ETSI GS NFV-IFA 008 V2.3.1 (2017-08)

Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Ve-Vnfm reference point - Interface and Information Model Specification

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
RGS/NFV-IFA008ed231

Keywords
configuration, interface, management, MANO,
NFV, virtualisation

ETSI

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

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

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is 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 IPR Policy, no investigation, 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.

Foreword

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

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**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 specifies the interfaces supported over the Ve-Vnfm-em and Ve-Vnfm-vnf reference points of the NFV-MANO architectural framework ETSI GS NFV-MAN 001 [i.3] as well as the information elements exchanged over those interfaces.

2 References

2.1 Normative references

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

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

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

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

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

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

[3] ETSI GS NFV-IFA 010: "Network Functions Virtualisation (NFV); Management and Orchestration; Functional Requirements Specification".

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

[5] Recommendation ITU-T X.733: "Information technology - Open Systems Interconnection - Systems Management: Alarm reporting function".

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] ISO/IEC 9646-7: "Information technology - Open Systems Interconnection - Conformance testing methodology and framework - Part 7: Implementation Conformance Statements".

[i.2] ETSI GS NFV 003: "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

[i.3] ETSI GS NFV-MAN 001: "Network Functions Virtualisation (NFV); Management and Orchestration".

[i.4] ETSI GS NFV-IFA 009: "Network Functions Virtualisation (NFV); Management and Orchestration; Report on Architectural Options".

[i.5] ETSI GS NFV-IFA 013: "Network Functions Virtualisation (NFV); Management and Orchestration; Os-Ma-nfvo reference point - Interface and Information Model Specification".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in ETSI GS NFV 003 [i.2] apply.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GS NFV 003 [i.2] and the following apply:

NOTE: An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in ETSI GS NFV 003 [i.2].

CP	Connection Point
CPD	Connection Point Descriptor
DF	Deployment Flavour
FB	Functional Block
LCM	Lifecycle Management
VDU	VNF Deployment Unit
VL	Virtual Link
VLD	Virtual Link Descriptor

4 Overview of interfaces and information elements associated to the Ve-Vnfm-em and Ve-Vnfm-vnf reference points

4.1 Introduction

This clause provides an overview of interfaces and information models associated to the Ve-Vnfm-em and Ve-Vnfm-vnf reference points.

The Ve-Vnfm-em reference point is used for exchanges between EM and VNF Manager, and supports the following interfaces:

- VNF Lifecycle Management (produced by VNFM, consumed by EM).
- VNF Performance Management, resulting from virtualised resource performance information, (produced by VNFM, consumed by EM).
- VNF Fault Management, resulting from virtualised resource fault information, (produced by VNFM, consumed by EM).
- VNF Indicator (produced by EM, consumed by VNFM).

The Ve-Vnfm-vnf reference point is used for exchanges between VNF and VNF Manager, and supports the following interfaces:

- VNF Lifecycle Management (produced by VNFM, consumed by VNF).
- VNF Performance Management, resulting from virtualised resource performance information (produced by VNFM, consumed by VNF).

- VNF Fault Management, resulting from virtualised resource fault information (produced by VNFM, consumed by VNF).
- VNF Indicator (produced by VNF, consumed by VNFM).
- VNF Configuration (produced by VNF, consumed by VNFM).

The information elements exchanged by the interfaces above are also part of the present document.

4.2 Relation to other NFV Group Specifications

The present document is referencing information from the following NFV Group Specifications:

- Report on Architectural Options ETSI GS NFV-IFA 009 [i.4]:
  - This report describes architectural options that may influence the way some of the interfaces associated to the Ve-Vnfm-em and/or Ve-Ve-Vnfm-vnf reference points are used or might even suggest the need for extension.
- Functional Requirements Specification ETSI GS NFV-IFA 010 [3]:
  - Interfaces associated with the Ve-Vnfm reference point are based on the functional requirements specified in ETSI GS NFV-IFA 010 [3] for the VNFM functional block (FB).
- Vi-Vnfm reference point - Interface and Information Model Specification ETSI GS NFV-IFA 006 [1]:
  - The Virtualised Resources Performance Management and the Virtualised Resources Fault Management interfaces defined in ETSI GS NFV-IFA 006 [1] is also used on the Ve-Vnfm-em reference point.
- Or-Vnfm reference point - Interface and Information Model Specification ETSI GS NFV-IFA 007 [2]:
  - The VNF Lifecycle Management and VNF Lifecycle Change Notification interfaces defined in ETSI GS NFV-IFA 006 [1] are also used on the Ve-Vnfm-em and Ve-Vnfm-vnf reference points.
- VNF Packaging Specification ETSI GS NFV-IFA 011 [4]:
  - The specification of the Virtualised Network Function Descriptor (VNFD) in ETSI GS NFV-IFA 011 [4] defines information elements that are also relevant in the present document.
- Os-Ma-nfvo reference point - Interface and Information Model Specification ETSI GS NFV-IFA 013 [i.5]:
  - The VNF Package Management interface defined in ETSI GS NFV-IFA 013 [i.5] is also used on the Or-Vnfm reference point.

4.3 Conventions

The following notations, defined in ISO/IEC 9646-7 [i.1], are used for the qualifier column of interface information elements:

- M mandatory - the capability is required to be supported;
- O optional - the capability may be supported or not;
- CM conditional mandatory - the capability is required to be supported and is conditional on the support of some condition. This condition shall be specified in the Description column;
- CO conditional optional - the capability may be supported or not and is conditional on the support of some condition. This condition shall be specified in the Description column.

The following notation is used for parameters that represent identifiers, and for attributes that represent identifiers in information elements and notifications:

- If parameters are referring to an identifier of an actual object, their type is "Identifier".

- If an object (information element or notification) contains an attribute that identifies the object, the type of that attribute is "Identifier" and the description states that the attribute is the identifier of that particular notification or information element.

EXAMPLE 1: Identifier "resourceId" of the "NetworkSubnet information element" has type "Identifier" and description "Identifier of this NetworkSubnet information element".

- If an object (information element or notification) contains an attribute that references another object or objects defined in an ETSI NFV GS, the type of the attribute is "Identifier", followed by the list of objects it references.

EXAMPLE 2: "Identifier (Reference to Vnfc)" or "Identifier (Reference to Vnfc, VirtualLink or VirtualStorage)".

If the type of a parameter or attribute has been marked as "Not specified" in the "Content" column, this means that its specification is left for the protocol design/data model design stage.

5 Reference point and interface requirements

5.1 Introduction

The following clauses specify requirements applicable to interfaces in the specific context of the Ve-Vnfm-em and Ve-Vnfm-vnf reference points.

In case the VNF has embedded management functionalities, it may implement the consumer part of the Ve-Vnfm-em reference point interfaces, and interact with a VNFM on behalf of an EM.

5.2 Ve-Vnfm-em Reference Point Requirements

5.2.0 Requirements applicable to the Ve-Vnfm-em reference point

Table 5.2.0-1 specifies requirements applicable to the Ve-Vnfm-em reference point.

