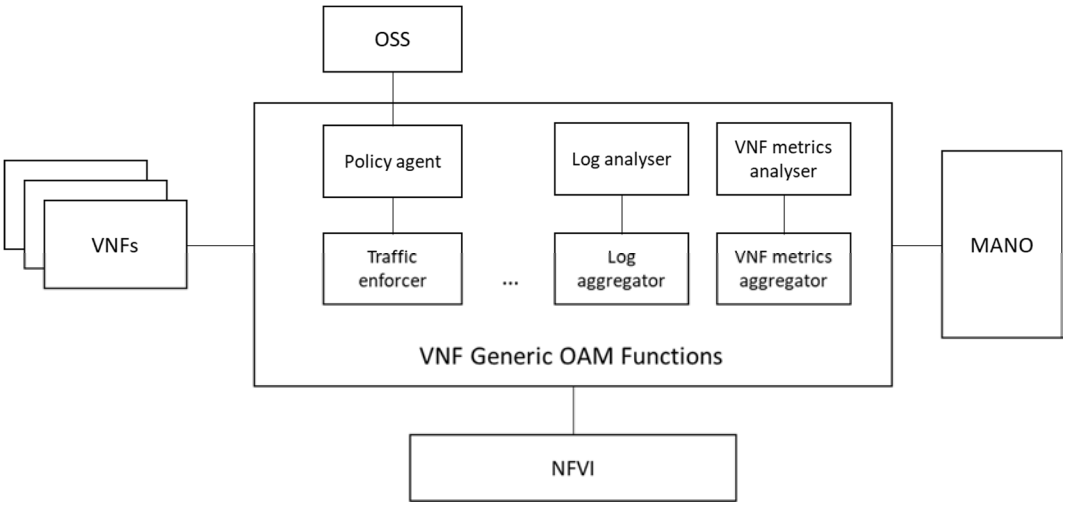

Figure 4.3.12.1-1: Policy transferring and policy enforcement through the Policy Agent

4.3.12.2 Actors and roles

Table 4.3.12.2-1 describes the use case actors and roles.

Table 4.3.12.2-1: Policy Agent function, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO, the VNF generic OAM functions and the Policy agent. It acts as a PAP.
2	OSS/BSS	The entity that receives the request from the Operator to configure the VNF generic OAM functions. It conveys the policy to the Policy agent.
3	Policy Agent	The entity responsible for parsing a policy from OSS/BSS and perform PF related operations.
4	Traffic enforcer	The entity that blocks and reroutes the traffic of VNFC instances via NFV-MANO based on Operator requests.

4.3.12.3 Trigger

Table 4.3.12.3-1 describes how the operational flow for this use case is triggered.

Table 4.3.12.3-1: Policy agent function, trigger

Trigger	Description
Operator requests via OSS/BSS to apply a policy related to the Traffic Enforcer.	

4.3.12.4 Pre-conditions

Table 4.3.12.4-1 describes the pre-conditions of this use case.

Table 4.3.12.4-1: Policy agent function, pre-conditions

#	Pre-condition	Description
1	VNF generic OAM functions (i.e. Policy Agent and Traffic Enforcer) are operational.	
2	NFV-MANO and VNF/VNFC instances are operational.	
3	The Policy Agent is exposing a Policy management interface towards OSS.	

4.3.12.5 Post-conditions

Table 4.3.12.5-1 describes the post-conditions of this use case.

Table 4.3.12.5-1: Policy agent function, post-conditions

#	Post-condition	Description
1	A policy has been enforced by the Policy Agent function related to the Traffic Enforcer VNF Generic OAM function.	
2	The operator has received the status of the policy transfer and the policy enforcement.	

4.3.12.6 Operational Flows

Table 4.3.12.6-1 describes the base flow of this use case.

Table 4.3.12.6-1: Policy agent function, base flow

#	Actor/Role	Description
1	Operator -> OSS/BSS	Operator is requesting via OSS/BSS to covey and apply a policy related to the Traffic Enforcer (e.g. restrict traffic to a specific set of VNFCs during an upgrade).
2	OSS/BSS -> Policy Agent	OSS/BSS passes the policy to the Policy Agent through a Policy management interface exposed by the Policy Agent. See note 1.
3	Policy Agent -> OSS/BSS -> Operator	The Policy Agent returns the result of the policy transfer request to the Operator via OSS/BSS.
4	Policy Agent -> Traffic Enforcer	The Policy Agent is acting as a PF and is responsible for the policy enforcement of the traffic change operation (e.g. restricting traffic to a specific set of VNFCs). For this, the Policy Agent subscribes to notifications related to events performed by corresponding VNF generic OAM functions. See notes 2, 3 and 4.
5	Policy Agent -> OSS/BSS -> Operator	The Policy Agent informs the operator through OSS/BSS about the enforcement of the policy. See note 2.
NOTE 1: The appropriate identifiers are used in the request, to identify the VNF Generic OAM function the policy targets. NOTE 2: Notification between entities can be sent through the Notification manager VNF generic OAM function. NOTE 3: PF operations executed by the Policy agent are not tailored to as specific solution (e.g. the use of a specific interface or mechanism) and different ways can be considered on how policies are enforced. NOTE 4: The Policy agent only enforces the policy, and if some further execution based on derived actions is needed, those executions are performed by other entities using existing interfaces. The Policy agent can resolve conflicts of the policies he executes, but it cannot resolve conflicts of policies executed by other entities.		

4.4 Use cases related to additional functionalities of VNF generic OAM functions

4.4.1 Use cases: Management aspects of VNF connectivity

4.4.1.1 Introduction

A service mesh can be understood as a -service-aware communication layer used to handle inter service communication. Examples of service mesh frameworks are Istio®/Envoy® and linkerd®. Their realization typically involves a centralized control plane and distributed data plane elements, supporting the following basic principles of operation:

NOTE 1: Istio® is a registered trademark of The Linux Foundation in the United States and/or other countries.

NOTE 2: Envoy® is a registered trademark of The Linux Foundation in the United States and/or other countries.

NOTE 3: Linkerd® is a registered trademark of The Linux Foundation in the United States and/or other countries.

- On the data plane, on behalf of each service all communication aspects (inbound and outbound) are delegated to a data plane element acting as an intelligent proxy used to perform operations like health checking, traffic steering and metrics exposure.
- On the control plane a centralized control entity is used to configure the data plane elements and perform operations like service discovery, certificate management and metrics collection for each service.

In OS container-based deployments, data plane elements are often deployed as a user space process running in an OS container within the same set of OS container(s) as the actual service. As an example, this deployment option is also known as a sidecar proxy , in the case of Kubernetes®. Another deployment option consists in deploying data plane elements in the kernel space in the form an eBPF (Extended Berkeley Packet Filter) program or as a combination of both a light sidecar proxy and an eBPF program.In ETSI GR NFV-IFA 037 [i.12], Solution #2D is described regarding container-based VNFs with service mesh "sidecars" support, considering the use of a Service Communication Proxy (SCP). According to ETSI TS 123 501 [i.14], an SCP is responsible for forwarding and routing to destination NF/NF service and perfectly aligns with service mesh principles of operation.

For the sidecar proxy association with a VNF, in ETSI GR NFV-IFA 037 [i.12], two options are considered:

- Option A: The VNF Package, besides the application related files, also includes and/or refers to the related OS container software images of the sidecar proxy:
  - In case the proxy is deployed as an OS container, a reference can be included in the swImageDesc attribute of the VNFD according to ETSI GS NFV-IFA 011 [i.15].
- Option B: The VNF Package of the VNF does not refer to specific software images for the sidecar proxy. In that case NFV-MANO would be responsible for the association of the VNF with the corresponding sidecar proxy:
  - In this case, association management is made on the NSD level, either with VNF as a form of the service mesh, or as new capabilities of the NS VL construct.Similar options can be envisioned when the data plane element is an eBPF program.In Kubernetes® the case of deploying sidecar proxies using CRDs is also possible (for example an envoy proxy can be installed as sidecar container). However, management of CRDs is not yet supported by the ETSI NFV framework. See Challenge #C.2 CRDs in ETSI NFV in ETSI GR NFV-IFA 046 [i.13] and Solution SOL-C2-1 (CRDs for containerized NFs) in the same referenced document for a possible remedy.

In the following use cases, it is described how the VNF generic OAM functions framework can be used to support operations related to connectivity between the different VNFs/VNFCs which are part of the service mesh.

For all the following use cases, it is considered that irrelevant of which solution is adopted for the data plane elements association with the VNF, this element is available and responsible for all the inbound and outbound traffic handling on behalf of the VNF/VNFCs.

4.4.1.2 Use case: Add VNF to the service mesh and establish connectivity using the Network Configuration Manager Function

4.4.1.2.1 Introduction

This use case is about adding a VNF to a service mesh. It is assumed that both the VNF and the associated data plane element are deployed and instantiated and the overall service mesh system and its control plane are operational.

VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers or a virtual machine.

4.4.1.2.2 Actors and roles

Table 4.4.1.2.2-1 describes the use case actors and roles.

Table 4.4.1.2.2-1: Add VNF to the service mesh and establish connectivity, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system including the NFV-MANO functional entities, the VNFs, the VNF generic OAM functions, the underlying infrastructure and the service mesh.
2	OSS/BSS	The entity instantiating and managing the VNF, through NFV-MANO. It also able to manage the service mesh. See note 1.
3	VNF/VNFC	The entities hosting the NF application, and which are associated with a service mesh data plane element .
4	Service Mesh Control VNF	The VNF entity hosting the service mesh control plane. It interacts with the service mesh data plane element through the Network Configuration manager to configure routing and managing traffic. Configuration is also about handling redirection of traffic and changing path to different VNFC.
5	NFV-MANO	The entity instantiating and managing the VNFs. See note 2.
6	Network configuration manager	The entity that manages the distribution of the target network configuration to the VNF/VNFC(s).
NOTE 1: The mechanisms by which OSS is able to manage the service mesh are out of scope of this use case description.		
NOTE 2: In case the service mesh control plane is deployed as a VNF, like also the service mesh data plane element are deployed as VNF containers, these are also managed by NFV-MANO.		

4.4.1.2.3 Trigger

Table 4.4.1.2.3-1 describes how the operational flow for this use case is triggered.

Table 4.4.1.2.3-1: Add VNF to the service mesh and establish connectivity, trigger

Trigger	Description
Operator is requesting to add the VNF in service mesh.	

4.4.1.2.4 Pre-conditions

Table 4.4.1.2.4-1 describes the pre-conditions of this use case.

Table 4.4.1.2.4-1: Add VNF to the service mesh and establish connectivity, pre-conditions

#	Pre-condition	Description
1	Service mesh system is operational.	
2	NFV-MANO environment is operational.	
3	The necessary VNF generic OAM functions (e.g. VNF network configuration manager) are operational.	
4	VNF Package is onboarded and the VNF instance to be connected to the service mesh is available.	Both VNF based and container based are supported.

4.4.1.2.5 Post-conditions

Table 4.4.1.2.5-1 describes the post-conditions of this use case.

Table 4.4.1.2.5-1: Add VNF to the service mesh and establish connectivity, post-conditions

#	Post-condition	Description
1	The VNF is added in service mesh.	
2	VNF traffic rules are configured in the service mesh. The network traffic management in VNFC work as intended by the service mesh control plane.	

4.4.1.2.6 Operational Flows

Table 4.4.1.2.6-1 describes the base flow of this use case.

Table 4.4.1.2.6-1: Add VNF to the service mesh and establish connectivity, base flow

#	Actor/Role	Description
1	Operator -> OSS/BSS	Operator is triggering to add a VNF in the service mesh.
2	OSS/BSS-> Network configuration manager	OSS/BSS instructs the Network configuration manager to prepare the network at the service mesh data plane element (IP addressing, namespaces etc.).
3	Network configuration manager	The network configuration is pushed to the service mesh data plane element.
4	Service Mesh Control VNF	After the network is configured (with appropriate IP addresses and namespaces etc.) the service mesh control plane can automatically detect and add the VNF in the service mesh.

4.4.1.3 Use case: Update network configuration in a Service mesh

4.4.1.3.1 Introduction

This use case is about updating network configuration for VNFs/VNFCs which are part of a service mesh. It is assumed that the overall service mesh system (data plane and control plane) is operational and VNFs are deployed, instantiated and added in the service mesh. The VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers or a virtual machine.

4.4.1.3.2 Actors and roles

Table 4.4.1.3.2-1 describes the use case actors and roles.

Table 4.4.1.3.2-1: Update network configuration in a Service mesh, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system including the NFV-MANO functional entities, the VNFs, the VNF generic OAM functions, the underlying infrastructure and the service mesh.
2	OSS/BSS	The entity instantiating and managing the VNF through NFV-MANO. It also able to manage service mesh. See note 1.
3	VNF/VNFC	The entities hosting the NF application, and which are associated with a service mesh data plane element .
4	Service mesh control VNF	The VNF entity hosting the service mesh control plane. It interacts with the service mesh data plane element through the Network configuration manager to configure routing and managing traffic. Configuration is also about handling redirection of traffic and changing path to different VNFC.
5	NFV-MANO	The entity instantiating and managing the VNFs. See note 2.
6	VNF Metrics aggregator	The entity that collects the metrics from the VNF/VNFC/NFV-MANO/NFVI and exposes interfaces towards the Operator or delegated entities (e.g. NFVO).
NOTE 1: The mechanisms by which OSS is able to manage the service mesh are out of scope of this use case description. NOTE 2: In case the service mesh control plane is deployed as a VNF, like also the service mesh data plane elements are deployed as VNF containers, these are also managed by NFV-MANO.		

4.4.1.3.3 Trigger

Table 4.4.1.3.3-1 describes how the operational flow for this use case is triggered.

Table 4.4.1.3.3-1: Update network configuration in a Service mesh, trigger

Trigger	Description
Operator is requesting to update network configuration for a VNF which is part of the service mesh.	

4.4.1.3.4 Pre-conditions

Table 4.4.1.3.4-1 describes the pre-conditions of this use case.

Table 4.4.1.3.4-1: Update network configuration in a Service mesh, pre-conditions

#	Pre-condition	Description
1	Service mesh system is operational.	
2	NFV-MANO environment is operational.	
3	The necessary VNF generic OAM functions (e.g. VNF network configuration manager) are operational.	
4	The VNF/VNFCs are operational and already part of the service mesh.	

4.4.1.3.5 Post-conditions

Table 4.4.1.3.5-1 describes the post-conditions of this use case.

Table 4.4.1.3.5-1: Update network configuration in a Service mesh, post-conditions

#	Post-condition	Description
1	The network configuration for the VNF is updated in service mesh and the network is operational.	

4.4.1.3.6 Operational Flows

Table 4.4.1.3.6-1 describes the base flow of this use case.

Table 4.4.1.3.6-1: Update network configuration in a Service mesh, base flow

#	Actor/Role	Description
1	Operator -> OSS/BSS	Operator is triggering to update a VNF network configuration in the service mesh.
2	OSS/BSS → Service Mesh Control VNF	OSS/BSS send request to service mesh control plane to update the network configuration at the service mesh of a VNF for application related traffic (IP addressing, namespaces, gateway, DNS name, alert settings etc.).
3	Service Mesh Control VNF-> NFV-MANO	The network configuration is updated in service mesh.
4	VNF Metrics aggregator	The VNF Metrics aggregator continues to collect metrics from the newly updated interface and send metrics to authorized consumers (e.g. NFV-MANO, VNF generic OAM functions, OSS etc). See note.
NOTE: Different means can be used to inform the VNF metrics aggregator about the identification of the interface (e.g. IP address, DNS gateway etc.).		

4.4.1.4 Use case: Intra-NFVI-PoP network connectivity testing

4.4.1.4.1 Introduction

Connectivity testing comprises two main cases:

a) testing reachability between a sender and a receiver; and

b) testing the quality of communication by means of QoS related metrics.

In both cases multilayer analysis is typically applied, since in case of a problematic communication many different causes can be considered (e.g. a node is down, a port is down, a virtual interface is down, there is congestion in the network, etc.).

Another dimension is about the network coverage the testing is about, and two main cases can also be considered:

- *Next hop reachability:* from the end node perspective this is about testing connectivity with the access network device. For intermediate network devices this is about testing connectivity with the adjacent network devices (e.g. switches, routers etc.). Usually considers L1/L2 connectivity testing (e.g. LLDP based) or L3 testing (e.g. ICMP messages).

- *End-to-end connectivity:* is about multi-layer testing regarding the communication between the sender and the receiver. Regarding QoS analysis sophisticated tools can be used related to network telemetry and Deep Packet Inspection (DPI). Besides ICMP messaging tools like tcpdump, Wireshark® etc. are widely used to analyse the end-to-end flows.

This use case is about the way VNF generic OAM functions can be used to support network connectivity testing for VNFs/VNFCs. The scenario described is related to on-demand testing driven by the operator or NFV-MANO management entities or VNF generic OAM functions during or after VNF onboarding. The mechanisms described are applicable also in the case of the VNF being part of the service mesh. In that case the scenario can be built upon use case "Add VNF to the service mesh and establish connectivity" described in clause 4.4.1.2. The mechanisms by which OSS/BSS is able to manage the service mesh are out of scope of this use case description.

From a VNF/VNFC point of view, connectivity testing aspects to consider are the following:

- which layer the test is about (e.g. application layer, IP, etc.);
- which network coverage is considered (e.g. sender to the next hop or to a gateway or to the final receiver);
- how the test is configured and started; and
- how the relevant metrics and logs are collected and reported.

The following assumptions are considered:

- A VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers or a virtual machine.
- If service mesh-based communications are used, the overall service mesh system (data plane and control plane) is operational and is within the NFVI-PoP boundaries.
- Network testing is about data-plane connectivity related to non-management traffic (e.g. VNFC-to-VNFC, VNF-to-VNF, VNF-to-NFVI PoP gateway, VNFC-to-DNS server, etc).
- The connectivity test is managed by the VNF generic OAM function VNF testing manager introduced in clause 4.3.11.
- The network test can be triggered on-demand or automatically due to a VNF LCM operation.
- The network test can be triggered by the operator or NFV-MANO management entities or VNF generic OAM functions.
- The network test can be triggered during or after VNF instantiation on the NFV system.
- For the case of service mesh, the network test can be triggered during the process of establishing the connectivity of the VNF instance to the service mesh, or after VNF instance is already connected.

In the following description the actors, pre-conditions, post-conditions and flow are described for the case where the connectivity test is issued by the operator after the VNF has been instantiated (i.e. the relevant CPs have been successfully configured). The description is about testing within the NFVI-PoP boundaries.

4.4.1.4.2 Actors and roles

Table 4.4.1.4.2-1 describes the use case actors and roles.

Table 4.4.1.4.2-1: Intra-NFVI-PoP network connectivity testing, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system including the NFV-MANO functional entities, the VNFs, the VNF generic OAM functions, the underlying infrastructure. Is also the entity which triggers the connectivity test.
2	OSS/BSS	The entity instantiating and managing the VNF through NFV-MANO and through VNF generic OAM functions.
3	VNF/VNFC	The entities hosting the NF application, whose communication and interconnectivity is about to be tested.
4	NFV-MANO	The entity instantiating and managing the VNFs.
56	VNF testing manager	The entity which configures and starts the connectivity tests. See clause 4.3.11 for a description of the VNF testing manager.

4.4.1.4.3 Trigger

Table 4.4.1.4.3-1 describes how the operational flow for this use case is triggered.

Table 4.4.1.4.3-1: Intra-NFVI-PoP network connectivity testing, trigger

Trigger	Description
Operator is requesting the VNF testing manager to start a connectivity test for a VNF/VNFC.	The request could also be issued automatically by management systems, e.g. based on some operational policy or as a result of completing certain VNF LCM operations.

4.4.1.4.4 Pre-conditions

Table 4.4.1.4.4-1 describes the pre-conditions of this use case.

Table 4.4.1.4.4-1: Intra-NFVI-PoP network connectivity testing, pre-conditions

#	Pre-condition	Description
1	NFV-MANO environment is operational.	
2	The necessary VNF generic OAM functions (e.g. the VNF testing manager and the VNF network configuration manager) are operational.	
3	The VNF instances are connected to the network. See note.	
NOTE: VNF/VNFCs network configuration can be made through the Network configuration manager. See Use case described in clause 4.3.8.		

4.4.1.4.5 Post-conditions

Table 4.4.1.4.5-1 describes the post-conditions of this use case.

Table 4.4.1.4.5-1: Intra-NFVI-PoP network connectivity testing, post-conditions

#	Post-condition	Description
1	The network connectivity and network performance for the VNF/VNFCs connection are properly tested and the report is available to the operator.	

4.4.1.4.6 Operational Flows

Table 4.4.1.4.6-1 describes the base flow of this use case.

Table 4.4.1.4.6-1: Intra-NFVI-PoP network connectivity testing, base flow

#	Actor/Role	Description
1	Operator -> OSS/BSS	Operator is triggering a network connectivity test for a VNF instance with another VNF instance.
2	OSS/BSS-> VNF testing manager	OSS/BSS instructs the VNF testing manager to test network connectivity for the identified VNF instances.
3	VNF testing manager	The VNF testing managerconfigures the connectivity test and starts the connectivity test. See note 1.
4	VNF testing manager-> OSS/BSS	Once the connectivity test is completed the VNF testing managercan report back to OSS/BSS the test result. See note 2 and note 3.
NOTE 1: For the case of end-to-end testing, both sender VNF and receiver VNF might need to be configured. NOTE 2: A notification can be send also using the Notification Manager VNF generic OAM function. Other authorized consumers may be also notified like NFV-MANO entities or other VNF generic OAM functions NOTE 3: The VNF testing manager can collect the test results from the application/tool used to run the connectivity test. Other alternatives are also possible.		

4.4.1.5 Use case: Inter-NFVI-PoP network connectivity testing

4.4.1.5.1 Introduction

As in the previous use case described in clause 4.4.1.4, connectivity testing can be related to next-hop reachability and/or end-to-end connectivity. It is also related to both reachability testing and communication quality and performance testing. In the light of VNF generic OAM framework, the relevant functions can support connectivity end-to end testing for both the underlay and the overlay networks.

When the test is within the NFVI-PoP boundaries, then the endpoints for both the sender and the receiver are within the same administration domain. Like in use case described in clause 4.4.1.4 this means that the relevant configuration for the test can be tuned by single entity like the VNF testing manager to perform the test (i.e. configure, execute and reporting collection).

Connectivity options and technologies for multi-domain NFV-MANO has been considered by ETSI GR NFV-IFA 028 [i.23] and ETSI GR NFV-IFA 035 [i.26] . Also Annex E of ETSI GS NFV-SOL 005 [i.24] provides additional information about the different network gateway management models for multi-site service connectivity, which illustrate different boundary management possibilities.

In this use case the scenario where senders and receivers reside in different NFVI-PoPs which are under the control of different operators (see Figure 4.4.1.5.1-1) is analyzed.

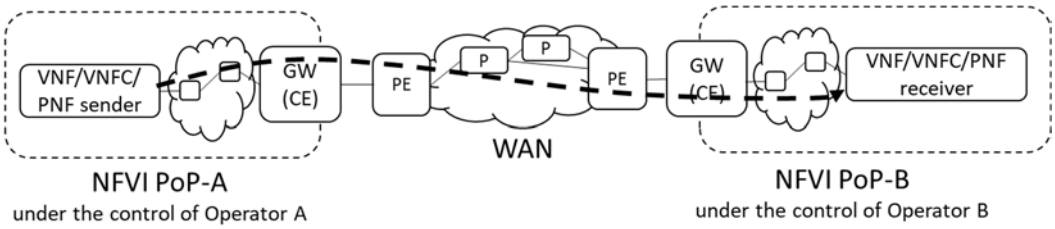

Figure 4.4.1.5.1-1: network testing for senders and receivers in different NFVI-PoPs

NOTE 1: The case of Service Mesh over multi-NFVI PoPs is not studied in the present document.

The following challenges can be identified regarding end-to-end connectivity testing in a multi-operator environment:

1) Who defines the endpoints and how these are communicated to the other operator. The actual communication path that is to be traversed until the destination is reached, passes by different network segments (e.g. VNF/VNFC to CE gateway, CE to PE, etc.). In case the receiver can only expose an internal IP address, then the test cannot be performed in an end-to-end fashion. In case an overlay exists (e.g. L2VPN) then testing could involve both underlay testing aspects (e.g. reachability of the CE) but also overlay aspects (e.g. reachability of the final receiver). In both cases, the level of information exposed depends on operator B policies.

2) How the test is triggered in both the domains.

3) How the testing results are collected from both the domains.

For this use case the following assumptions are considered:

1) A VNF is composed of VNFCs and each VNFC is deployed on a group of OS containers or a virtual machine.

2) WAN is under the control of an external OSS through WIM.

3) Network testing is about data-plane connectivity related to non-management traffic (e.g. VNFC-to-VNFC, VNF-to-VNF, VNF-to-NFVI-PoP gateway, VNFC-to-DNS server, etc).

4) The network test can be triggered on-demand or automatically due to a VNF LCM operation.

5) The network test can be triggered by an operator or NFV-MANO management entities or other VNF generic OAM functions.

6) The network test can be triggered by one of the operators during or after VNF instantiation on the NFV system.

7) At each NFVI-PoP a VNF testing manager VNF generic OAM function belonging to the Operator is used to perform connectivity testing related operations (e.g. endpoint configuration, installation of libraries etc.). The two VNF testing managers belonging to different operators are expected to communicate either directly or via some kind of proxy or gateway to exchange connectivity test related information (e.g. IP addresses and ports to be used).

NOTE 2: This use case is tailored to the case where the VNF testing manager VNF generic OAM function is shared between multiple VNFs. The description is not applicable to the case where generic OAM functions are deployed inside the VNF.

4.4.1.5.2 Actors and roles

Table 4.4.1.5.2-1 describes the use case actors and roles.

Table 4.4.1.5.2-1: Inter-NFVI-PoP network connectivity testing, actors and roles

#	Actor	Description
1	Operators A and B	A human being or an organization that operates the system including the NFV-MANO functional entities, the VNFs, the VNF generic OAM functions, the underlying infrastructure. One of the operators triggers the connectivity test.
2	OSS/BSS	The entity instantiating and managing the VNFs through NFV-MANO and through VNF generic OAM functions. Each operator has its own OSS/BSS.
3	VNF/VNFC	The entities hosting the NF application, whose communication and interconnectivity is about to be tested.
4	NFV-MANO	The entity instantiating and managing the VNFs. One NFV-MANO system per operator is assumed.
5	VNF testing managers A and B	The entities which configure, start the connectivity test and collect test results. The VNF testing manager A is under the control of Operator A and the VNF testing manager B is under the control of operator B. VNF testing manager A and VNF testing manager B are able to communicate.

4.4.1.5.3 Trigger

Table 4.4.1.5.3-1 describes how the operational flow for this use case is triggered.

Table 4.4.1.5.3-1: Inter-NFVI-PoP network connectivity testing, trigger

Trigger	Description
Operator A is requesting to start a connectivity test for a VNF/VNFC where the receiver VNF/VNFC resides in an external NFVI-PoP managed by another operator B.	The request could also be issued automatically by management systems, e.g. based on some operational policy or because of completing certain VNF LCM operations.

4.4.1.5.4 Pre-conditions

Table 4.4.1.5.4-1 describes the pre-conditions of this use case.

Table 4.4.1.5.4-1: Inter-NFVI-PoP network connectivity testing, pre-conditions

#	Pre-condition	Description
1	NFV-MANO environment is operational in both operator A and operator B NFVI-PoPs.	
2	The necessary VNF generic OAM functions (e.g. VNF testing managers) are operational.	
3	The VNF instance in NFVI-PoP (operator A) and the receiver (operator B) are connected to the network. See note.	
4	VNF testing manager A and VNF testing manager B are able to communicate with each other (e.g. directly or through a proxy).	
NOTE: VNF/VNFCs network configuration can be made through the Network configuration manager. See use case described in clause 4.3.8.		

4.4.1.5.5 Post-conditions

Table 4.4.1.5.5-1 describes the post-conditions of this use case.

Table 4.4.1.5.5-1: Inter-NFVI-PoP network connectivity testing, post-conditions

#	Post-condition	Description
1	The network connectivity and network performance for the VNF/VNFCs are tested and the report is available to the operator A and B.	
2	The testing report is available to the calling entity (e.g. OSS, or a NFV-MANO entity due to LCM).	

4.4.1.5.6 Operational Flows

Table 4.4.1.5.6-1 describes the base flow of this use case.

Table 4.4.1.5.6-1: Inter-NFVI-PoP network connectivity testing, base flow

#	Actor/Role	Description
1	Operator A -> OSS/BSS A	Operator A is triggering a network connectivity test for a VNF instance in NFVI-PoP under the control of Operator A, with another VNF instance in another NFVI-PoP under the control of operator B.
2	OSS/BSS-> VNF testing manager A	OSS/BSS A instructs the VNF testing manager A to test network connectivity for the identified VNF instances.
3	VNF testing manager A->VNF testing manager B	VNF testing manager A communicates with the VNF testing manager B to configure the endpoint accordingly for the test (e.g. start a service or a daemon). See note 1.
4	VNF testing manager A and VNF testing manager B	Both VNF testing managers configure the connectivity test and VNF testing manager A starts the connectivity test.
5	VNF testing manager B-> VNF testing manager A	Once the connectivity test is completed the VNF testing manager B can report back to VNF testing manager A the test results as seen by the receiver.
6	VNF testing manager A-> OSS/BSS A and VNF testing manager B-> OSS/BSS B	VNF testing manager A and VNF testing manager B can report back to OSS/BSS A and OSS/BSS B respectively the test result. See note 2.
NOTE 1: Communication between VNF testing managers can be achieved by other means (e.g. through a testing broker service proxy/gateway). NOTE 2: A notification of the result can be sent using the Notification Manager VNF generic OAM function. Other consumers can also be notified like NFV-MANO entities or other VNF generic OAM functions.		