Table 5.2.0-1: Ve-Vnfm-em reference point requirements

Numbering	Functional requirements description
Ve-Vnfm-em.001	The Ve-Vnfm-em reference point shall support the VNF Lifecycle Management interface produced by the VNFM.
Ve-Vnfm-em.002	Void.
Ve-Vnfm-em.003	The Ve-Vnfm-em reference point shall support the VNF Performance Management interface produced by the VNFM.
Ve-Vnfm-em.004	The Ve-Vnfm-em reference point shall support the VNF Fault Management interface produced by the VNFM.
Ve-Vnfm-em.005	The Ve-Vnfm-em reference point may support the VNF Indicator interface produced by the EM (see notes 1 and 2).
Ve-Vnfm-em.006	Void.
NOTE 1: VNF Indicators are information supplied by the VNF or the EM to provide some indication on the VNF behaviour. VNFM can use these indicators in conjunction with virtualised resource data to perform auto-scaling decisions. NOTE 2: The support of VNF Indicator interface is VNF provider's decision. At least one indicator declared by the VNF provider in VNFD with source (VNF or EM) implies that the corresponding reference point (Ve-Vnfm-vnf or Ve-Vnfm-em) supports this interface.	

5.2.1 Interface Requirements

5.2.1.1 VNF Lifecycle Management interface requirements

Table 5.2.1.1-1 specifies the requirements applicable to the VNF Life cycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point.

Table 5.2.1.1-1: VNF Lifecycle Management interface requirements

Numbering	Functional requirements description
Ve-Vnfm-em.VnfLcm.001	The VNF Life cycle Management interface produced by the VNFM on the Ve-vnfm-em reference point shall support scaling a VNF instance.
Ve-Vnfm-em.VnfLcm.002	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support instantiating a VNF instance.
Ve-Vnfm-em.VnfLcm.003	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support terminating a VNF instance.
Ve-Vnfm-em.VnfLcm.004	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support querying information about a VNF instance.
Ve-Vnfm-em.VnfLcm.005	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support requesting VNF healing.
Ve-Vnfm-em.VnfLcm.006	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support requesting to change the state of a VNF instance/VNFC instance(s) (see note 1).
Ve-Vnfm-em.VnfLcm.007	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support querying the status of a VNF Lifecycle Management operation.
Ve-Vnfm-em.VnfLcm.008	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support modifying information about a VNF instance (see note 2).
Ve-Vnfm-em.VnfLcm.009	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support creating a VNF instance identifier and the associated instance of a VNF information element.
Ve-Vnfm-em.VnfLcm.010	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support deleting a VNF instance identifier and the associated instance of a VNF information element.
Ve-Vnfm-em.VnfLcm.011	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support providing to the EM notifications to the EM about changes of a VNF instance that are related to the VNF lifecycle management operation occurrences, further referred to as VNF lifecycle management operation occurrence notifications.
Ve-Vnfm-em.VnfLcm.012	VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall contain information about the type of VNF lifecycle management operation, the identification of the VNF instance, and the identification of the lifecycle management operation occurrence.
Ve-Vnfm-em.VnfLcm.013	VNF lifecycle management operation occurrence notifications provided on the VNF lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall contain information about the addition/deletion of VNF Components, and about the changes on virtualised resources associated to VNFC(s) as result of the VNF lifecycle management operation occurrence.
Ve-Vnfm-em.VnfLcm.014	VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall contain information about the virtual networks and connection point (CP)s that are added/deleted as part of the VNF lifecycle management operation occurrence (see note 3).
Ve-Vnfm-em.VnfLcm.015	VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support indicating the start of the lifecycle management operation occurrence, the end and the results of the lifecycle management operation occurrence, including any error produced from the lifecycle management operation occurrence.
Ve-Vnfm-em.VnfLcm.016	VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support indicating updates to the VNF instance information including configurable properties.

Numbering	Functional requirements description
Ve-Vnfm-em.VnfLcm.017	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support providing to the EM notifications about creation and deletion of a VNF identifier and the associated instance of a VNF information element, further referred to as VNF identifier creation/deletion notifications.
Ve-Vnfm-em.VnfLcm.018	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support managing subscriptions to VNF lifecycle management operation occurrence notifications and to VNF identifier creation/deletion notifications.
Ve-Vnfm-em.VnfLcm.019	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support providing to the VNFM configuration parameters for a VNF/VNFC instance (see note 4).
Ve-Vnfm-em.VnfLcm.020	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support requesting to change the external connectivity of a VLs to which a VNF instance is connected.
Ve-Vnfm-em.VnfLcm.021	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the capability to invoke VNF error handling operation(s) after the VNF life cycle operation occurrence fails (see notes 5 and 6).
NOTE 1: Change the state of a VNF instance/VNFC instance(s) refers to start or stop a VNF/VNFC instance. These operations are complementary to instantiate or terminate a VNF. NOTE 2: The requirement refers to the information that is writable. NOTE 3: This provides information about virtual networks and connections points that are internal to the VNF and whose creation was triggered by the VNFM. NOTE 4: Configuration parameters referred in this clause include those set at initial configuration and any other configurable parameter declared in the VNFD. NOTE 5: It is up to the protocol design stage to design the detail error handling operation(s). NOTE 6: It depends on the VNF capabilities whether and how the operation(s) are supported by a particular VNF.	

5.2.1.2 Void

5.2.1.3 VNF Fault Management interface requirements

Table 5.2.1.3-1 specifies requirements applicable to the VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point.

Table 5.2.1.3-1: VNF Fault Management interface requirements

Numbering	Functional requirements description
Ve-Vnfm-em.VnfFm.001	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support collecting VNF/VNFC fault information (see note).
Ve-Vnfm-em.VnfFm.002	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support alarm acknowledgement.
Ve-Vnfm-em.VnfFm.003	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support providing alarm notifications related to faults on VNF/VNFC instances.
Ve-Vnfm-em.VnfFm.004	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support providing notification when there is a change in alarm information on VNF/VNFC instances.
Ve-Vnfm-em.VnfFm.005	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the sending of notification to the EM when an alarm on a VNF/VNFC instance has been created.
Ve-Vnfm-em.VnfFm.006	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the sending of notification to the EM when an alarm on a VNF/VNFC instance has been cleared.
Ve-Vnfm-em.VnfFm.007	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the sending of notification(s) to the EM when the alarm list has been rebuilt.
Ve-Vnfm-em.VnfFm.008	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall allow unambiguous identification of the alarm on a VNF/VNFC instance sent to the EM.
Ve-Vnfm-em.VnfFm.009	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall allow unambiguous identification of the VNF/VNFC instance causing the alarm.

Numbering	Functional requirements description
Ve-Vnfm-em.VnfFm.010	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall allow unambiguous identification of the alarm cause.
Ve-Vnfm-em.VnfFm.011	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support providing to the EM notifications about alarms on a VNF instance and its VNFC as a consequence of state changes in the virtualised resources used by the VNF and its VNFC.
Ve-Vnfm-em.VnfFm.012	Notifications related to the alarms associated with the state changes of virtualised resources of a VNF instance provided on the VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall contain information necessary to identify the VNF and its VNFC(s), the origin (VIM and virtualised resource(s)) of the virtualised resource change notification(s), the type of alarm, and information about the cause of the alarm.
Ve-Vnfm-em.VnfFm.013	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall enable the VNFM to manage subscriptions to notifications related to alarms.
Ve-Vnfm-em.VnfFm.014	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support an operation to escalate when the perceived severity of an alarm needs to be changed, from the perspective of the EM.
NOTE: Fault information on a given VNF/VNFC instance can include the information related to the alarm (e.g. alarm created, alarm cleared, etc.), alarm causes and identification of this VNF/VNFC instance and fault information concerning the virtualised resources supporting the constituent VNF/VNFC instance.	

5.2.1.4 VNF Indicator interface requirements

Table 5.2.1.4-1 specifies the requirements applicable to the VNF Indicator interface produced by the EM on the Ve-Vnfm-em reference point.