4.4.2 Use cases: VNF generic OAM functions for autonomous management

4.4.2.1 Overview

The three key mechanisms used to enable automation in NFV-MANO, as currently specified in referenced documents, are Intent Management, MDA and Policy management. Intent management and MDA are firstly introduced in ETSI GR NFV-IFA 041 [i.11], while the relevant interfaces specifications are in ETSI GS NFV-IFA 050 [i.18] and ETSI GS NFV-IFA 047 [i.16], respectively. A policy information model is specified in ETSI GS NFV-IFA 048 [i.17], and relevant policy management interfaces are exposed by each NFV-MANO FB and specified in associated reference point interface specifications, such as ETSI GS NFV-IFA 013 [i.9], ETSI GS NFV-IFA 007 [i.10], ETSI GS NFV-IFA 008 [i.5] for NFVO and VNFM; the policy management interface supports relevant CRUD operations on policies.

In the context of the VNF generic OAM functions, data analysis mechanisms have been introduced in the VNF metrics analyser and the Log analyser functions. However, the interplay of these two functions with the rest of the NFV-MANO automation mechanisms has not been delineated.

In the following clauses, use cases elaborate on the different ways VNF generic OAM functions can interwork with the relevant NFV-MANO automation entities.

4.4.2.2 Use case: VNF generic OAM functions in automated Network Service alarm analysis (without the Log analyser and the VNF metrics analyser)

4.4.2.2.1 Introduction

This use case uses as a baseline the *Network service alarm incident analysis* use case from ETSI GR NFV-IFA 041 [i.11] (clause 5.3.2). For this use case the MDAF is performing the necessary data collection via the NFVO, as described in clause 7.2.2 of ETSI GR NFV-IFA 041 [i.11].

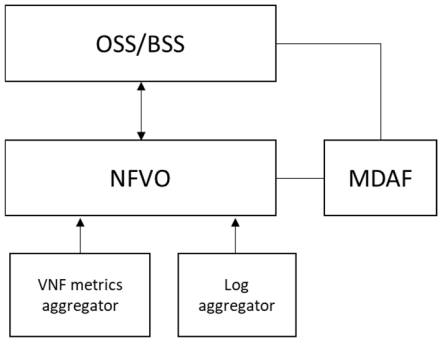

Figure 4.4.2.2.1-1: VNF metrics aggregator and Log aggregator provide data to NFVO

When considering VNF generic OAM functions, log data collected from the Log aggregator and metrics data collected by the VNF metrics aggregator are sent to the NFVO which forwards relevant information to MDAF for further analysis. All the data analytics results are provided by MDAF, which is the responsible entity for the root cause analysis using the appropriate models. In this use case the Log analyser and the Metrics analyser function are not exploited. Also, it is assumed that there is no messaging from VNFs/VNFCs to VNFM through the indicator interface.

4.4.2.2.2 Actors and roles

Table 4.4.2.2.2-1 describes the use case actors and roles.

Table 4.4.2.2.2-1: Management of analytics without Log and VNF metrics analysers involvement, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO, the Log aggregator, the VNF metrics aggregator functions and MDAF.
2	OSS/BSS	The entity that receives the request from the Operator to configure the Log aggregator, the VNF metrics aggregator and MDAF.
3	VNF/VNFC/NFV-MANO/ NFVI	The entities that expose VNF-specific logs and metrics like also logs and metrics from NFV-MANO and NFVI to the Log aggregator and the VNF metrics aggregator.
4	Log Aggregator	The entity that collects logs from the VNF/VNFC/NFV-MANO/NFVI and exposes interfaces towards the Operator or delegated entities (e.g. NFVO).
5	VNF metrics Aggregator	The entity that collects the metrics from the VNF/VNFC/NFV-MANO/NFVI and exposes interfaces towards the Operator or delegated entities (e.g. NFVO).
6	NFVO	Responsible for NS PM/FM, collect logs for alarms and performance measurements in the management domain.
7	MDAF	The function that performs root cause analysis for collected alarms and deteriorated performance measurements.

4.4.2.2.3 Trigger

Table 4.4.1.2.3-1 describes how the operational flow for this use case is triggered.

Table 4.4.1.2.3-1: Management of analytics without Log and VNF metrics analysers involvement, trigger

Trigger	Description
Request from the OSS/BSS to start collecting logs and metrics using the VNF generic OAM functions to perform NS failure or performance degradation root cause analysis.	

4.4.2.2.4 Pre-conditions

Table 4.4.2.2.4-1 describes the pre-conditions of this use case.

Table 4.4.2.2.4-1: Management of analytics without Log and VNF metrics analysers involvement, pre-conditions

#	Pre-condition	Description
1	Log aggregator, VNF metrics aggregator functions and MDAF are instantiated.	
2	NFV-MANO, NFVI and VNF/VNFC instances generate logs that can be collected by the Log aggregator and metrics collected by the VNF metrics aggregator.	

4.4.2.2.5 Post-conditions

Table 4.4.2.2.5-1 describes the post-conditions of this use case.

Table 4.4.2.2.5-1: Management of analytics without Log and VNF metrics analysers involvement, post-conditions

#	Post-condition	Description
1	The report of NS alarm root cause analysis report is returned to the OSS/BSS.	

4.4.2.2.6 Operational Flows

Table 4.4.2.2.6-1 describes the base flow of this use case, etc.

Table 4.4.2.2.6-1: Management of analytics without Log and VNF metrics analysers involvement, base flow

#	Actor/Role	Description
1	Operator -> OSS/BSS -> Log aggregator/VNF metrics aggregator	The Operator requests the retrieval of the logs to the Log aggregator function and metrics from the VNF metrics aggregator function via OSS/BSS using a filter.
2	VNF/VNFC/NFV-MANO/NFVI <-> Log aggregator/VNF metrics aggregator	The Log aggregator collects logs and the VNF metrics aggregator collects metrics from the monitored targets and they aggregate the information.
3	Log aggregator/VNF metrics aggregator -> NFVO	The Log aggregator sends logs and the VNF metrics aggregator sends metrics to NFVO.
4	NFVO	An alarm is detected by NFVO in the logs sent by the Log aggregator to NFVO. The alarm can be due to an error or performance degradation/threshold violation. Additional relevant historical logs and measurements can be requested by NFVO to the Log aggregator and the VNF metrics aggregator, respectively.
5	NFVO -> MDAF	The NFVO sends the logs and metrics indicating the error to the MDAF for analysis of the root cause.
6	MDAF	The MDAF uses its internal AI/ML models and algorithms for analysing the input alarms and performance measurements and derives an analytics report including the root alarm or root cause of the NS fault.
7	MDAF -> NFVO	The MDAF returns the analytics report to the NFVO.
8	NFVO -> OSS/BSS	The NFVO and forwards the result of the root cause analysis to OSS/BSS.

4.4.2.3 Use case: VNF generic OAM functions in automated Network Service alarm analysis (with Log and Metrics analysers involvement)

4.4.2.3.1 Introduction

This use case builds upon the previous use case described in clause 4.4.2.2 and is also about Network service alarm incident analysis. Log data collected from the Log aggregator are sent to the Log analyser, and metrics data collected by the VNF metrics aggregator are sent to the VNF metrics analyser. Both analysers send data to NFVO, who can then forward to MDAF for further analysis. Data analytics are provided partially by MDAF and by the corresponding log and metrics analysers. MDAF is acting as an umbrella automation entity able to further process both logs and metrics analysis results to derive root cause analysis using the appropriate models.

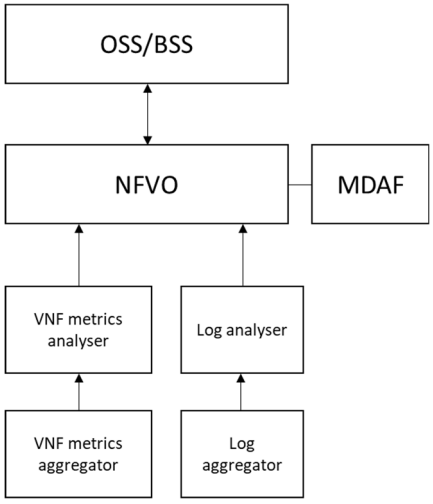

Figure 4.4.2.3.1-1 Interaction between analysers and MDAF through NFVO

In this use case, it is assumed that there is no messaging from VNFs/VNFCs to VNFM through the Indicator interface. It is also considered that the Log analyser and the VNF metrics analyser are aware of the mapping between VNFs and VMs/stes of OS containers (this information can be obtained from the VNFM).

4.4.2.3.2 Actors and roles

Table 4.4.2.3.2-1 describes the use case actors and roles.

Table 4.4.2.3.2-1: Management of analytics using the Log analyser, the VNF metrics analyser and MDAF, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO, the VNF generic OAM functions and MDAF.
2	OSS/BSS	The entity that receives the request from the Operator to configure the VNF generic OAM functions and MDAF.
3	VNF/VNFC/NFV-MANO/ NFVI	The entities that expose VNF logs (e.g. alarms) and metrics to the Log aggregator.
4	Log/VNF metrics aggregator	The entities that collect logs/metrics from the VNF/VNFC/NFV-MANO/NFVI and exposes interfaces towards the Log/VNF metrics analysers respectively.
5	Log/VNF metrics analysers	The entities that analyze the logs/metrics from the VNF/VNFC/NFV-MANO/NFVI and exposes interfaces towards the Operator or delegated entities (e.g. NFVO).
6	NFVO	Responsible for NS PM/FM, collect logs for alarms and performance measurements in the management domain.
7	MDAF	The function that performs root cause analysis for collected alarms and deteriorated performance measurements.

4.4.2.3.3 Trigger

Table 4.4.2.3.3-1 describes how the operational flow for this use case is triggered.

Table 4.4.2.3.3-1: Management of analytics using the Log analyser, the VNF metrics analyser and MDAF, trigger

Trigger	Description
Operator is requesting via OSS/BSS the analysis of the logs and metrics from MDAF.	

4.4.2.3.4 Pre-conditions

Table 4.4.2.3.4-1 describes the pre-conditions of this use case.

Table 4.4.2.3.4-1: Management of analytics using the Log analyser, the VNF metrics analyser and MDAF, pre-conditions

#	Pre-condition	Description
1	Log and VNF metrics aggregators, Log and VNF metrics analysers and MDAF are instantiated.	
2	NFV-MANO, NFVI and VNF/VNFC instances generate logs and metrics that can be collected by the Log aggregator and metrics collected by the VNF metrics aggregator.	

4.4.2.3.5 Post-conditions

Table 4.4.2.3.5-1 describes the post-conditions of this use case.

Table 4.4.2.3.5-1: Management of analytics using the Log analyser, the VNF metrics analyser and MDAF, post-conditions

#	Post-condition	Description
1	The report of NS alarm root cause analysis is available to the NFVO and then to OSS/BSS.	

4.4.2.3.6 Operational Flows

Table 4.4.2.3.6-1 describes the base flow of this use case.

Table 4.4.2.3.6-1: Management of analytics using the Log analyser, the VNF metrics analyser and MDAF, base flow

#	Actor/Role	Description
1	Operator -> OSS/BSS -> Log aggregator/Metrics Aggregator	The Operator requests the retrieval of the logs to the Log aggregator function and metrics from the VNF metrics aggregator function via OSS/BSS using a filter.
2	VNF/VNFC/NFV-MANO/ NFVI <-> Log aggregator/VNF metrics aggregator	The Log aggregator collects logs and the VNF metrics aggregator collects metrics from the monitored targets and they aggregate the information.
3	Log aggregator/VNF metrics aggregator <-> Log analyser/VNF metrics analysers	The Log aggregator sends logs and the VNF metrics aggregator sends metrics to the corresponding analysers from the monitored targets.
4	Log analyser/VNF metrics analysers <-> NFVO	The Log analyser and the VNF metrics analyser pre-process logs and metrics (e.g. abnormal behaviour detection, threshold crossing etc.) and send the analysis results to NFVO. See Note.
5	NFVO	An alarm is detected by NFVO in the log analysis sent by the Log analyser to NFVO. The alarm can be due to an error or performance degradation/threshold violation. Additional relevant historical logs and measurements can be requested by NFVO to the Log aggregator or the VNF metrics aggregator, respectively.
6	NFVO -> MDAF	The NFVO sends the log analysis indicating the error to the MDAF like also relevant metrics for analysis of the root cause.
7	MDAF	The MDAF uses its internal AI/ML models and algorithms by correlating log analysis and metrics analysis results for further analysing the input alarms and performance measurements and derives an analytics report including the root alarm or root cause of the NS fault.
8	MDA -> NFVO	The MDAF returns the analytics report to the NFVO.
7	NFVO -> OSS/BSS	The NFVO informs about the root cause on the NS and forwards the result to OSS.
NOTE: Log analyser and Metrics analyser can send notifications to NFVO through the Notification manager VNF generic OAM function (e.g. about thresholds crossing).		

4.4.2.4 Use case: VNF generic OAM functions in automated Network Service alarm analysis (extending the scope of MDAF)

4.4.2.4.1 Introduction

This use case uses as a baseline MDA deployment option 2.b according to clause 7.2.2 in ETSI GR NFV-IFA 041 [i.11], where the MDAF is represented as new entity with data collection capability from all NFV-MANO FBs.

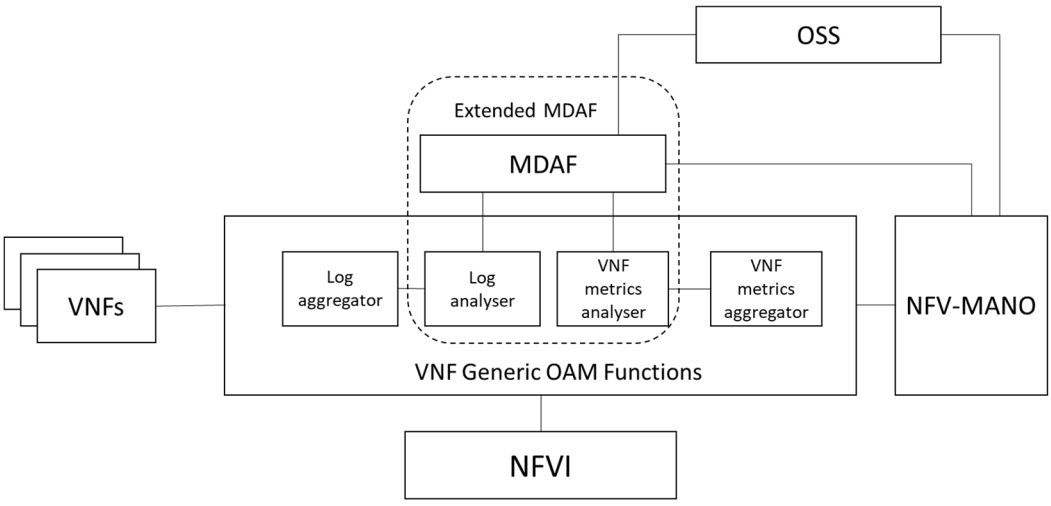

Figure 4.4.2.4.1-1 Interaction between analysers and extended MDAF

In this use case, it is considered that the VNF metrics analyser and the Log analyser can interact directly with MDAF, providing extended automation functionality, without interaction with the NFVO. The VNF metrics analyser and Log analysers, as described in the present document, perform a pre-processing analysis step. Then data are sent to MDAF to perform end-to-end analysis, considering all available log and metrics data from the VNFs, NFVI and NFV-MANO.