Table 5.2.1.4-1: VNF Indicator interface requirements

Numbering	Functional requirements description
Ve-Vnfm-em.Ind.001	The VNF Indicator interface produced by the EM on the Ve-Vnfm-em reference point shall support providing notifications related to indicator value change, and to manage subscriptions related to such notifications.
Ve-Vnfm-em.Ind.002	The VNF Indicator interface produced by the EM on the Ve-Vnfm-em reference point shall support retrieving indicator values.

5.2.1.5 Void

5.2.1.6 VNF Performance Management interface requirements

Table 5.2.1.6-1 specifies requirements applicable to the VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point.

Table 5.2.1.6-1: VNF Performance Management interface requirements

Numbering	Functional requirements description
Ve-Vnfm-em.VnfPm.01	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the EM to control the collection and reporting of VNF/VNFC performance information, resulting from virtualised resources performance information, on the VNF(s) it manages (see note 1).
Ve-Vnfm-em.VnfPm.02	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the capability to notify the EM about the availability of VNF performance information.
Ve-Vnfm-em.VnfPm.03	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the EM to create a PM job specifying the VNF performance information that the EM requires from the VNFM.
Ve-Vnfm-em.VnfPm.04	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point hall support the EM to delete one or more PM job(s).
Ve-Vnfm-em.VnfPm.05	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall enable the EM to receive notifications of data availability for a PM job, and to manage subscriptions to such notifications.
Ve-Vnfm-em.VnfPm.06	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the EM to query the details of one or more PM job(s).
Ve-Vnfm-em.VnfPm.07	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the EM to manage the thresholds on specified VNF/VNFC performance information and VNF(s) (see note 2).
Ve-Vnfm-em.VnfPm.08	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall support the capability to notify the EM about a threshold defined for a specified metric of a VNF/VNFC being crossed.
Ve-Vnfm-em.VnfPm.09	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-em reference point shall enable the EM to receive notifications related to threshold crossing, and to manage subscriptions to such notifications.
NOTE 1: Performance information on a given VNF/VNFC results from collected performance information of the virtualised resources that are mapped to this VNF/VNFC instance. NOTE 2: Management of thresholds include creation, deletion and query the thresholds on specified VNF performance information and VNF(s).	

5.3 Ve-Vnfm-vnf Reference Point Requirements

5.3.0 Requirements applicable to the Ve-Vnfm-vnf reference point

Table 5.3.0-1 specifies requirements applicable to the Ve-Vnfm-vnf reference point.

Table 5.3.0-1: Ve-Vnfm-vnf reference point requirements

Numbering	Functional requirements description
Ve-Vnfm-vnf.001	The Ve-Vnfm-vnf reference point shall support the VNF Lifecycle Management interface produced by the VNFM.
Ve-Vnfm-vnf.002	The Ve-Vnfm-vnf reference point may support the VNF Configuration interface produced by the VNF (see note 3).
Ve-Vnfm-vnf.003	The Ve-Vnfm-vnf reference point shall support the VNF Performance Management interface produced by the VNFM.
Ve-Vnfm-vnf.004	The Ve-Vnfm-vnf reference point shall support the VNF Fault Management interface produced by the VNFM.
Ve-Vnfm-vnf.005	The Ve-Vnfm-vnf reference point may support the VNF Indicator interface produced by the VNF (see notes 1 and 2).
Ve-Vnfm-vnf.006	The Ve-Vnfm-vnf reference point shall support the means for the configuration of a VNF and for the VNFM to determine whether the VNF configuration has been completed (see note 4).
NOTE 1: VNF Indicators are information supplied by the VNF or the EM to provide some indication on the VNF behaviour. For example, VNFM can use these indicators in conjunction with virtualised resource data to perform auto-scaling decisions, or to determine whether the configuration changes have been completed, etc.	
NOTE 2: The support of VNF Indicator interface is VNF provider's decision. At least one indicator declared by the VNF provider in VNFD with source (VNF or EM) implies that the corresponding reference point (Ve-Vnfm-vnf or Ve-Vnfm-em) supports this interface.	
NOTE 3: The dependency on and support of VNF Configuration interface is declared by the VNF provider in the VNFD.	
NOTE 4: The configuration of a VNF can be performed by the VNFM via the VNF Configuration interface, or the VNF can acquire the configuration information from the VNFM. At least one of the mechanisms shall be supported for a particular VNF and be declared by the VNF provider in the VNFD. Examples of how the alternative mechanisms may be used for VNF configuration are given in Annex A.	

5.3.1 Interface Requirements

5.3.1.1 VNF Lifecycle Management interface requirements

Table 5.3.1.1-1 specifies the requirements applicable to the VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point.

Table 5.3.1.1-1: VNF Lifecycle Management interface requirements

Numbering	Functional requirements description
Ve-Vnfm-vnf.VnfLcm.001	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support scaling a VNF instance.
Ve-Vnfm-vnf.VnfLcm.002	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support requesting VNF healing.
Ve-Vnfm-vnf.VnfLcm.003	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support querying the status of a VNF Lifecycle Management operation (see note 1).
Ve-Vnfm-vnf.VnfLcm.004	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support querying information about a VNF instance.
Ve-Vnfm-vnf.VnfLcm.005	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support providing to the VNF notifications about changes of a VNF instance that are related to the VNF lifecycle management operation occurrences, further referred to as VNF lifecycle management operation occurrence notifications.
Ve-Vnfm-vnf.VnfLcm.006	VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall contain information about the type of VNF lifecycle management operation, the identification of the VNF instance, and the identification of the lifecycle management operation occurrence.
Ve-Vnfm-vnf.VnfLcm.007	VNF lifecycle management operation occurrence notifications provided on the VNF lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall contain information about the addition/deletion of VNF Components, and about the changes on virtualised resources associated to VNFC(s) as result of the VNF lifecycle management operation occurrence.
Ve-Vnfm-vnf.VnfLcm.008	VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall contain information about the virtual networks and connection point (CP)s that are added/deleted as part of the VNF lifecycle management operation occurrence (see note 2).
Ve-Vnfm-vnf.VnfLcm.009	VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support indicating the start of the lifecycle management operation occurrence, the end and the results of the lifecycle management operation occurrence, including any error produced from the lifecycle management operation occurrence.
Ve-Vnfm-vnf.VnfLcm.010	VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support indicating updates to the VNF instance information including configurable properties.
Ve-Vnfm-vnf.VnfLcm.011	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support managing subscriptions to VNF lifecycle management operation occurrence notifications.
Ve-Vnfm-vnf.VnfLcm.012	The VNF Lifecycle Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support the capability to invoke VNF error handling operation(s) after the VNF life cycle operation occurrence fails (see notes 3 and 4).
NOTE 1: The status of the operations can be queried only limited to the operations initiated by the VNF. NOTE 2: This provides information about virtual networks and connections points that are internal to the VNF and whose creation was triggered by the VNFM. NOTE 3: It is up to the protocol design stage to design the detailed error handling operation(s). NOTE 4: It depends on the VNF capabilities whether and how the operation(s) are supported by a particular VNF.	

5.3.1.2 VNF Configuration interface requirements

Table 5.3.1.2-1 specifies the requirements applicable to the VNF Configuration interface produced by the VNF on the Ve-Vnfm-vnf reference point.

Table 5.3.1.2-1: VNF Configuration interface requirements

Numbering	Functional requirements description
Ve-Vnfm-vnf.VnfConf.001	The VNF Configuration interface on the Ve-Vnfm-vnf reference point shall support setting of initial virtualisation-related configuration parameters for a VNF/VNFC instance.
Ve-Vnfm-vnf.VnfConf.002	The VNF Configuration interface on the Ve-Vnfm-vnf reference point shall support updating virtualisation-related configuration parameters for a VNF/VNFC instance.