In this use case, it is assumed that there is no messaging from VNFs/VNFCs to VNFM through the Indicator interface. It is also considered that the Log and VNF metrics analysers are aware of the mapping between VNFs and VMs/sets of OS containers (this information can be obtained from the VNFM).

A visual representation of the main architectural components assumed in the use case is depicted in Figure 4.4.2.4.1-1. Lines in Figure 4.4.2.4.1-1 represent points of interaction. Communication between NFV-MANO to OSS is made exploiting existing interfaces over the the Os-Ma-nfvo reference point. Other interactions like between VNFs and NFV-MANO, etc. are according to the NFV-MANO architecture and related reference points.

4.4.2.4.2 Actors and roles

Table 4.4.2.4.2-1 describes the use case actors and roles.

Table 4.4.2.4.2-1: Management of analytics using the extended MDAF, actors and roles

#	Actor	Description
1	Operator	A human being or an organization that operates the system and has enabled NFV-MANO, the VNF generic OAM functions and MDAF.
2	OSS/BSS	The entity that receives the request from the Operator to configure the VNF generic OAM functions and MDAF.
3	VNF/VNFC/NFV-MANO/ NFVI	The entities that expose logs (e.g. alarms) and metrics to the Log aggregator.
4	Log/VNF metrics aggregators	The entities that collect logs/metrics from the VNF/VNFC/NFV-MANO/NFVI and exposes interfaces towards the Log/VNF metrics analysers respectively.
5	Log/VNF metrics analysers	The entities that analyse the logs/metrics from the VNF/VNFC/NFV-MANO/NFVI and exposes interfaces towards the Operator or delegated entities (e.g. MDAF).
6	NFVO	Responsible for NS PM/FM, collect notifications from extended MDAF for alarms and performance degradation in the management domain. Communication between NFVO and MDAF is through the MDAF exposed northbound interface.
7	Extended MDAF	The function that performs root cause analysis for collected alarms and deteriorated performance measurements. Log/VNF metrics analysers are part of the extended MDAF. Communication between the MDAF (as defined in ETSI GR NFV-IFA 041 [i.11]) and Log and Metrics analysers (as defined by the present document) is made through the northbound interface exposed by the generic OAM functions.

4.4.2.4.3 Trigger

Table 4.4.2.4.3-1 describes how the operational flow for this use case is triggered.

Table 4.4.2.4.3-1: Management of analytics using the extended MDAF, trigger

Trigger	Description
Operator is requesting via OSS/BSS the analysis of the logs and metrics from MDAF.	

4.4.2.4.4 Pre-conditions

Table 4.4.2.4.4-1 describes the pre-conditions of this use case.

Table 4.4.2.4.4-1: Management of analytics using the extended MDAF, pre-conditions

#	Pre-condition	Description
1	Log aggregator and VNF metrics aggregator, Log analyser and VNF metrics analyser and MDAF are instantiated.	
2	NFV-MANO and VNF/VNFC instances generate logs and metrics that can be collected by the Log aggregator and metrics collected by the VNF metrics aggregator.	

4.4.2.4.5 Post-conditions

Table 4.4.2.4.5-1 describes the post-conditions of this use case.

Table 4.4.2.4.5-1: Management of analytics using the extended MDAF, post-conditions

#	Post-condition	Description
1	The report from the extended MDAF of NS alarm root cause analysis is available to NFVO and to OSS/BSS.	

4.4.2.4.6 Operational Flows

Table 4.4.2.4.6-1 describes the base flow of this use case.

Table 4.4.2.4.6-1: Management of analytics using the extended MDAF, base flow

#	Actor/Role	Description
1	Operator -> OSS/BSS -> MDAF	The Operator requests the *extended MDAF* to continuously track abnormalities and perform root cause analysis based on results collected from the Log analyser and the VNF metrics analyser.
2	VNF/VNFC/NFV-MANO /NFVI <-> Log aggregator/VNF metrics aggregator	The Log aggregator collects logs and the VNF metrics aggregator collects metrics from the monitored targets, and they aggregate the information.
3	Log aggregator/VNF metrics aggregator<-> Log analyser/VNF metrics analyser	The Log aggregator sends logs and the VNF metrics aggregator sends metrics to the corresponding analysers from the monitored targets, and they analyse the information.
4	Log analyser/VNF metrics analyser <-> MDAF	The Log analyser and the VNF metrics analyser pre-process logs and metrics (e.g. abnormal behaviour detection, threshold crossing etc.) and send the analysis results to MDAF. See note.
5	MDAF	The MDAF uses its internal AI/ML models and algorithms by correlating log analysis and metrics analysis results for further analysing the input alarms and performance measurements and derives an analytics report.
6	MDA -> NFVO and OSS/BSS	The MDAF provides the analytics report to the NFVO and OSS.
NOTE: Log analyser and VNF metrics analyser can send notifications to MDAF through the Notification manager VNF generic OAM function (e.g. about thresholds crossing).		

4.4.2.5 Use case: VNF generic OAM functions in automated Network Service health monitoring

4.4.2.5.1 Introduction

This use case is about Network Service health monitoring when considering possible interaction between the MDAF defined by ETSI NFV, the MDA function defined by 3GPP and the VNF generic OAM functions framework.

The Management Data Analytics Function (MDAF) concept has been analysed for NFV environments in ETSI GR NFV-IFA 041 [i.11]. For the mobile network specified by 3GPP there are two entities defined to support data analytics processing. These are the NWDAF (defined in ETSI TS 123 288 [i.19] and MDA function (defined in ETSI TS 128 104 [i.20]). Table 4.4.2.5.1-1 summarizes the basic functionalities and interactions by means of the interfaces produced by each entity.

Table 4.4.2.5.1-1: Summary of functionality and interactions of NWDAF, 3GPP MDA and NFV MDAF

	NWDAF ETSI TS 123 288 [i.19]	3GPP MDA ETSI TS 128 104 [i.20]	NFV MDAF ETSI GR NFV-IFA 041 [i.11] and ETSI GS NFV-IFA 047 [i.16]
Functionality / basic principles	• NWDAF provides analytics to 5GC NFs, and OAM. • NWDAF has no knowledge about NF application logic.	• Provides analytics output, i.e. statistics or predictions, root cause analysis issues, and can also include recommendations to enable necessary actions for network and service operations.	• Used to improve the closed-loop decision making capability of the management and orchestration In the management domain of NFV-MANO (e.g. supporting Network service health analysis, Network service resource utilization analysis.)
Interfaces	• *Nnf interface* (NF is the producer, NWDAF is the consumer) to request subscription to data delivery, etc. • *Nnwdaf interface* (NF is the consumer) to request subscription to network analytics delivery, etc.	• *MnS:* enables any authorized consumer to request and receive analytics. MDA can be also a consumer of *MnS.*	• *MDA-1:* The MDAF offers one or multiple MDA services, which are exposed by corresponding data analytics service interface named MDA-1. MDA-1 is specified in ETSI GS NFV-IFA 047 [i.16].
Example of analytics result	AMF may use analytics on UE mobility and/or UE communication generated by NWDAF to decide connection parameters.	Identify the type of the E2E latency issue, including, RAN-related latency issue, CN-related latency issue, TN-related latency issue, UE-related latency issue and service provider originated latency issue.	Based on AI/ML models NS health analysis is performed by MDAF, which provides a report to NFVO. NFVO can further initiate lifecycle operations (e.g. NS scaling or healing) based on policies.

Since both SDOs, i.e. 3GPP and ETSI NFV, define Management Data Analytics and use the same term (i.e. MDA) for the ease of reading, in this clause *nfv-MDAF* refers to the case of MDA function defined in ETSI GR NFV-IFA 041 [i.11] and *3gpp-MDA* to the case of MDA functions defined in ETSI TS 128 104 [i.20].

Like in use case described in clause 4.4.2.4 (VNF generic OAM functions in automated Network Service alarm analysis -extending the scope of MDAF) the overall *nfv-MDAF* functionality is provided by an internal *MDAF* entity (as described in ETSI GR NFV-IFA 041 [i.11]), interacting with the VNF generic OAM functions. The VNF metrics analyser and Log analyser are used to perform pre-processing analysis. Then data (e.g. PM/FM etc.) are sent to the internal *nfv-MDAF* entity to perform end-to-end analysis, considering all available log and metrics data from both the VNFs and NFV-MANO.

This use case considers that the *3gpp-MDA* is a consumer of the service interface exposed by the *nfv-MDAF*. *The 3gpp-MDA* may also receive input data from NWDAF. Through this interface data analytics about the network context and the NFs operational status can be collected, to assist the overall NS health status reasoning and are reported to OSS/BSS.

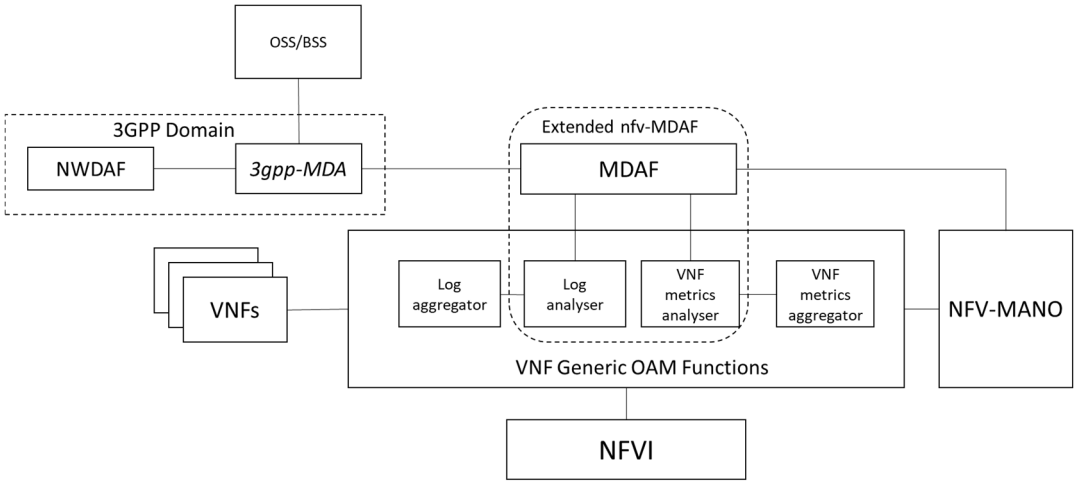

Figure 4.4.2.5.1-1: Overview of interaction between analysers, the nfv-MDAF and 3gpp-MDA

It is assumed that there is no messaging from VNFs/VNFCs to VNFM through the Indicator interface. It is also considered that the Log and Metrics analysers are aware of the mapping between VNFs and VMs/sets of OS containers (this information can be obtained from the VNFM).

In principle NS health reporting to OSS/BSS is possible from the 3gpp-MDA, from nfv-MDAF, or even from an external MDA entity interacting with both the 3gpp-MDA and the nfv-MDAF. The selection is rather an operator implementation choice. Furthermore, different interactions can be foreseen between the nfv-MDAF and the VNF generic OAM functions. For example, one solution could consider the execution of analytics pre-processing in the VNF metrics analyser and log analyser to confine the input data send to nfv-MDAF. Another solution could consider a direct interaction between the aggregator functions (i.e. VNF metrics aggregator and logs aggregator) and MDAF without performing any data pre-processing. In the latter case MDAF is the only responsible entity to perform data analytics based on the available data.

In this use case the VNF generic OAM functions are used to perform data pre-processing with the result send to nfv-MDAF for further analysis. Furtermore, the NS health status is reported by the 3gpp-MDA function to OSS/BSS.

A visual representation of the main architectural components is depicted in Figure 4.4.2.5.1-1. Lines in Figure 4.4.2.5.181 represent points of interaction assumed in the use case. Communication between NFV-MANO to OSS is made exploiting existing interfaces over the the Os-Ma-nfvo reference point. Other interactions like between VNFs and NFV-MANO, etc. are also according to the NFV-MANO architecture and related reference points.

4.4.2.5.2 Actors and roles

Table 4.4.2.5.2-1 describes the use case actors and roles.

Table 4.4.2.5.2-1: Network Service health monitoring using the extended nfv-MDAF and the 3gpp-MDA function, actors and roles

#	Actor	Description
1	Operator	Same description as in use case described in clause 4.4.2.4.
2	OSS/BSS	Same description as in use case described in clause 4.4.2.4.
3	VNF/VNFC/NFV-MANO	Same description as in use case described in clause 4.4.2.4.
4	Log aggregator /VNF metrics aggregator	Same description as in use case described in clause 4.4.2.4.
5	Log analyzer/VNF metrics analyser	Same description as in use case described in clause 4.4.2.4.
6	NFVO	Same description as in use case described in clause 4.4.2.4.
7	nfv-MDAF	Same description as in use case described in clause 4.4.2.4.
8	3gpp-MDA function	Used to expose analytics related to the mobile network. It performs operations as defined in ETSI TS 128 104 [i.20]. It is able to process input data exposed by nfv-*MDAF*.

4.4.2.5.3 Trigger

Table 4.4.2.5.3-1 describes how the operational flow for this use case is triggered.

Table 4.4.2.5.3-1: Network Service health monitoring using the extended nfv-MDAF and the 3gpp-MDA function, trigger

Trigger	Description
Operator is requesting via OSS/BSS the analysis of the logs and metrics from 3gpp-MDAF regarding the health status of a NS.	

4.4.2.5.4 Pre-conditions

Table 4.4.2.5.4-1 describes the pre-conditions of this use case.

Table 4.4.2.5.4-1: Network Service health monitoring using the extended nfv-MDAF and the 3gpp-MDA function, pre-conditions

#	Pre-condition	Description
1	Logs and VNF metrics aggregators, Log and VNF metrics analysers, nfv-MDAF and 3gpp-MDAF are operational.	
2	NFV-MANO and VNF/VNFC instances generate logs and metrics that can be collected by the Log aggregator and metrics collected by the VNF metrics aggregator.	
3	The 3gpp-MDA function is registered as an authorized consumer of analytics send by nfv-MDAF.	

4.4.2.5.5 Post-conditions

Table 4.4.2.5.5-1 describes the post-conditions of this use case.

Table 4.4.2.5.5-1: Network Service health monitoring using the extended nfv-MDAF and the 3gpp-MDA function, post-conditions

#	Post-condition	Description
1	The report from the 3gpp-MDA function of NS health status is available to OSS/BSS.	

4.4.2.5.6 Operational Flows

Table 4.4.2.5.6-1 describes the base flow of this use case.

Table 4.4.2.5.6-1: Network Service health monitoring using the extended nfv-MDAF and the 3gpp-MDA function, base flow

#	Actor/Role	Description
1	Operator -> OSS/BSS -> Log aggregator/VNF metrics aggregator & nfv-MDAF	The Operator requests the nfv-MDAF to report about the health status of a NS to 3gpp-MDA.
2	VNF/VNFC/NFV-MANO <-> Log aggregator/VNF metrics aggregator	The Log aggregator collects logs and the VNF metrics aggregator collects metrics from the monitored targets, and they aggregate the information.