5.3.1.3 VNF Indicator interface requirements

Table 5.3.1.3-1 specifies the requirements applicable to the VNF indicator interface produced by the VNF on the Ve-Vnfm-vnf reference point.

Table 5.3.1.3-1: VNF Indicator interface requirements

Numbering	Functional requirements description
Ve-Vnfm-vnf.VnfInd.001	The VNF Indicator interface provided by the VNF on the Ve-Vnfm-vnf reference point shall support providing notifications related to indicator value change, and to manage subscriptions related to such notifications.
Ve-Vnfm-vnf.VnfInd.002	The VNF Indicator interface provided by the VNF on the Ve-Vnfm-vnf reference point shall support retrieving indicator value.

5.3.1.4 VNF Performance Management interface requirements

Table 5.3.1.4-1 specifies requirements applicable to the VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point.

Table 5.3.1.4-1: VNF Performance Management interface requirements

Numbering	Functional requirements description
Ve-Vnfm-vnf.VnfVrPm.001	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support the VNFM to provide to VNF the performance information, resulting from virtualised resources performance information, on the VNF(s) it manages (see note).
Ve-Vnfm-vnf.VnfVrPm.002	The VNF Performance Management interface produced by the VNFM on the Ve-Vnfm- vnf reference point shall support the VNF to receive notifications related to virtualised resources performance information of the virtualised resources mapped to this VNF, and to manage subscriptions to such notifications.
NOTE: Performance information on a given VNF results from collected performance information of the virtualised resources that are mapped to this VNF instance.	

5.3.1.5 VNF Fault Management interface requirements

Table 5.3.1.5-1 specifies requirements applicable to the VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point.

Table 5.3.1.5-1: VNF Fault Management interface requirements

Numbering	Functional requirements description
Ve-Vnfm-vnf.VnfFm.001	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support collecting VNF/VNFC fault information related to virtualised resources used by the VNF/VNFC (see note).
Ve-Vnfm-vnf.VnfFm.002	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support alarm acknowledgement.
Ve-Vnfm-vnf.VnfFm.003	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support providing alarm notifications related to virtualised resources used by the VNF/VNFC instances.
Ve-Vnfm-vnf.VnfFm.004	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support providing notification when there is a change in alarm information related to virtualised resources used by VNF/VNFC instances.
Ve-Vnfm-vnf.VnfFm.005	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support the sending of notification to the VNF when an alarm related to virtualised resources used by a VNF/VNFC instance has been created.
Ve-Vnfm-vnf.VnfFm.006	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support the sending of notification to the VNF when an alarm related to virtualised resources used by a VNF/VNFC instance has been cleared.
Ve-Vnfm-vnf.VnfFm.007	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support the sending of notification(s) to the VNF when the alarm list has been rebuilt.

Numbering	Functional requirements description
Ve-Vnfm-vnf.VnfFm.008	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall allow unambiguous identification of the alarm related to virtualised resources used by a VNF/VNFC instance sent to the VNF.
Ve-Vnfm-vnf.VnfFm.009	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall allow unambiguous identification of the virtualised resources used by a VNF/VNFC instance causing the alarm.
Ve-Vnfm-vnf.VnfFm.010	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall allow unambiguous identification of the alarm cause.
Ve-Vnfm-vnf.VnfFm.011	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support providing to the VNF notifications about alarms on a virtualised resources mapped to VNF instance and its VNF Components as a consequence of state changes in the virtualised resources used by the VNF and its VNF Components.
Ve-Vnfm-vnf.VnfFm.012	Notifications related to the alarms associated with the state changes of virtualised resources of a VNF instance provided on the VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall contain information necessary to identify the VNF and the VNFC(s), the origin (VIM and virtualised resource(s)) of the virtualised resource change notification(s), the type of alarm, and information about the cause of the alarm.
Ve-Vnfm-vnf.VnfFm.013	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall enable the VNFM to manage subscriptions to notifications related to alarms.
Ve-Vnfm-vnf.VnfFm.014	The VNF Fault Management interface produced by the VNFM on the Ve-Vnfm-vnf reference point shall support an operation to escalate when the perceived severity of an alarm needs to be changed, from the perspective of the VNF.
NOTE: Fault information related to virtualised resources used by a given VNF/VNFC instance can include the information related to the alarm (e.g. alarm created, alarm cleared, etc.), alarm causes and identification of this VNF/VNFC instance and fault information concerning the virtualised resources supporting the constituent VNF/VNFC instance.	

6 VNF exposed interfaces

6.1 Introduction

This clause defines the interfaces exposed by the VNF towards the VNFM over the Ve-Vnfm reference point.

NOTE: The fact that information elements and attributes are presented in tabular form does not preclude protocol designs in which these information elements and attributes are encoded in different parts of request and response messages. For example, in a RESTful interface, parts of them may be encoded in the URL, in the message header, in the message body or any combination thereof.

6.2 VNF configuration interface

6.2.1 Description

This interface enables a VNFM to perform configuration operations on a VNF instance and its VNFC instance(s) or individual VNFC instances. Configuration parameters referred in this clause are those declared in the VNFD.

NOTE: The present document does not specify a Read/Query Configuration Information operation, but does not preclude it from being implemented at the protocol design stage (e.g. for protocol consistency).

The following operations shall be supported:

- SetConfiguration.

6.2.2 Void

6.2.3 Set Configuration

6.2.3.1 Description

This operation enables a VNFM to set the configuration parameters of a VNF instance and its VNFC instance(s) or individual VNFC instances. Table 6.2.3.1-1 lists the information flow exchanged between the VNF and the VNFM.

NOTE: The flow in table 6.2.3.1-1 does not preclude using utilities provided by VIM or NFVI (e.g. HOT or cloudinit) as an intermediate step between VNFM and VNF.

Table 6.2.3.1-1: ModifyConfiguration operation

Information Flow	Requirement	Direction
SetConfigurationRequest	Mandatory	VNFM → VNF
SetConfigurationResponse	Mandatory	VNF → VNFM

6.2.3.2 Input parameters

The input parameters sent when invoking the operation are provided in table 6.2.3.2-1.

Table 6.2.3.2-1: SetConfiguration operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	0..1	Identifier	Uniquely identifies the VNF instance. See note 3.
vnfConfigurationData	M	0..1	VnfConfiguration	Configuration data for the VNF instance. See note 1.
vnfcConfigurationData	M	0..N	VnfcConfiguration	Configuration data for a VNFC instance. See note 2.
NOTE 1: Cardinality of 0 is used when the operation is used for configuration of only individual VNFC instances. NOTE 2: Cardinality of 0 is used when the operation is used for configuration of only a VNF instance. NOTE 3: When present, this identifier is not part of the actual configuration data to be applied. The cardinality 0 is used when the unique instance identification is not possible (e.g. when VnfConfiguration is used for the initial configuration of the VNF, where VnfConfiguration may contain the vnfInstanceId to be set).				

6.2.3.3 Output parameters

The output parameters sent when responding to the operation is provided in table 6.2.3.3-1.

Table 6.2.3.3-1: SetConfiguration operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfConfigurationData	M	0..1	VnfConfiguration	Correspond to the vnfConfigurationData in the input information elements of the SetConfiguration operation if it has.
vnfcConfigurationData	M	0..N	VnfcConfiguration	Correspond to the vnfcConfigurationData in the input information elements of the SetConfiguration operation if it has.

6.2.3.4 Operation results

As a result of this operation, the producer (VNF) shall indicate to the consumer (VNFM) whether or not the operation was successful.

6.3 VNF Indicator interface

6.3.1 Description

This interface allows the VNF to provide information on value changes of VNF related indicators. VNF related indicators are declared in the VNFD.

The following operations are defined for this interface:

- Subscribe.
- Notify.
- Get Indicator Value.
- Terminate Subscription.
- Query Subscription Info.

6.3.2 Subscribe operation

6.3.2.1 Description

This operation enables the VNFM to subscribe with a filter for the notifications related to VNF indicator value changes sent by the VNF. Table 6.3.2.1-1 lists the information flow exchanged between the VNFM and the VNF.

NOTE: Specification of filtering mechanism is left for the protocol design stage.

Table 6.3.2.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	VNFM → VNF
SubscribeResponse	Mandatory	VNF → VNFM

6.3.2.2 Input parameters

Table 6.3.2.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting VNF related indicators.

6.3.2.3 Output parameters

Table 6.3.2.3-1: Subscribe operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription returned.

6.3.2.4 Operation results

As a result of this operation, the VNF shall indicate to the VNFM in the SubscribeResponse message whether the subscription was successful or not.

For a particular subscription, only notifications matching the filter will be delivered to the consumer.

6.3.3 Notify operation

6.3.3.1 Description

This operation distributes notifications to subscribers. It is a one-way operation issued by the VNF towards the VNFM that cannot be invoked as an operation by the consumer (VNFM).

In order to receive notifications, the VNFM shall have a subscription. Table 6.3.3.1-1 lists the information flow exchanged between the VNFM and the VNF.

Table 6.3.3.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	VNF → VNFM

The following notification can be notified/sent by this operation:

- IndicatorValueChangeNotification see clause 9.6.2.

6.3.4 Get Indicator Value operation

6.3.4.1 Description

This operation enables VNFM to request the actual value of a given indicator from the VNF. Table 6.3.4.1-1 lists the information flow exchanged between the VNFM and the VNF.

Table 6.3.4.1-1: GetIndicatorValue operation

Message	Requirement	Direction
GetIndicatorValueRequest	Mandatory	VNFM → VNF
GetIndicatorValueResponse	Mandatory	VNF → VNFM

6.3.4.2 Input parameters

Table 6.3.4.2-1: Get Indicator Value operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting VNF related indicators.

6.3.4.3 Output parameters

Table 6.3.4.3-1: Get Indicator Value operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
indicatorInformation	M	0..N	IndicatorInformation	The requested indicator values as a complex structure having the VNF Instance ID, Indicator and the value of the Indicator.

6.3.4.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular request, only indicators matching the filter will be delivered to the VNFM.

6.3.5 Terminate Subscription operation

6.3.5.1 Description

This operation enables the VNFM to terminate a particular subscription.

Table 6.3.5.1-1 lists the information flow exchanged between the VNFM and the VNF.

Table 6.3.5.1-1: Terminate Subscription operation

Message	Requirement	Direction
TerminateSubscriptionRequest	Mandatory	VNFM → VNF
TerminateSubscriptionResponse	Mandatory	VNF → VNFM

6.3.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.3.5.2-1.

Table 6.3.5.2-1: Terminate Subscription operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription to be terminated.

6.3.5.3 Output parameters

None.

6.3.5.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the VNFM will not receive notifications related to that subscription any longer. The result of the operation shall indicate if the subscription termination has been successful or not with a standard success/error result.

6.3.6 Query Subscription Info operation

6.3.6.1 Description

This operation enables the VNFM to query information about subscriptions.

Table 6.3.6.1-1 lists the information flow exchanged between the VNFM and the VNF.

Table 6.3.6.1-1: Query Subscription operation

Message	Requirement	Direction
QuerySubscriptionInfoRequest	Mandatory	VNFM → VNF
QuerySubscriptionInfoResponse	Mandatory	VNF → VNFM

6.3.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.3.6.2-1.

Table 6.3.6.2-1: Query Subscription Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filtering criteria to select one or a set of subscriptions. Details are left for the protocol design stage.

6.3.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.3.6.3-1.

Table 6.3.6.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query. Details are left for the protocol design stage.

6.3.6.4 Operation results

After successful operation, the VNF has queried the internal subscription objects. The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular query, information about the subscriptions to notifications related to VNF indicator value changes that the VNFM has access to and that are matching the filter shall be returned.

7 VNFM exposed interfaces

7.1 Introduction

This clause defines the interfaces exposed by the VNFM towards the VNF/EM over the Ve-Vnfm-vnf/Ve-Vnfm-em reference points.

NOTE: The fact that information elements and attributes are presented in tabular form does not preclude protocol designs in which these information elements and attributes are encoded in different parts of request and response messages. For example, in a RESTful interface, parts of them can be encoded in the URL, in the message header, in the message body or any combination thereof.

7.2 VNF Lifecycle Management interface

7.2.1 Description

This interface allows the VNF/EM to invoke VNF lifecycle management operations towards the VNFM.

The following operations are defined for VNF to invoke:

- Scale VNF.
- Heal VNF.
- Get Operation Status.

The following operations are defined for EM to invoke:

- Create VNF Identifier.
- Instantiate VNF.
- Scale VNF.
- Scale VNF to Level.
- Change VNF Flavour.
- Terminate VNF.
- Delete VNF Identifier.

- Query VNF.
- Heal VNF.
- Operate VNF.
- Modify VNF Information, including VNF configurable properties.
- Get Operation Status.
- Change External VNF connectivity.

An identifier (i.e. lifecycleOperationOccurrenceId) is generated for each VNF management lifecycle operation occurrence, except for Query VNF, Create VNF Identifier, Delete VNF Identifier and Get Operation Status.

Furthermore, this interface allows the EM to manage subscriptions to notifications sent by the VNFM which inform about changes of a VNF instance that are related to VNF lifecycle, management operation occurrences, related to updates of VNF information attributes as well as related to the creation/deletion of a VNF instance identifier and the associated instance of a VnfInfo information element. It further allows the VNFM to provide such notifications to the subscriber.

7.2.2 Create VNF Identifier operation

7.2.2.1 Description

This operation creates a VNF instance identifier, and an associated instance of a VnfInfo information element, identified by that identifier, in the NOT_INSTANTIATED instantiation state without instantiating the VNF or doing any additional lifecycle operation(s). It allows returning right away a VNF instance identifier that can be used in subsequent lifecycle operations, like the Instantiate VNF operation.

This operation shall be supported for all VNFs.

Table 7.2.2.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.2.2.1-1: Create VNF Identifier operation

Message	Requirement	Direction
CreateVnfIdentifierRequest	Mandatory	EM → VNFM
CreateVnfIdentifierResponse	Mandatory	VNFM → EM

7.2.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.2.2-1.

Table 7.2.2.2-1: Create Vnf Identifier operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfdId	M	1	Identifier	Identifier that identifies the VNFD which defines the VNF instance to be created. See note.
vnfInstanceName	M	0..1	String	Human-readable name of the VNF instance to be created.
vnfInstanceDescription	M	0..1	String	Human-readable description of the VNF instance to be created.
NOTE: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [4], clause 7.1.2.2.				

7.2.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.2.3-1.

Table 7.2.2.3-1: Create VNF Identifier operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	VNF instance identifier just created.

7.2.2.4 Operation results

In case of success, an instance of a VnfInfo information element, in the NOT_INSTANTIATED instantiation state has been created and can be used in subsequent lifecycle operations and the corresponding VnfIdentifierCreationNotification has been sent. In case of failure, appropriate error information is returned.