#	Actor/Role	Description
3	Log aggregator/VNF metrics aggregator<-> Log analyser/VNF metrics analyser	The Log aggregator sends logs and the VNF metrics aggregator sends metrics to the corresponding analysers from the monitored targets.
4	Log analyzer/VNF metrics analyser <-> nfv-MDAF	The Log analyser and the VNF metrics analyser pre-process logs and metrics (e.g. abnormal behaviour detection, threshold crossing, etc.) and send the analysis results to nfv-MDAF. See note 1.
5	nfv-MDAF	Uses its internal AI/ML models and algorithms to correlate log analysis and metrics analysis results for the NS (e.g. considering relevant VNFs health status, virtualized resources, etc.).
6	nfv-MDAF->3gpp-MDA function	The nfv-MDAF sends to 3gpp-MDA function a report with the analysis report of the NS health status.
7	3gpp-MDA	The 3gpp-MDA function is capable to process analytics provided by NWDAF but also nfv-MDAF and also uses its internal AI/ML models and algorithms to correlate log analysis and metrics analysis results. It further analyses all inputs and derives an analytics report about the overall NS health status.
8	3gpp-MDA function -> OSS/BSS	The 3gpp-MDAfunction provides the analytics report to OSS.
NOTE: Log analyser and VNF metrics analyser can send notifications to nfv-MDAF through the Notification manager VNF generic OAM function (e.g. about thresholds crossing).		

4.4.2.6 Use case: Extended MDAF and Policy agent for automated traffic rerouting and isolation

4.4.2.6.1 Introduction

The use cases described in clauses 4.4.2.2, 4.4.2.3 and 4.4.2.4 study the coupling and/or inter-working of the different VNF generic OAM functions with MDAF, as studied in ETSI GS NFV-IFA 041 [i.11]. The use case described in clause 4.4.2.5 is about potential coupling and interactions between VNF generic OAM functions, MDAF for NFV-MANO as defined by ETSI GS NFV-IFA 041 [i.11] but also mobile network related MDA as defined by ETSI TS 123 501 [i.14].

In all the use cases the basic assumption is that MDA output is related to results of the analytics and recommended actions provided in an analytics report. In ETSI GS NFV-IFA 047 [i.16], the *AnalyticsOutput* information element of the *DataAnalyticsChangeNotification* operation includes the *recommendedActions* attribute. This indicates that the MDA is not the responsible entity for the actual decision making.

In this use case, the Policy agent introduced in use case Policy Management for VNF generic OAM functions in clause 4.3.12 is used to enable closed-loop control and ease administration tasks when considering the VNF generic OAM functions framework. In more detail, the Policy agent generic OAM function is used to perform the following actions:

- Registers to analytics reports send by the extended MDAF (see ETSI GS NFV-IFA 047 [i.16]).
- As the output of the analytics report only provides a set of recommended actions, the Policy agent selects the appropriate actions to be taken based on PF operations, policies conformance, runtime system information, VNFs status, etc.
- Based on the actions selected, the Policy agent executes these actions (e.g. by calling the appropriate northbound interface of other generic OAM functions). For example, the Policy agent can call the VNF generic OAM function Traffic enforcer to take an action or the Network configuration management function to reconfigure the CPs for a set of VNFs.

In the following, an example use case of automated closed loop-control is described. In more detail, the Policy agent is used to reroute traffic in an automated way, based on analytics report provided by the extended MDAF, by calling the Traffic enforcer (see also Figure 4.4.2.6.1-1). Lines in Figure 4.4.2.6.1-1 represent points of interaction assumed in the use case.

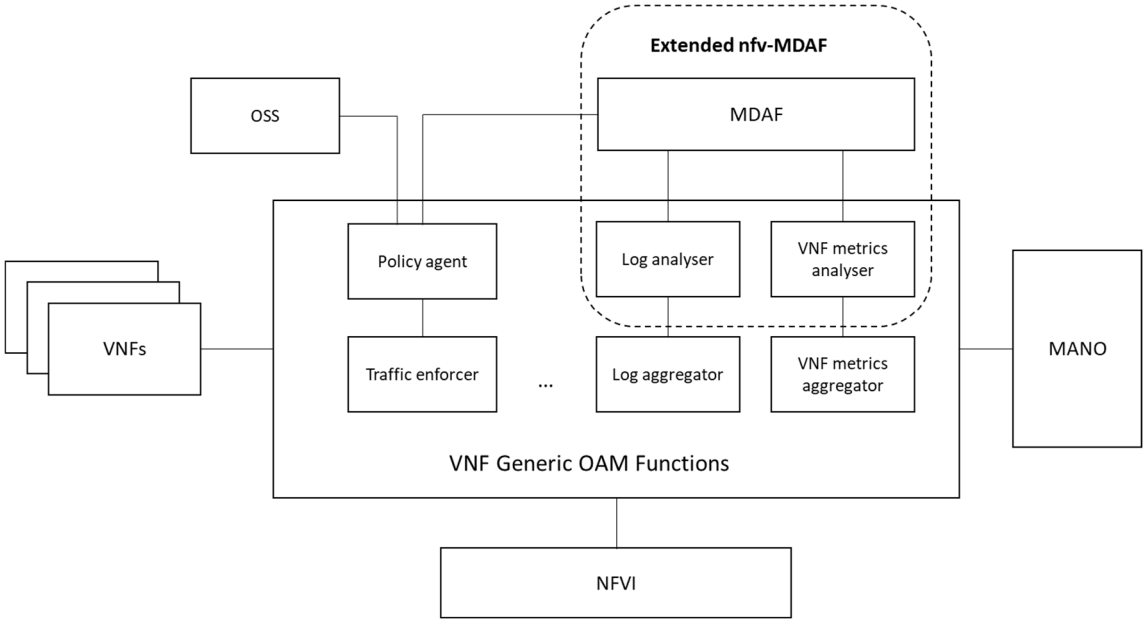

Figure 4.4.2.6.1-1: Extended MDAF and Policy Agent for automated traffic rerouting and isolation

In the context of this use case the following assumptions are considered:

- The Policy agent is considered as a new type of VNF generic OAM function.
- The Policy agent can directly interact with MDAF (or extended MDA), OSS, NFV-MANO, VNF generic OAM functions, NFVI and VNFs.
- Traffic enforcer policies are passed from OSS to the Policy agent which performs relevant PF operations.
- Based on the output of data analytics and all other relevant information (e.g. running configuration, VNF status, policy enforcement, etc.) the Policy agent selects the appropriate action to be taken and calls the corresponding entity which is responsible to execute the action. For example, can call the Traffic enforcer to perform a traffic rerouting and isolation operations.

NOTE: The present use case does not preclude that functionality like the one provided by Policy agent could be considered as part of NFV-MANO (e.g. NFVO). In NFV-MANO the actual decision making can be made by the entity who is using the MDA-1 interface exposed by MDAF (e.g. NFVO or OSS). This is not described in the present document, since the use case focuses on adding more automation functionality into the VNF generic OAM functions framework, by considering the Policy agent as a new type of VNF generic OAM function.

4.4.2.6.2 Actors and roles

Table 4.4.2.6.2-1 describes the use case actors and roles.

Table 4.4.2.6.2-1: VNF generic OAM functions in automated network management,actors and roles

#	Actor	Description
1	Operator	Same description as in use case described in clause 4.4.2.4.
2	OSS/BSS	Same description as in use case described in clause 4.4.2.4.
3	VNF/VNFC/NFV-MANO	Same description as in use case described in clause 4.4.2.4.
4	Traffic enforcer	The entity that blocks and reroutes the traffic of VNFC instances.
5	extended-MDAF	Same description as in use case described in clause 4.4.2.4.
6	Policy agent	The VNF generic OAM function responsible for decision making in an automated way.

4.4.2.6.3 Trigger

Table 4.4.2.6.3-1 describes how the operational flow for this use case is triggered.

Table 4.4.2.6.3-1: VNF generic OAM functions in automated network management, trigger

Trigger	Description
Operator is requesting via OSS/BSS the automated handling of traffic isolation and rerouting by the Policy agent.	

4.4.2.6.4 Pre-conditions

Table 4.4.2.6.4-1 describes the pre-conditions of this use case.

Table 4.4.2.6.4-1: VNF generic OAM functions in automated network management, pre-conditions

#	Pre-condition	Description
1	VNF generic OAM functions (e.g. Log and VNF metrics analysers, Traffic enforcer, Policy agent, etc.) are operational.	
2	NFV-MANO and VNF/VNFC instances generate logs and metrics that can be collected by the Log aggregator and metrics collected by the VNF metrics aggregator.	
3	VNF generic functions can interact with the Policy agent.	
4	Extended-MDAF is able to compile analytics reports and notify the Policy agent for results and recommended actions.	
5	The Policy agent is an authorized consumer of analytics sent by the extended-MDAF and is also authorized to request the Traffic enforcer to perform traffic rerouting and isolation.	

4.4.2.6.5 Post-conditions

Table 4.4.2.6.5-1 describes the post-conditions of this use case.

Table 4.4.2.6.5-1: VNF generic OAM functions in automated network management, post-conditions

#	Post-condition	Description
1	The affected VNFC instances have been successfully blocked, e.g. have been isolated and traffic is re-routed.	
2	Operator has been notified about blocking and rerouting from Traffic enforcer function.	

4.4.2.6.6 Operational Flows

Table 4.4.2.6.6-1 describes the base flow of this use case.

Table 4.4.2.6.6-1: VNF generic OAM functions in automated network management, base flow

#	Actor/Role	Description
1	Operator -> OSS/BSS -> Policy agent	Operator requests via OSS/BSS the automated handling of traffic isolation and rerouting by the Policy agent.
2	VNF/VNFC/NFV-MANO /VNF generic OAM functions/VNF-> extended-MDAF	Extended-MDAF collects data from all possible sources which can be used to support the analytics process.
3	extended-MDAF	Uses its internal AI/ML models and algorithms to correlate log analysis and metrics analysis results for the NS (e.g. considering relevant VNFs status, virtualized resources, etc.).
4	extended-MDAF-> Policy agent	When an analytics report is available a notification is sent to the Policy agentincluding the analytics report and the recommended actions set. See note 1.
5	Policy agent->Traffic enforcer	The Policy agentdecides the action to be taken, identifies the appropriate configuration (if needed) and requests the Traffic enforcer to reroute traffic and perform traffic isolation for the appropriate VNFs. See note 2.
6	Traffic enforcer <-> NFV-MANO	The Traffic enforcer function performs the required traffic rerouting and blocking operations on the VNFC instances by consuming the interfaces exposed by NFV-MANO and reroutes the traffic.
7	Traffic enforcer -> OSS/BSS -> Operator	The Traffic enforcer function returns the result of the traffic blocking and rerouting request to the Operator via OSS/BSS.
NOTE 1: Notification between entities (i.e. NFV-MANO entities and VNF generic OAM functions) can be sent through the Notification manager VNF generic OAM function. NOTE 2: A notification can be sent regarding the action to be taken to other authorized entities (e.g. NFVO, OSS/BSS)		

5 Use Cases analysis

5.1 Overview

This clause provides an analysis of the use cases described in clause 4 of the present document. The analysis of the LCM related use cases focuses on commonalities and differences between VNF generic OAM functions and other VNFs. The analysis of the use cases related to types of VNF generic OAM functions identifies different categories of VNF generic OAM functions providing characteristics and further examples for each group. Use cases related to additional functionalities of VNF generic OAM functions are about support of new connectivity forms (i.e. service mesh) and enabling automation considering interactions between VNF generic OAM functions and MDAF.

5.2 Use cases related to LCM of VNF generic OAM functions

Clause 4.2 provides a representative set of use cases related to LCM of VNF generic OAM functions managed by NFV-MANO. The operational flows described in clause 4.2 indicate that additional procedures are necessary during the lifecycle of VNF or NS instances, e.g.:

- Instantiation of VNF generic OAM functions (see step 2 in Table 4.2.1.6-1).
- Termination of VNF generic OAM functions (see step 2 in Table 4.2.2.6-1).
- Scale up/out of VNF generic OAM functions (see step 8 in Table 4.2.3.6-1).
- Scale down/in of VNF generic OAM functions (see step 12 in Table 4.2.3.6-1).

It is worth noting that the above list of additional procedures is not complete and other use cases for the purpose of onboarding, healing or updating of VNF generic OAM functions are not introduced in the present document. The management and modelling of VNF generic OAM function is left to further detailed specification. For instance, it is expected that new information elements need to be specified to support newly introduced concepts like scaling of VNF generic OAM functions and additional information need to be considered by NFV-MANO at run-time when scaling a VNF instance connected to one or more VNF generic OAM functions.

5.3 Use cases related to types of VNF generic OAM functions

The use cases described in clause 4.3 can be grouped into the following categories of VNF generic OAM functions:

- Performance management:
  - Characteristics:
    - Functions monitoring the performance of individual entities and the overall system to check for signs of performance degradation (e.g. due to equipment failure or overload situations) or to simply monitor any kind of QoS / KPI of interest.
    - If a deviation from the default performance is detected, a notification will be triggered providing sufficient information for the subscribed consumer(s) to take a countermeasure.
    - The functions also provide an interface so that authorized consumers can pull current and historical performance data.
  - Examples:
    - VNF metrics aggregator (see clause 4.3.4).
    - VNF metrics analyser (see clause 4.3.5).
    - Time function (see clause 4.3.6).
- Fault and log management:
  - Characteristics:
    - Functions related to failures handling and maintenance. The functions will collect, process, and analyse logs produced by different entities (e.g. application, container, database access logs) to identify failure situations and to identify information to support the analysis of the failure and its root cause.
    - Issues observed will be raised as notifications based on the configured severity level.
    - The functions also provide an interface so that authorized consumers can pull current and historical log data in a unified manner.
    - Functions in this group may also take actions in case of failure or maintenance situations, e.g. isolate a component.
  - Examples:
    - Log aggregator (see clause 4.3.1).
    - Log analyser (see clause 4.3.2).
    - Traffic enforcer function (see clause 4.3.3).
    - VNF metrics analyser (see clause 4.3.5).
- Configuration management:
  - Characteristics:
    - Functions related to the configuration of a VNF/VNFC. The function will take over certain management aspects and coordinate the setting of the configuration towards the target VNF/VNFCs. The configuration can cover both the NFV-MANO layer (e.g. configure connection points between VNF instances) and/or the application layer (e.g. set application-related thresholds).
    - The functions provide an interface so that authorized consumers can request to the VNF generic OAM function to set the new configuration to the target instances, as well as to check on the status of the requested actions.

  - Example:
    - Network configuration manager function (see clause 4.3.8).
    - VNF configuration manager function (see clause 4.3.10).

- Software modification management:
  - Characteristics:
    - Functions related to the software modification management that e.g. coordinate the distribution of a new software version to the VNF/VNFCs.
  - Example:
    - Upgrade VNF function (see clause 4.3.9).