7.2.3 Instantiate VNF operation

7.2.3.1 Description

This operation instantiates a particular Deployment Flavour (DF) of a VNF that has been in the NOT_INSTANTIATED instantiation state, based on the definition in the VNFD.

This operation shall be supported for all VNFs.

Table 7.2.3.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.2.3.1-1: Instantiate VNF operation

Message	Requirement	Direction
InstantiateVnfRequest	Mandatory	EM → VNFM
InstantiateVnfResponse	Mandatory	VNFM → EM

7.2.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.3.2-1.

Table 7.2.3.2-1: Instantiate VNF operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.
flavourId	M	1	Identifier	Identifier of the VNF DF to be instantiated.
instantiationLevelId	M	0..1	Identifier	Identifier of the instantiation level of the DF to be instantiated. If not present, the default instantiation level as declared in the VNFD shall be instantiated.
extVirtualLink	M	0..N	ExtVirtualLinkData	Information about external virtual link (VL)s to connect the VNF to.
extManagedVirtualLink	M	0..N	ExtManagedVirtualLinkData	Information about internal VLs that are managed by other entities than the VNFM (see note).

Parameter	Qualifier	Cardinality	Content	Description
localizationLanguage	M	0..1	Not specified	Localization language of the VNF to be instantiated. The localization languages supported by a VNF can be declared in the VNFD. If this parameter is not provided and the "defaultLocalizationLanguage" attribute is declared in the VNFD, the "defaultLocalizationLanguage" shall be used to determine the localization language VNF to be instantiated.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the EM as input to the instantiation process, specific to the VNF being instantiated, as declared in the VNFD (see clause 7.1.5.3 in ETSI GS NFV-IFA 011 [4]).
NOTE: The indication of externally-managed internal VLs is needed in case networks have been pre-configured for use with certain VNFs, for instance to ensure that these networks have certain properties such as security or acceleration features, or to address particular network topologies.				

7.2.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.3.3-1.

Table 7.2.3.3-1: Instantiate Vnf operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the VNF lifecycle operation occurrence.

7.2.3.4 Operation results

In case of success, the VNF has been instantiated and initially configured, and the associated instance of a VnfInfo information element has been updated. The VNF instance is in the INSTANTIATED instantiation state.

In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

NOTE: In the present document, the operation result of the VNF instantiation request by EM cannot be determined to have the same result as VNF instantiation request as part of the NS LCM update operation performed through the NFVO (see clauses 7.3.5 and 8.3.4.10 in ETSI GS NFV-IFA 013 [i.5]), in particular in what concerns re-using the newly VNF instance at the NFVO level and associating it to a corresponding NS.

The VNFM shall first return the lifecycleOperationOccurrenceId and second send the "start" LCM Operation Occurrence Notification before additional notifications or messages as part of this operation are issued, or operations towards the NFVO or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the VNFM shall send the "result" LCM Operation Occurrence Notification.

7.2.4 Scale VNF operation

7.2.4.1 Description

This operation enables a VNF instance or EM to request a VNFM to perform a scaling procedure.

This operation provides methods to request scaling a VNF.

This interface provides methods for request scaling a VNF in multiple ways:

- horizontal scaling:
  - *scale out*: adding additional VNFC instances to the VNF to increase capacity;
  - *scale in:* removing VNFC instances from the VNF, in order to release unused capacity;
- vertical scaling (not supported in the present document):
  - *scale up:* adding further resources to existing VNFC instances, e.g. increase memory, CPU capacity or storage size of the virtualisation container hosting a VNFC instance, in order to increase VNF capacity;
  - *scale down:* removing resources from existing VNFC instances, e.g. decrease memory, CPU capacity or storage size of the virtualisation container hosting a VNFC instance, in order to release unused capacity.

Potentially, different aspects of a VNF can be scaled independently. For example, a VNF could be designed to provide static capacity such as database nodes and dynamic capacity such as query processing nodes. Such a VNF might be scaled w.r.t. two separate aspects: the 'static capacity' aspect can be scaled by adding VNFCs from VNF Deployment Units (VDU)s defining database nodes, and the 'dynamic capacity' aspect can be scaled by adding VNFCs from VDUs defining query processing nodes.

In complex VNF designs, scaling a VNF often requires adding/removing a number of related VNFC instances of several different types, possibly based on multiple VDUs. For example, in a high availability configuration, it might be required to add in each scaling step a pair of VNFC instances, one in active and one in standby configuration.

The ScaleVnfRequest in the interface allows the consumer to specify the scaling aspect. The scaling aspects valid for a particular VNF are defined in the VNFD. After receiving a scale request, the VNFM will figure out the necessary set of VNFCs and the related set of resources based on VNF-specific rules, for instance using the lifecycle management script associated to the Scale VNF event.

When scaling a VNF for a particular aspect, the number of scaling steps to apply to that aspect can be provided as a parameter. A scaling step is the smallest unit by which a particular aspect of a VNF can be scaled, and is mapped by the VNFM to the addition (or removal) of a certain number of resources, based on one or more VDUs. For each scaling aspect, the maximum scale level is defined in the VNFD. The minimum scale level is assumed as zero; the maximum scale level corresponds to the maximum number of steps that can be performed within this aspect, starting at the minimum scale level (i.e. zero). At each point in time between the completed VNF instantiation and the VNF termination, the "size" of the VNF w.r.t. a particular aspect can be expressed by the current scale level w.r.t. that aspect, and can be obtained, among other information, by invoking the "QueryVNF" operation. When the VNF is instantiated, the current scale level is initialized with values that are defined as part of the instantiation level in the VNFD for the associated aspect. Figure 7.2.4.1-1 illustrates the concepts described above.

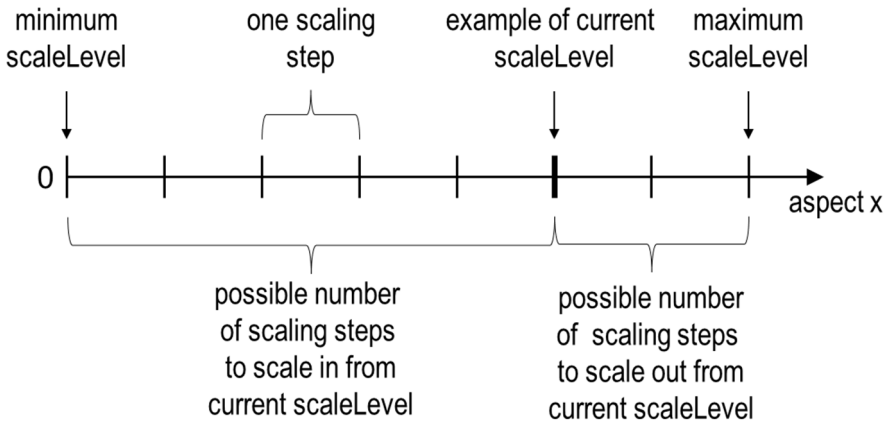

Figure 7.2.4.1-1: Illustrating the concepts of scaleLevel and scaling steps for a particular scaling aspect

The VNFM will then communicate information about the necessary resource changes via the GrantVnfLifecycleOperationRequest to the NFVO.

It depends on the VNF capabilities, and is declared in the VNFD, whether and how this operation is supported for a particular VNF.

Table 7.2.4.1-1 lists the information flows exchanged between the VNF/EM and the VNFM.