- Notification management:
  - Characteristics:
    - Function to distribute notifications. The function also allows managing the distribution of notifications, e.g. remove duplicates, keep a notification history, provide a query functionality, allow to silence notifications, etc.
  - Example:
    - Notification manager function (see clause 4.3.7).

- Policy management:
  - Characteristics:
    - Functionality related to conveying and enforcing a type-specific policy related to a VNF generic OAM function or VNF.
  - Example:
    - Policy agent function (see clause 4.3.12).

- Automation management:
  - Characteristics:
    - Function which can be used for the actual decision making to ease administration tasks and offload responsibility from the operator and OSS/BSS. Used to enable closed-loop control through interaction with NFV-MANO, other VNF generic OAM functions and other functions like MDA.
  - Example:
    - Policy agent function (see clause 4.3.12).

- Testing management:
  - Characteristics:
    - Functionality related to communication testing for VNF/VNFC instances, considering cases when different components reside within the same NFVI-PoP and/or different NFVI-PoPs under the control of a single or different operators.
  - Example:
    - VNF testing manager (see clause 4.3.11).

5.4 Use cases related to functionality currently provided by VNFs

From the type of uses cases, it can be seen that the VNF generic OAM functions described are quite typical functions from OAM perspective. As such, most Telco VNFs already currently support many of such functionalities in a vendor-specific way. Many of the listed VNF generic OAM functions provide quite basic and rather simple functionalities (compared to the in large part complex VNFs), while still easing the VNF design when offloaded to a VNF generic OAM function.

Examples of such use cases are:

- Certain log/VNF-specific metrics aggregation and analysis (see use cases in clauses 4.3.1, 4.3.2, 4.3.4 and 4.3.5).
- Notifications (see use case in clause 4.3.7).

5.5 Use cases related to functionality currently provided by OSS/BSS and EM

From an analysis of the user stories related to the use cases, it can also be seen, that having a generic/common way to provide automated management and functionalities in a unified manner, using templates/configuration setting, it is expected to relieve the OSS/BSS and EM from certain tasks.

Example of such use cases are:

- Traffic management (see use case in clause 4.3.3).
- Distribution of configuration data and new software versions (see use cases in clauses 4.3.9 and 4.3.10).
- Time synchronization (see use case in clause 4.3.6).
- Connectivity testing (see use cases in clause 4.4.1.4 and 4.4.1.5).
- Closed-loop control for automated decision making (see use case in clause 4.4.2.6).

5.6 Characteristics of VNF generic OAM functions

Based on the type of VNF generic OAM functions as well as the use cases on the lifecycle of the VNF generic OAM functions, the following characteristics can be identified:

- VNF generic OAM functions interact with the following entities:
  - NFV-MANO, e.g:
    - in the use cases where the VNF generic OAM functions are managed by NFV-MANO (see use cases in clause 4.2); or
    - receiving a request to isolate a containerized workload (see use case in clause 4.3.3); or
    - requesting to perform an operation as a result of a closed-loop control operation performed by the Policy agent (see use case in clause 4.4.2.6); or
    - interacting with MDAF and final processing of analytics data (see use cases in clause 4.4.2).
  - Operator and OSS/BSS, e.g:
    - requesting the VNF generic OAM function to retrieve and process logs from the VNF/VNFCs, NFV-MANO and NFVI (see use cases in clauses 4.3.1, 4.3.2); or
    - requesting the VNF generic OAM function to retrieve and process VNF specific metrics from the VNF/VNFCs and NFV-MANO (see use cases in clauses 4.3.4, 4.3.5); or

        - requesting traffic isolation (see use case in clause 4.3.3); or
        - requesting to distribute configuration to the VNF/VNFCs (see use case in clause 4.3.10);
        - requesting the addition of a VNF to a service mesh (see use case in clause 4.4.1.2);
        - requesting the update of network configuration for a service mesh (see use cases in clauses 4.4.1.3);
        - requesting to execute a connectivity test between VNF/VNFCs (see use case in clause 4.4.1.4 for intra-NFVI-PoP and use case in clause 4.4.1.5 for inter-NFVI-PoP network testing);
        - requesting to distribute a policy for a specific VNF generic OAM function or a set of VNF generic OAM functions through the Policy agent (see use case 4.3.12).
    - VNF/VNFC, e.g:
        - providing logs/VNF-specific metrics (see use cases in clauses 4.3.1 and 4.3.4); or
        - receiving configuration from the VNF generic OAM function (see use case in clause 4.3.10).
    - NFVI/host, e.g:
        - providing logs (see use case in clause 4.3.1); or
        - VNF related metrics (see use case in clause 4.3.4); or
        - running a time client (see use case in clause 4.3.6).
    - Other VNF generic OAM functions, e.g:
        - the Log/VNF metrics analyser processing logs/VNF-specific metrics provided by the Log/VNF metrics aggregator (see use cases in clauses 4.3.2 and 4.3.5); or
        - the Notification manager processing and routing notifications created by other entities (see use cases in clauses 4.3.2, 4.3.5 and 4.3.6);
        - the Policy agent interacting with other VNF generic OAM functions to enable closed-loop control and decision making. See closed-loop automation use case in clause 4.4.2.6;
        - the Policy agent interacting with other VNF generic OAM functions when enforcing a policy. See policy distribution and enforcement use case in clause 4.3.12.
    - Other functions, external both to NFV-MANO and the VNF Generic OAM functions e.g.:
        - MDA exposed by 3GPP according to ETSI TS 128 104 [i.20] (see use cases in clause 4.4.2.5).
- Lifecycle is independent from the entities identified above.

NOTE: The uses cases described in clause 4.2 mention that a VNF generic OAM function is instantiated, if not yet available to be consumed, when required for a VNF, and may be scaled-in or terminated, if the VNF is terminated. However, this does not exclude that a VNF generic OAM function can exist without being consumed.

- Can be consumed/shared northbound by one or multiple services/entities at a time and can handle southbound one or multiple entities/instances at a time.
- Can be consumed by any type of authorized VNF, authorized NFV-MANO functional entity, OSS/BSS, or other authorized VNF generic OAM function.
- Different instances of VNF generic OAM functions can co-exist providing the same functionality.
- Can be built in a modular way, e.g. a "VNF metrics function" could be built from a "VNF metrics aggregator function" and a "VNF metrics analyser function".
- Can be modelled as a VNF or as a new object type.

5.7 Comparison of VNF generic OAM functions and VNF common services

Looking at the use cases described in clause 4 and the characteristics of VNF generic OAM functions listed in the previous clause of the present document, it can be noticed that those use cases and characteristis have certain commonalities with the use cases and characteristics of "VNF common services" as described in clause 5.1 (use cases) and clause 4.2 (characteristics) of ETSI GR NFV-IFA 029 [i.2]. Table 5.7-1 compares the VNF generic OAM functions with the VNF common services with the goal to evaluate e.g. whether VNF generic OAM functions can be realized/designed in a similar / the same way as VNF common services.

Table 5.7-1: Comparison of VNF generic OAM functions and VNF common services

Aspect	VNF generic OAM function	VNF common service
Motivation	• Reduce complexity of VNF design (e.g. decouple VNF from underlying resources, host, network). • Simplify OSS/BSS operation (e.g. generic way for configuration and monitoring) so that OSS/BSS can focus on services management.	• Reduce complexity of NS and VNF design (e.g. services provide common functionalities like messaging, database, logging), so that developers can focus on the application design and user experience. • Simplify for NFVI providers the process of providing their platform to their customers.
Functional scope	OAM functionalities of the VNFs (and partially the EM) (see use cases in clause 4.3).	Protocol messaging, databases, logging, etc. associated to VNF instances (see clause 5.1.2.1 of ETSI GR NFV-IFA 029 [i.2]). See note.
Consumers	OSS/BSS, VNFs, NFV-MANO, and other VNF generic OAM functions.	VNFs, and other VNF common/dedicated services.
Provide services or functions that can be required by many consumers, i.e. functional scope is common to multiple consumers	Yes	Yes
Can be consumed by any type of authorized VNF or authorized other service/function	Yes	Yes
Can be built in a modular way	Yes	Yes
Lifecycle indenpendent of any consumer	Yes	Yes
Can be consumed by multiple consumer instances at a time	Yes	Yes
Can only be terminated in case it is not consumed by any consumer	Yes	Yes
Design options	• Modelled as VNF. • Modelled as new type of object. • Modelled as new type of object specific to PaaS layer.	• Modelled as VNF. • Modelled as new type of NFVI resource. • Modelled as new type of object specific to PaaS layer.
Requires a descriptor, such as VNFD, to provide information about the function/service to the consumers.	Yes	Yes
Recommended additional functions to support the function/service	Service registry, service discovery, service binding.	Service registry, service discovery, service binding.
NOTE: All use cases described in clause 4.3 could be realized as VNF common services.		

6 Framework and potential solutions

6.1 Introduction

This clause describes the framework around VNF generic OAM function(s) and lists potential solutions to realize generic OAM function(s).

6.2 Framework

6.2.1 Overview of interactions

As shown in the use cases and their analysis, the VNF generic OAM functions will provide functionality in a unified manner, which was typically provided/required individually by a majority of VNFs, thereby harmonising those functionalities, easing the management of the different types of VNFs from an Operator's point of view, and simplifying the design of the individual VNFs.

Figure 6.2.1-1 shows the interactions around a VNF generic OAM function identified in clause 5.6. The operator through OSS/BSS can communicate with the VNF generic OAM function to e.g. support distribution of configuration to VNF/VNFCs (see use case in clause 4.3.10). The VNF can also communicate with the VNF generic OAM function to e.g. provide logs related to the VNF which once processed can be consumed by NFV-MANO/VNFM (See use case in clause 4.3.1).

Interactions between VNF generic OAM functions and NFVI are also possible (e.g. logs collection, see use case description in clause 4.3.1). Some interactions between NFVI and VNF generic OAM functions can go through the NFV-MANO, if there is some responsible management function for the NFVI that is responsible for such operation (e.g. logs collection made by PIM).

The generic OAM function also supports certain functionality to support the provisioning, connectivity, and monitoring (e.g. FM, PM) of VNFs.

The architectural splitting of functions is made on the logical level, without making any assumption about its implementation. For instance, Element Management (EM) is shown in Figure 6.2.1-1 as a functional block that comprises element management type of functionality.

NOTE: VNF generic OAM functions can be used to perform EM related operations like VNF configuration and/or exposure of the *Indicator* and *LCM Coordination* interfaces according to ETSI GS NFV-IFA 008 [i.5] in a generic simplified way. VNF generic OAM functions can interact with any type of VNFs and relax the restriction that the configuration of different VNF types is coupled with dedicated management systems (i.e, EMs). For simplicity interactions between VNFs and EMs are not depicted in Figure 6.2.1-1.

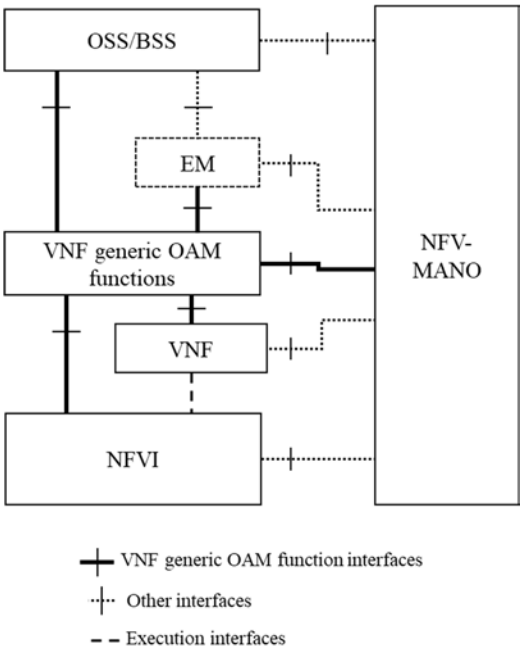

Figure 6.2.1-1: Interactions of the VNF generic OAM functions with different entities in the context of NFV-MANO

6.2.2 Types of functions of generic OAM

As described in clauses 4.3.1 to 4.3.10, VNF generic OAM function can be one out of several types.

Table 6.2.2-1: Name and roles of VNF generic OAM functions

Name of functions	Roles
Log aggregator function	Function which collects and stores the logs from the VNF/VNFC/NFVI/NFV-MANO.
Log analyser function	Function which analyses any type of log entries and can be configured to send notifications based on e.g. statistical processing or threshold crossing.
Traffic enforcer function	Function which blocks and reroutes the traffic of VNFC instances.
VNF metrics aggregator function	Function which collects the VNF-specific metrics.
VNF metrics analyser function	Function which analyses any type of VNF-specific metrics and can be configured to send notifications based on e.g. statistical processing, abnormal behavior detection or threshold crossing.
Time function	Function which ensures the time synchronicity of multiple VNFs and their VNFCs with the master time server in the Operator's network.
Notification manager function	Function which handles notifications, such as alerts, sent by other VNF generic OAM functions.
Network configuration manager function	Function which handles the configuration of the external connectivity of VNFs/VNFCs.
Upgrade VNF function	Function which handles the software upgrade of VNF/VNFC instances to run with new software version.
VNF configuration manager function	Function which handles changes to the configuration of a VNF/VNFC.
Policy agent function	Function used to receive policies for itself but also for any other VNF Generic OAM function and perform policy enforcement operations.
Testing manager function	Functions used to support multilayer testing functionalities up to the application layer related to VNF and NFV-MANO operations.

6.3 Solution A: Introducing generic OAM as a new functional block

6.3.1 Introduction

VNF generic OAM functions have roles related to Performance management, Fault management, Configuration management, Software modification management, and Notification management. In NFV environment, VNF generic OAM functions may be recognized as one Functional Block (FB) which performs operation and maintenance. See Figure 6.3.1-1 for a visual representation.

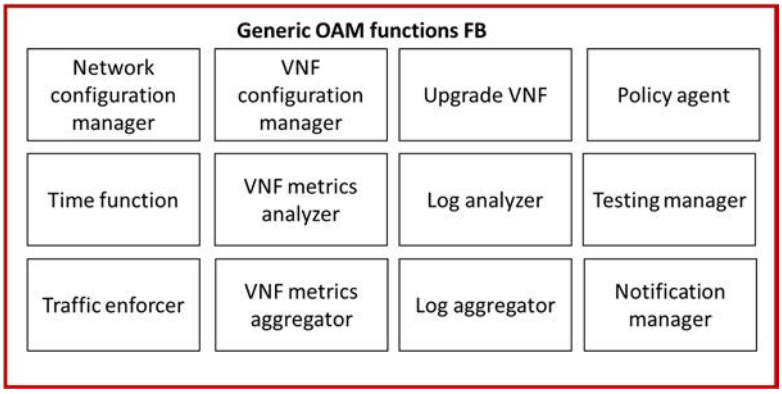

Figure 6.3.1-1: VNF generic OAM functions as a new FB

6.3.2 Internal interactions of each function in generic OAM FB

Generic OAM works as a functional block to operate and maintain VNFs in NFV environment. Inside the Generic OAM FB the functions can interact with each other as illustrated in the example depicted in Figure 6.3.2-1.