Table 7.2.4.1-1: Scale VNF operation

Message	Requirement	Direction
ScaleVnfRequest	Mandatory	VNF → VNFM (see note) EM → VNFM
ScaleVnfResponse	Mandatory	VNFM → VNF (see note) VNFM → EM
NOTE: In case of VNF without EM, the scaling request is invoked by management function within the VNF. The management function may implement the consumer part of the VNF LCM interface on the Ve-Vnfm-em reference point.		

7.2.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.4.2-1.

Table 7.2.4.2-1: Scale Vnf operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance to which this scaling request is related.
type	M	1	Enum	Defines the type of the scale operation requested (scale out, scale in). The set of types actually supported depends on the capabilities of the VNF being managed as declared in the VNFD. See note 1.
aspectId	M	1	Identifier	Identifies the aspect of the VNF that is requested to be scaled, as declared in the VNFD.
numberOfSteps	M	0..1	Integer	Number of scaling steps to be executed as part of this ScaleVnf operation. It shall be a positive number. Defaults to 1. The VNF Provider defines in the VNFD whether or not a particular VNF supports performing more than one step at a time. Such a property in the VNFD applies for all instances of a particular VNF. See note 2.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the VNF/EM as input to the scaling process, specific to the VNF being scaled, as declared in the VNFD (see clause 7.1.5.4 in ETSI GS NFV-IFA 011 [4]).
NOTE 1: ETSI GS NFV-IFA 010 [3] specifies that the lifecycle management operations that expand or contract a VNF instance include scale in, scale out, scale up and scale down. Vertical scaling (scale up, scale down) is not supported in the present document. NOTE 2: A scaling step is the smallest unit by which a VNF can be scaled w.r.t. a particular scaling aspect.				

7.2.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.4.3-1.

Table 7.2.4.3-1: Scale VNF operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the VNF lifecycle operation occurrence.

7.2.4.4 Operation results

In case of success, the VNF has been scaled according to the request, and the associated instance of a VnfInfo information element has been updated.

In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

The VNFM shall first return the lifecycleOperationOccurrenceId and second send the "start" LCM Operation Occurrence Notification before additional notifications or messages as part of this operation are issued, or operations towards the NFVO or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the VNFM shall send the "result" LCM Operation Occurrence Notification.

7.2.5 Scale VNF to Level operation

7.2.5.1 Description

This operation scales an instantiated VNF of a particular DF to a target size. The target size is either expressed as an instantiation level of that DF as defined in the VNFD, or given as a list of scale levels, one per scaling aspect of that DF. Instantiation levels and scaling aspects are declared in the VNFD. Typically, the result of this operation is adding and/or removing Network Functions Virtualisation Infrastructure (NFVI) resources to/from the VNF.

It depends on the VNF capabilities, and is declared in the VNFD, whether this operation is supported for a particular VNF.

Table 7.2.5.1-1 lists the information flow exchanged between the VNF/EM and the VNFM.

Table 7.2.5.1-1: Scale VNF To Level operation

Message	Requirement	Direction
ScaleVnfToLevelRequest	Mandatory	VNF → VNFM EM → VNFM
ScaleVnfToLevelResponse	Mandatory	VNFM → VNF VNFM → EM

7.2.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.5.2-1.

Table 7.2.5.2-1: Scale VNF To Level operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance to which this scaling request is related.
instantiationLevelId	M	0..1	Identifier	Identifier of the target instantiation level of the current DF to which the VNF is requested to be scaled. Either instantiationLevelId or scaleInfo but not both shall be present.
scaleInfo	M	0..N	ScaleInfo	For each scaling aspect of the current DF, defines the target scale level to which the VNF is to be scaled. The VNF Provider defines in the VNFD whether or not a particular VNF supports scaling according to this parameter. Such a property in the VNFD applies for all instances of a particular VNF. Either instantiationLevelId or scaleInfo but not both shall be present.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the EM as input to the scaling process, specific to the VNF being scaled, as declared in the VNFD (see clause 7.1.5.4 in ETSI GS NFV-IFA 011 [4]).

7.2.5.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.5.3-1.

Table 7.2.5.3-1: Scale VNF To Level operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the VNF lifecycle operation occurrence.

7.2.5.4 Operation results

In case of success, the VNF has been scaled according to the request, and the associated instance of a VnfInfo information element has been updated.

In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

The VNFM shall first return the lifecycleOperationOccurrenceId and second send the "start" LCM Operation Occurrence Notification before additional notifications or messages as part of this operation are issued, or operations towards the NFVO or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the VNFM shall send the "result" LCM Operation Occurrence Notification.

7.2.6 Change VNF Flavour operation

7.2.6.1 Description

This operation changes the DF of a VNF instance.

It depends on the VNF capabilities, and is declared in the VNFD, whether this operation is supported for a particular VNF. This operation may be service-disruptive.

Table 7.2.6.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.2.6.1-1: Change VNF Flavour operation

Message	Requirement	Direction
ChangeVnfFlavourRequest	Mandatory	EM → VNFM
ChangeVnfFlavourResponse	Mandatory	VNFM → EM

7.2.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.6.2-1.

Table 7.2.6.2-1: Change VNF Flavour operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance to be modified.
newFlavourId	M	1	Identifier	Identifier of the new VNF DF to apply to this VNF instance.
instantiationLevelId	M	0..1	Identifier	Identifier of the instantiation level of the DF to be used. If not present, the default instantiation level as declared in the VNFD shall be used.
extVirtualLink	M	0..N	ExtVirtualLinkData	Information about external VLs to connect the VNF to.
extManagedVirtualLink	M	0..N	ExtManagedVirtualLink Data	Information about internal VLs that are managed by other entities than the VNFM (see note).
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the EM as input to the flavour change process, specific to the VNF being modified, as declared in the VNFD (see clause 7.1.5.4 in ETSI GS NFV-IFA 011 [4]).
NOTE: The indication of externally-managed internal VLs is needed in case networks have been pre-configured for use with certain VNFs, for instance to ensure that these networks have certain properties such as security or acceleration features, or to address particular network topologies.				

7.2.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.6.3-1.

Table 7.2.6.3-1: Change VNF Flavour operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the VNF lifecycle operation occurrence.

7.2.6.4 Operation results

In case of success, the VNF has been modified to use the new DF and initially configured, and the associated instance of a VnfInfo information element has been updated.

In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

The VNFM shall first return the lifecycleOperationOccurrenceId and second send the "start" LCM Operation Occurrence Notification before additional notifications or messages as part of this operation are issued, or operations towards the NFVO or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the VNFM shall send the "result" LCM Operation Occurrence Notification.

7.2.7 Terminate VNF operation

7.2.7.1 Description

This interface enables an EM to request a VNFM to perform termination procedures on a VNF instance that has been in the INSTANTIATED instantiation state.

Terminating a VNF instance does not delete the instance of the VnfInfo information element.

This operation shall be supported for all VNFs.

Table 7.2.7.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.2.7.1-1: Terminate VNF operation

Message	Requirement	Direction
TerminateVnfRequest	Mandatory	EM → VNFM
TerminateVnfResponse	Mandatory	VNFM → EM

7.2.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.7.2-1.

Table 7.2.7.2-1: Terminate VNF operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance to be terminated.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the EM as input to the Terminate VNF operation, specific to the VNF being terminated, as declared in the VNFD (see clause 7.1.5.7 in ETSI GS NFV-IFA 011 [4]).

7.2.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.7.3-1.

Table 7.2.7.3-1: Terminate VNF operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the VNF lifecycle operation occurrence.