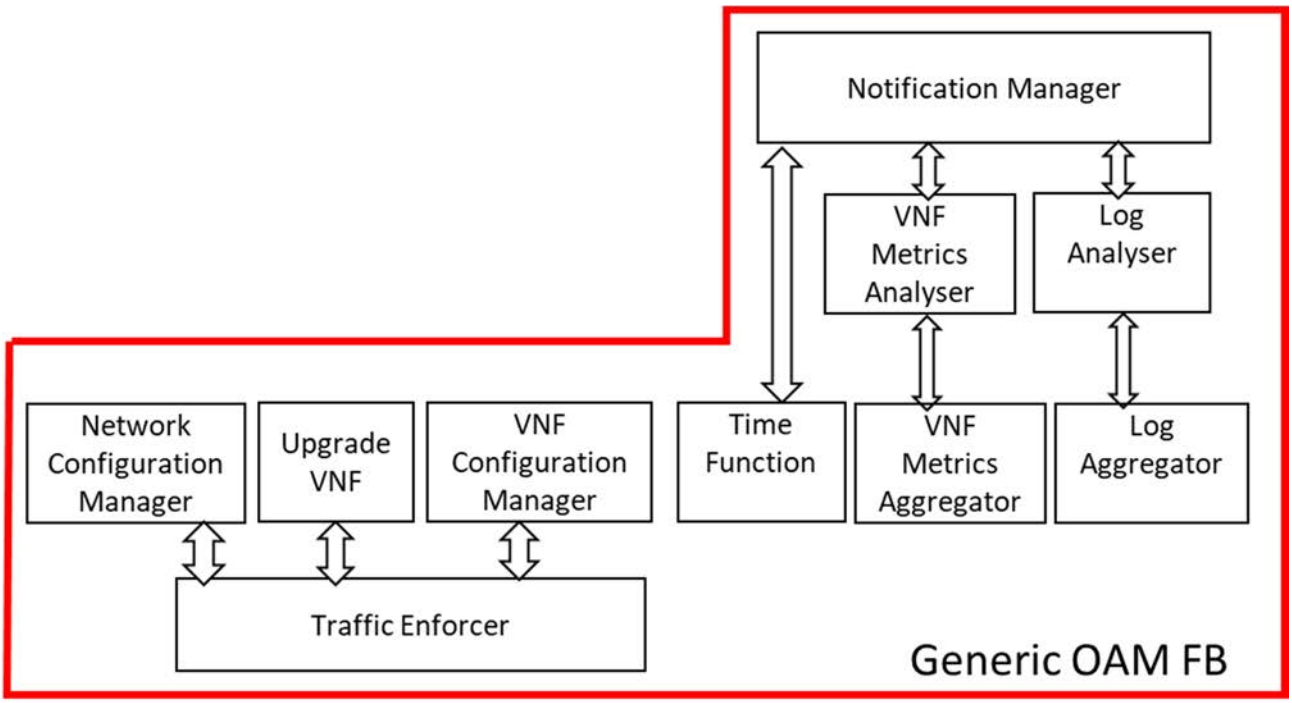

Figure 6.3.2-1: Internal interactions of functions in Generic OAM FB

6.3.3 Interaction of Generic OAM FB and other functions/functional blocks

This solution assumes that Operators send requests to Generic OAM FB via OSS/BSS and NFV-MANO. Generic OAM FB interacts with VNF/VNFC instance and functions like MDAF and CISM to complete operations and maintenance.

NOTE: In this solution, and in Figure 6.3.3-1, the CISM function is assumed to be part of NFV-MANO.

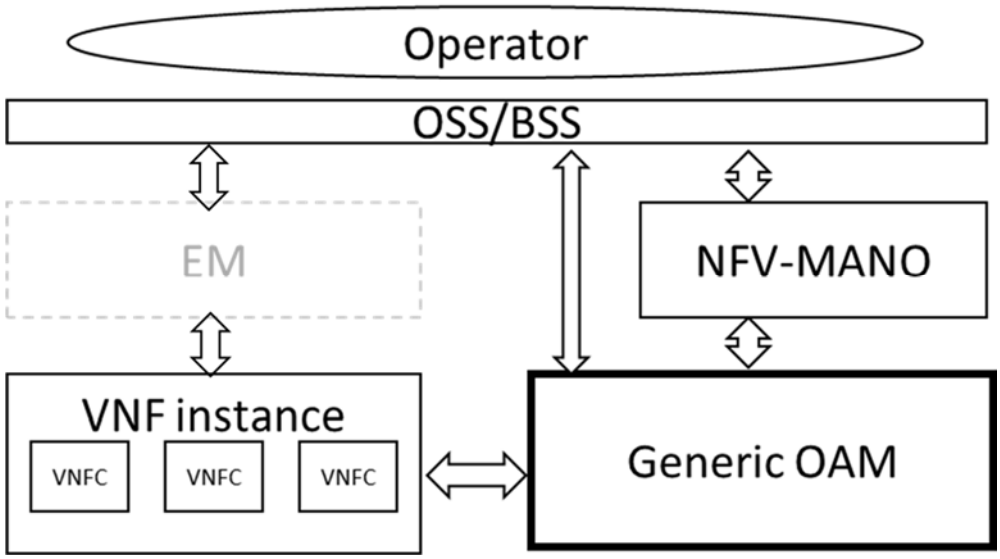

Figure 6.3.3-1: Interaction of Generic OAM FB and other functions/functional blocks

6.4 Solution B: Extending existing functional blocks for Generic OAM functionality

6.4.1 Introduction

VNF generic OAM functions have roles related to Performance management, Fault management, Configuration management, Software modification management, and Notification management. In NFV environment, VNF generic OAM functions can be spread to existing function as extensions of those functional blocks. There are two possible ways to split those functionalities. Clause 6.4.2 describes the first solution where both VNF-specific metrics aggregator/analyser are implemented in the Element Management (EM) for handling application or service specified metrics. Clause 6.4.3 describes the second solution where both the above VNF generic OAM functions are implemented in NFV-MANO for handling metrics consumed by the VNF instances.

6.4.2 Solution B1: Splitting of functionalities into existing functional blocks

Each functionality could be split as depicted in the Figure 6.4.2-1:

- EM: Network configuration manager function, Upgrade VNF function, VNF configuration manager function and VNF metrics aggregator/analyser functions can be defined as extended functionality of EM. The reason for introducing them to the EM is when considering the operation of these functionalities using existing interfaces, e.g. VNF lifecycle management and LCM coordination interface as defined in ETSI GS NFV-IFA 008 [i.5]. Also, it is assumed in this solution B1 that VNF-specific metric related functions are processing application or service specified metrics such as, number of processing calls or routing related metrics and therefore are sorted to EM side.

NOTE: In this solution, the realization of a "generic EM" is not implied. It only highlights what "VNF generic OAM functions" can be reused and be mapped to functionalities that are typically, in legacy environment, mapped to EM responsibilities.

NFV-MANO: Log aggregator/analyser functions, Notification manager function, Time function and Traffic enforcer function can be defined as extended functionality of NFV-MANO. The reason for splitting them to NFV-MANO is that Time function, Log aggregator function interact with NFVI layer and the Traffic enforcer function needs to be aware of the state of the NFVI resources which is not known to EM.

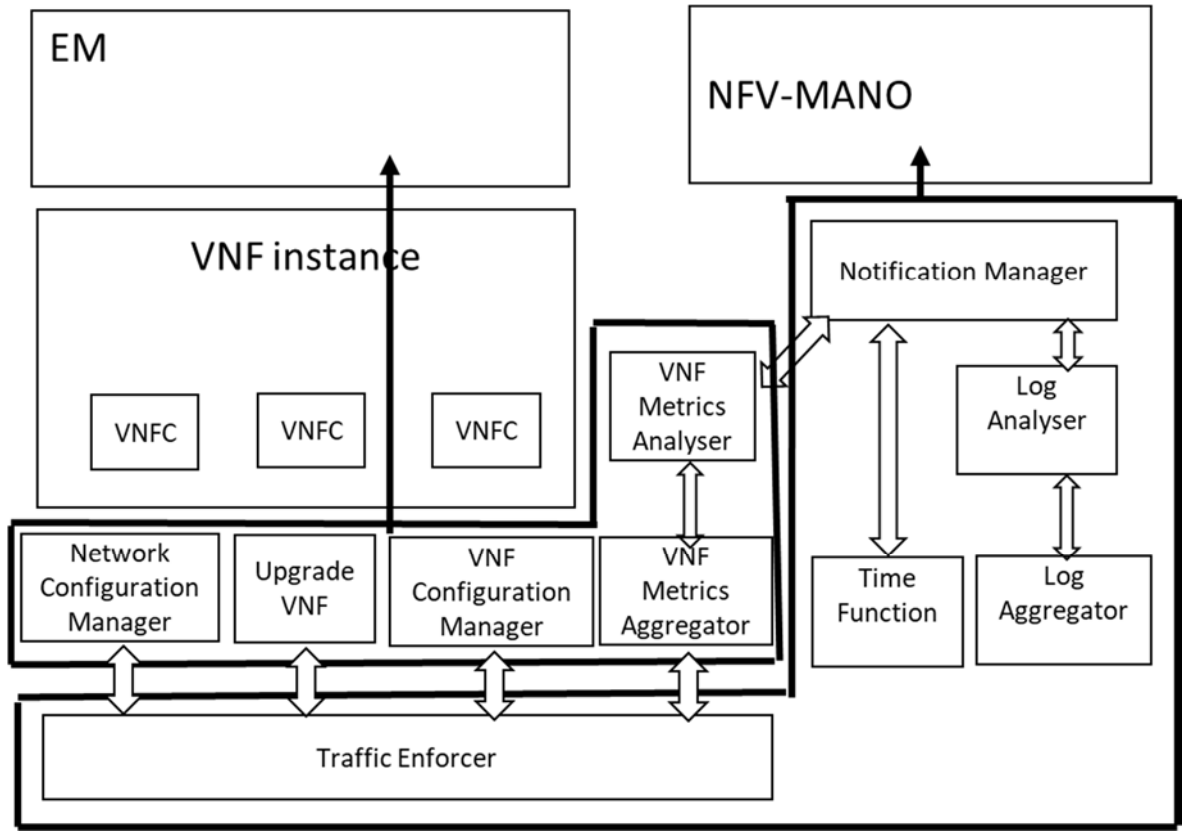

Figure 6.4.2-1: Splitting of functionalities into existing functional blocks

6.4.3 Solution B2: Splitting of functionalities into existing functional block

Each functionality could be split as depicted in the Figure 6.4.3-1. The only difference with Figure 6.4.2-1 is how to implement the VNF-specific metric related functions. It is assumed in this solution B2 that these functions are processing metrics consumed by VNF instances such as packets sent and therefore are sorted to NFV-MANO.

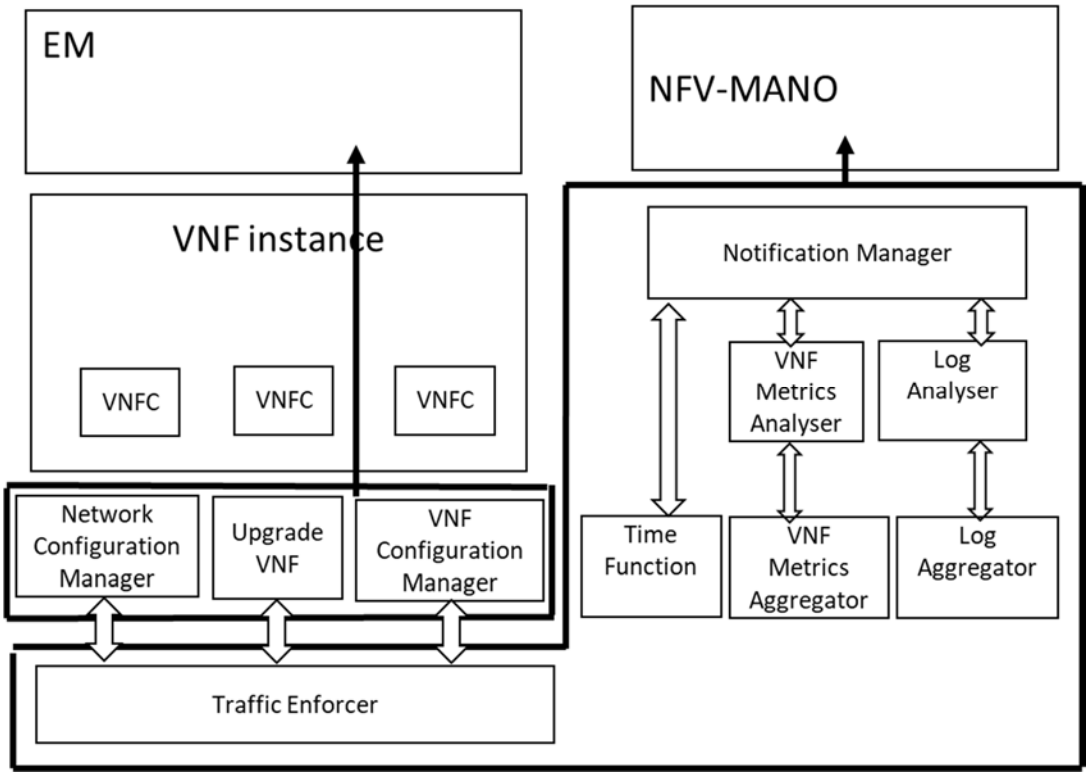

Figure 6.4.3-1: Splitting of functionalities into existing functional blocks

6.5 Solution C: Generic OAM functions as VNF

Generic OAM functions share use cases and characteristics with "VNF common services" as described in clause 5.1 (use cases) and clause 4.2 (characteristics) of ETSI GR NFV-IFA 029 [i.2]. The comparison is listed in Table 5.7-1, including the common design option to be modelled as VNF.

ETSI GR NFV-IFA 029 [i.2] further describes in its clause 7.1.1.2 "PaaS services as VNFs" the modelling and management of PaaS services which are hosted by VNFs. The described case when the PaaS services are visible at the NS design level can be applied to generic OAM functions. The same solutions as described in clause 7.1.1.2 of ETSI GR NFV-IFA 029 [i.2] for PaaS services hosted by VNFs apply for the solution of generic OAM functions as VNF.

6.6 Analysis

6.6.1 Introduction

The previous clauses of clause 6 provide options for potential solutions on how the generic OAM functions can be realized within the NFV system. The analysis of these options provides descriptions of their advantages and disadvantages in the following clauses.

The analysis considers the cases where all generic OAM functions are realized according to the same solution. However, it is worth noting that neither solution A, nor solution B cover the aspects to support the use case of "LCM of generic OAM functions" (clause 4.2). The reason is because no further solutions are analysed on how to perform LCM for generic OAM functions that go beyond or are an alternative of reusing LCM functionality provided by NFV-MANO. Besides, the mapping to the EMs as proposed for solutions B1 & B2 may not be feasible for all EM deployment options described in ETSI GS NFV-IFA 009 [i.8]. Hybrid cases where some generic OAM functions are realized according to one solution and other generic OAM functions are realized according to a different solution are therefore for further study.

6.6.2 Solution A: Introducing generic OAM as a new functional block

Advantages:

- No need to specify internal interactions among individual generic functions. Existing and best available solutions can be leveraged and their functionality can be mapped to the functional scope of this new functional block to ease potential implementations.
- Clear separation of concerns, the responsibility between this new functional block and existing NFV-MANO functional blocks and functions can be further specified during the normative work.

Disadvantages:

- The new functional block is expected to be also managed by the network operator.
- LCM of generic OAM function needs to be implemented separately.
- Additional interface requirements and information elements (e.g. specific to the generic OAM functions entities) are expected to be specified in ETSI GS NFV-IFA 031 [i.6].

6.6.3 Solution B: Extending existing functional blocks for Generic OAM functionality

Advantages:

- No new functional block is expected to be managed by the network operator.

Disadvantages:

- No separation of concerns between the generic OAM functions and existing NFV-MANO functional blocks.
- Additional interface requirements and information elements (e.g. specific to the generic OAM functions entities which are not realized in the same NFV-MANO functional block) are expected to be specified.
- For extending the functional scope of existing NFV-MANO functional blocks further specification in ETSI GS NFV-IFA 010 [i.7] is expected.

6.6.4 Solution C: Generic OAM functions as VNF

Advantages:

- The generic OAM functions can be managed by the network operator using existing mechanisms of NFV-MANO for the lifecycle management of these functions.
- Clear separation of concerns between the generic OAM functions and existing NFV-MANO functional blocks.
- No need to specify internal interactions among individual generic functions. Existing and best available solutions can be leveraged.

Disadvantages:

- Additional interface requirements and information elements (e.g. specific to the generic OAM functions entities) for the interaction between the VNF and NFV-MANO are expected to be specified.

7 Recommendations

7.1 Overview

Clause 7 is providing recommendations related to the VNF generic OAM functions. Clause 7.2 is collecting recommendations related to the suggested types of VNF generic OAM functions based on the use cases analysed. Clause 7.2 also provides a set of recommendations related to the expected characteristics and functionality of the VNF generic OAM functions.

7.2 Recommendations towards VNF generic OAM functions

Clause 5.3 identified and categorized different types of VNF generic OAM functions based on the set of use cases described in clause 4 of the present document. Table 7.2-1 provides recommendations related to these identified types of VNF generic OAM functions.

Table 7.2-1: Recommendations related to types of VNF generic OAM functions and their expected functionality

Identifier	Recommendation description
GenericOamType.001	It is recommended to specify functional requirements for a VNF generic OAM function of type "VNF metrics aggregator" to support to: • Collect different types of metrics from a set of VNF instances determined by a filter (see note 2). • Pre-process the metrics, e.g. harmonize the format of the metrics. • Aggregate the metrics in a configurable manner, e.g. aggregate all metrics related to performance, aggregate metrics from different instances belonging to the same VNF, aggregate metrics of VNF instances managed by the same VNFM, etc. • Store historical metric records, e.g. for abnormal behaviour detection or root cause analysis. • Expose the metrics (selected by a filter) to authorized consumers. See note 1.
GenericOamType.002	It is recommended to specify functional requirements for a VNF generic OAM function of type "VNF metrics analyser" to support to: • Analyse and process different types of metrics based on a set of analysis functions, e.g. abnormal behaviour detection, threshold crossing, statistical processing, etc. • Provide easy configuration of the analytics/processing to be applied, e.g. set thresholds, define the composition of the analytic function from a set of basic analytic functions. • Send notifications based on findings from the analysis of the metrics. • Expose the analytic results to authorized consumers. See note 1.
GenericOamType.003	It is recommended to specify functional requirements for a VNF generic OAM function of type "Log aggregator" to support to: • Collect different types of logs from a set of VNF instances determined by a filter (see note 2). • Pre-process the logs, e.g. to harmonize the format of the logs. • Aggregate the logs in a configurable manner, e.g. aggregate all logs with a certain log level, aggregate logs from different instances belonging to the same VNF, aggregate logs of VNF instances managed by the same VNFM, etc. • Store historical log records, e.g. for later root-cause analysis. • Expose logs (selected by a filter) to authorized consumers. See note 1.

Identifier	Recommendation description
GenericOamType.004	It is recommended to specify functional requirements for a VNF generic OAM function of type "Log analyser" to support to: • Analyse and process different types of logs based on a set of analysis functions, e.g. abnormal behaviour detection, threshold crossing, statistical processing, etc. • Provide easy configuration of the analytics/processing to be applied, e.g. set threshold, define the composition of the analytic function from a set of basic analytic functions. • Send notifications based on findings from the analysis of the logs. • Expose the analytic results to authorized consumers. See note 1.
GenericOamType.005	It is recommended to specify functional requirements for a VNF generic OAM function of type "Time function" to support to: • Ensure that the system time of all VNFs and their components is synchronized, i.e. the time skew is within a certain boundary. • Configure the time protocol(s) used in the system, e.g. configure the time source for all VNFs and their components. • Provide notifications and alerts, e.g. to support administrative actions and troubleshooting. • Set the correct time in the VNF components or host system. • Record and provide logs to other functions, e.g. log aggregator function. See note 1.
GenericOamType.006	It is recommended to specify functional requirements for a VNF generic OAM function of type "Traffic enforcer" to support to: • Perform appropriate actions based on the request received in order to achieve partial or full traffic isolation (both block or unblock traffic) and rerouting of traffic of one or more VNFC instances. See note 1.
GenericOamType.007	It is recommended to specify functional requirements for a VNF generic OAM function of type "VNF configuration manager" to support to: • Set configuration information (e.g. NFV-MANO-related configurations, certain application-related thresholds) to one or more VNF/VNFC instances. • Query configuration information from VNF/VNFC instances, e.g. current value of a parameter. See note 1.
GenericOamType.008	It is recommended to specify functional requirements for a VNF generic OAM function of type "Network configuration manager" to support to: • Set network configuration information related to one or more VNF/VNFC instances. See note 1.
GenericOamType.009	It is recommended to specify functional requirements for a VNF generic OAM function of type "Notification manager" to: • Handle (e.g. group, deduplicate, routes) notifications sent by other generic OAM functions and route them to authorized consumers. See note 1.
GenericOamType.010	It is recommended to specify functional requirements for a VNF generic OAM function of type "Upgrade VNF" to support to: • Provide new service by updating to a new software version and adding network connectivity to new type of VNF instance, i.e. support to update software of VNF/VNFC, import new service name, import new certificate for other VNF in load balancer, setting configuration of CP in load balancer, etc. • Add an additional virtual resource to a VNFC instance after VNF upgrading, i.e. support adding CPU or memory, or adding or extending volume to use by extending the storage size, etc. • Coordinate updating VNFs to run with new software, i.e. reference to software images (VM or OS container images), database schema change, application configuration files, etc. See notes 1 and 3.
GenericOamType.011	It is recommended to specify functional requirements for a VNF generic OAM function of type "Policy agent (PA)" to support to: • Perform automated decision making to ease administration tasks. • Notify to consumers (e.g. OSS/BSS) about events related to policy managements actions. • Parsing and enforcing VNF and VNF generic OAM functions policies. See note 4.

Identifier	Recommendation description
GenericOamType.012	It is recommended to specify functional requirements for a VNF generic OAM function of type "VNF testing manager" to support to: • Manage multilayer testing for VNFs. • Set and apply configuration related to testing (e.g. VNF-to-VNF connectivity testing). See note 5. • Execute the test upon request from OSS or due to VNF or NS LCM operation. • Collect testing results from different entities related to testing. See note 5.
NOTE 1: Due to the modular structure recommended for the VNF generic OAM functions (see Table 7.2-2), it is possible that an implementation of a VNF generic OAM function aggregates multiple basic types of VNF generic OAM functions, e.g. a "VNF metrics function" could support functionalities of a "VNF metrics aggregator" and "VNF metrics analyser". NOTE 2: The filter is recommended to support to filter the VNF/VNFC instances by type of the VNF/VNFC, vendor, host, zone, VNF instance identifier, etc. The filter is further recommended to be able to filter by metric/log type, severity level, etc. NOTE 3: Even though the function is named "Upgrade VNF", the function does not only support the upgrade, but also update procedures of a VNF, and in general on any kind of software modification involved. NOTE 4: To enable closed loop control, interactions are expected between the Policy agent and VNFs, NFV-MANO, OSS/BSS, NFVI management systems, other VNF generic OAM functions, MDA, etc. NOTE 5: Configuration of the test and results collection can be facilitated by the interaction of VNF testing manager with other VNF generic OAM functions like the Network configuration manager for the actual network configuration and the Notifications manager for the results collection.	

Table 7.2-2 provides recommendations related to characteristics of VNF generic OAM functions based on the analysis performed in clause 5.6.

Table 7.2-2: Recommendations related to characteristics of VNF generic OAM functions

Identifier	Recommendation description
GenericOamFunction.001	It is recommended to specify a functional requirement for a VNF generic OAM function to expose interfaces and operations used to call the functionality of the VNF generic OAM function, to configure the VNF generic OAM function, to query information and supported functionality and capabilities, to allow to subscribe to notifications provided by the VNF generic OAM function, and to expose metrics, logs and other information related to the VNF generic OAM function, including its lifecycle.
GenericOamFunction.002	It is recommended to specify a functional requirement for a VNF generic OAM function to support being consumed by any authorized VNF, authorized NFV-MANO functional entity, OSS/BSS, or other authorized generic OAM function.
GenericOamFunction.003	It is recommended to specify a functional requirement for a VNF generic OAM function to support being consumed/shared by one or multiple services/entities at a time.
GenericOamFunction.004	It is recommended to specify a functional requirement for a VNF generic OAM function to support requesting specific operations to NFV-MANO, VNF/VNFC instances, NFVI/hosts, and other VNF generic OAM functions as required to provide the expected functionality of the VNF generic OAM function.
GenericOamFunction.005	It is recommended to specify a functional requirement for a VNF generic OAM function to support consuming services by multiple entities/instances at a time.
GenericOamFunction.006	It is recommended to specify a functional requirement for a VNF generic OAM function to have a lifecycle independent of any consumer.
GenericOamFunction.007	It is recommended to specify a requirement for a VNF generic OAM function to be capable of being terminated in a graceful manner.
GenericOamFunction.008	It is recommended to specify a requirement for a VNF generic OAM function to be able to scale (see note).
NOTE: By being scalable, the VNF generic OAM function can compensate for the increasing/decreasing demand of its service.	

In Table 7.2-3, additional recommendations are provided for existing NFV-MANO functional blocks and functions.

Table 7.2-3: Other recommendations

Identifier	Recommendation description
vnf.generic.func.001	It is recommended to specify functional requirements for MDAF to be able to receive and handle pre-processed data analytics from the VNF metrics analyser and the Log analyser VNF generic OAM functions. See note.
NOTE: By means of interactions expected between MDAF and the Log analyser and the VNF metrics aggregator, MDAF can be a consumer of the northbound interface exposed by the VNF generic OAM functions as specified in ETSI GS NFV-IFA 049 [i.25].	

8 Conclusion

The present document presents use cases related to the functionality that would benefit from the use of VNF generic OAM functions. Based on the analysis of these use cases, recommendations have been proposed focusing on the suggested types and expected characteristics and functionality of the VNF generic OAM functions and focusing on the identification of potential requirements for NFV-MANO.

The present document also provides high level design of three potential solutions to provision VNF generic OAM functions. Solutions A and C have similar benefits in terms of being independent functional parts and are recommended to be considered for normative work. Solution C does not introduce differences at the functional level compared to Solution A and can be seen as a way to deploy VNF generic OAM functions. Solution C is the only solution documented in the present document that covers the aspects to support the use cases related to LCM of VNF generic OAM functions (clause 4.2). The primary benefit of Solution B is that no new functional block is expected to be managed by the network operator but is not recommended to be further considered during the normative work as it does not provide a good separation of concerns between VNF generic OAM functions and existing functional blocks and imply additional standardization efforts to specify or profile interfaces in between some of the VNF generic OAM functions.

Annex A:
Change History

Date	Version	Information about changes
October 2019	0.0.1	Initial draft. Implemented contributions NFVEVE(19)000085r1, NFVEVE(19)000086r1
December 2019	0.0.2	Implemented contributions NFVEVE(19)000088r3, NFVEVE(19)000097r3
March 2020	0.0.3	- Implemented contributions NFVEVE(20)000016r1, NFVEVE(20)000017r2, NFVEVE(20)000019r2, NFVEVE(20)000020r1, NFVEVE(20)000029r1 - Ensure consistent use of "Log aggregator" -> s/Logging/Log aggregator
March 2020	0.0.4	- Implemented contributions NFVEVE(20)000032r2, NFVEVE(20)000044r1, NFVEVE(20)000045r1, NFVEVE(20)000047, NFVEVE(20)000052
May 2020	0.0.5	- Implemented contributions NFVEVE(20)000051r1, 60r1 and 71 - Switched trigger and actors clauses as agreed during EVE#130 meeting
May 2020	0.0.6	- Implemented contributions NFVEVE(20)000085r3, 84, 73r1 and 72r1 - Fixed headings (level 4 added)
June 2020	0.0.7	Implemented contributions NFVEVE(20)000090, 94, 95r1, 96, 98 and 100
July 2020	0.0.8	Implemented contributions NFVEVE(20)000097r1, 105, 108r1, 109r1, 112r2, 116
October 2020	0.0.9	Implemented contributions NFVEVE(20)000107r1, 118r2, 120, 121, 137
November 2020	0.0.10	Implemented contributions NFVEVE(20)000144, 145r1, 146r1
December 2020	0.1.0	Implemented contributions NFVEVE(20)000156, 162r1, 164, 175r2
April 2021	0.1.1	Implemented contributions NFVEVE(20)184, NFVEVE(21)16r2, 17r2, 18, 21, 25r1, 26r1
August 2021	0.1.2	Implemented contributions NFVEVE(21)42r2, 43, 53r1, 54r3, 69r1
September 2021	0.2.0	Final draft for approval. Implemented contributions NFVEVE(21)83, 86
November 2022	5.0.1	Implemented contributions: NFVEVE(22)000114, 135r1,136r1, 137r1,138r2
December 2022	5.0.2	Implemented contributions: NFVEVE(22)000196r1, 197,198r1,202r2,Editorial actions: fixed section 4.4 numbering, added missing references
December 2022	5.0.3	Implemented contributions: NFVEVE(22)000215r2, 224r1, Editorial action: updated references (removed version numbers)
March 2023	5.0.4	Implemented contributions: NFVEVE(23)000004r1 , 005r2
April 2023	5.0.5	Implemented contributions: NFVEVE(23)000026r3, 027r2, 028r1, 0029, 030r2, 031r1, 032, 033r1, 034r3, 057r2. Editorial action: included missing references [i.23] and [i.24]
May 2023	5.0.6	Implemented contributions: NFVEVE(23)000084,085,086,093r1,095, 096
July 2023	5.0.7	Implemented contributions: NFVEVE(23)000111r1,112r1,114,117,124,129r2,130r1,138

History

Document history		
V5.1.1	October 2023	Publication