7.2.7.4 Operation results

In case of success, the VNF instance has been terminated and resources used by the VNF have been released. The associated instance of a VnfInfo information element has been updated. The VNF instance is in the NOT_INSTANTIATED instantiation state.

In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

The VNFM shall first return the lifecycleOperationOccurrenceId and second send the "start" LCM Operation Occurrence Notification before additional notifications or messages as part of this operation are issued, or operations towards the NFVO or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the VNFM shall send the "result" LCM Operation Occurrence Notification.

7.2.8 Delete VNF Identifier operation

7.2.8.1 Description

This operation deletes a VNF instance identifier and the associated instance of a VnfInfo information element in the NOT_INSTANTIATED instantiation state.

This operation shall be supported for all VNFs.

Table 7.2.8.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.2.8.1-1: Delete VNF Identifier operation

Message	Requirement	Direction
DeleteVnfIdentifierRequest	Mandatory	EM → VNFM
DeleteVnfIdentifierResponse	Mandatory	VNFM → EM

7.2.8.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.8.2-1.

Table 7.2.8.2-1: Delete VNF Identifier operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	VNF instance identifier to be deleted.

7.2.8.3 Output parameters

No output parameter.

7.2.8.4 Operation results

In case of success, the VNF instance identifier and the associated instance of the VnfInfo information element has been deleted and can no longer be used and the corresponding VnfIdentifierDeletionNotification has been sent. If the VNF instance was not terminated (i.e. the VNF is in INSTANTIATED instantiation state), the operation shall be rejected.

In case of failure, appropriate error information is returned.

7.2.9 Query VNF operation

7.2.9.1 Description

This operation provides information about VNF instances. The applicable VNF instances can be chosen based on filtering criteria, and the information can be restricted to selected attributes.

This operation shall be supported for all VNFs.

Table 7.2.9.1-1 lists the information flow exchanged between the producer (VNFM) and the consumer (EM or VNF).

Table 7.2.9.1-1: Query VNF operation

Message	Requirement	Direction
QueryVnfRequest	Mandatory	EM → VNFM VNF → VNFM
QueryVnfResponse	Mandatory	VNFM → EM VNFM → VNF

7.2.9.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.9.2-1.

Table 7.2.9.2-1: Query VNF operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filter to select the VNF instance(s).
attributeSelector	M	0..N	String	Provides a list of attribute names. If present, only these attributes are returned for the VNF instance(s) matching the filter. If absent, the complete information is returned for the VNF instance(s) matching the filter.

7.2.9.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.9.3-1.

Table 7.2.9.3-1: Query VNF operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInfo	M	0..N	VnfInfo	The information items about the selected VNF instance(s) that are returned. If attributeSelector is present, only the attributes listed in attributeSelector are returned for the selected VNF instance(s).
NOTE: The lower cardinality is 0 since there may be no matches to the provided filter.				

7.2.9.4 Operation results

In case of success, information related to the VNF instances that match the filter is returned. In case of failure, appropriate error information is returned.

7.2.10 Heal VNF operation

7.2.10.1 Description

This operation enables either a VNF instance or an EM to request a VNFM to perform a VNF healing procedure.

It depends on the VNF capabilities, and is declared in the VNFD, whether this operation is supported for a particular VNF.

Table 7.2.10.1-1 lists the information flow exchanged between the VNF/EM and the VNFM.

Table 7.2.10.1-1: Heal VNF operation

Message	Requirement	Direction
HealVnfRequest	Mandatory	VNF → VNFM (see note) EM → VNFM
HealVnfResponse	Mandatory	VNFM → VNF (see note) VNFM → EM
NOTE: In case of VNF without EM, the healing request is invoked by management function within the VNF. The management function may implement the consumer part of the VNF Lifecycle Management (LCM) interface on the Ve-Vnfm-em reference point.		

7.2.10.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.10.2-1.

Table 7.2.10.2-1: Heal VNF operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifies the VNF instance requiring a healing action.
vnfcInstanceId	M	0..N	Identifier	List of VNFC instances requiring a healing action.
cause	M	0..1	String	Indicates the reason why a healing procedure is required.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the VNF/EM as input to the healing process, specific to the VNF being healed, as declared in the VNFD (see clause 7.1.5.6 in ETSI GS NFV-IFA 011 [4]). EXAMPLE: Input parameters to VNF-specific healing procedures.
healScript	M	0..1	String	Provides link to a script that should be executed as part of the healing action or a set of rules for healing procedure.

7.2.10.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.10.3-1.

Table 7.2.10.3-1: Heal VNF operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the VNF lifecycle operation occurrence.

7.2.10.4 Operation results

In case of success, the VNF/VNFC has been healed, and the associated instance of a VnfInfo information element has been updated.

In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

The VNFM shall first return the lifecycleOperationOccurrenceId and second send the "start" LCM Operation Occurrence Notification before additional notifications or messages as part of this operation are issued, or operations towards the NFVO or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the VNFM shall send the "result" LCM Operation Occurrence Notification.

7.2.11 Operate VNF operation

7.2.11.1 Description

This operation enables requesting to change the state of a VNF instance or VNFC instance(s), including starting and stopping the VNF/VNFC instance.

NOTE 1: These operations are complementary to instantiating and terminating a VNF.

NOTE 2: In the present document, only starting and stopping the VNF/VNFC instance(s) are supported. Extension of this operation to support other VNF/VNFC state changes is left for future specification.

A VNF/VNFC instance can be in the following states:

- STARTED: the VNF/VNFC instance is up and running.
- STOPPED: the VNF/VNFC instance has been shut down. A VNF instance is stopped if all its VNFC instances are also stopped.

In the state STOPPED, the virtualised container(s), where the VNFC instance(s) of the VNF run, are shut down but not terminated. In addition, if the workflow requires a graceful stop, as part of this process the VNFM (producer of the interface) will interact with VNF/EM to gracefully stop the VNF/VNFC application. Once a VNF is instantiated, i.e. all instantiation steps have been completed, the VNF/VNFC instance is in the state STARTED.

Figure 7.2.11.1-1 illustrates the VNF/VNFC operate state diagram. The desired change of state is indicated as an input in the OperateVnfRequest operation.

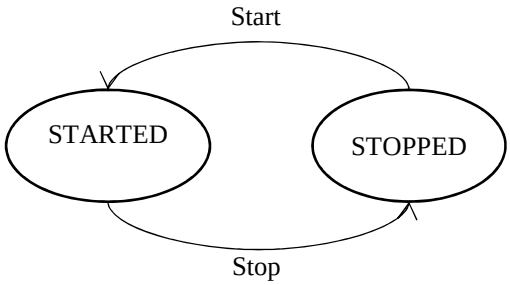

Figure 7.2.11.1-1: Operate VNF/VNFC state diagram

It depends on the VNF capabilities, and is declared in the VNFD, whether this operation is supported for a particular VNF.

Table 7.2.11.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.2.11.1-1: Operate VNF operation

Message	Requirement	Direction
OperateVnfRequest	Mandatory	EM → VNFM
OperateVnfResponse	Mandatory	VNFM → EM

7.2.11.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.11.2-1.

Table 7.2.11.2-1: Operate VNF operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.
vnfcInstanceId	M	0..N	Identifier	Identifier of the VNFC instance. Cardinality can be "0" to denote that the request applies to the whole VNF and not a specific VNFC instance.
changeStateTo	M	1	Enum	The desired state to change the VNF/VNFC to. Permitted values are: start, stop.
stopType	M	0..1	Enum	It signals whether forceful or graceful stop is requested. Allowed values are: forceful and graceful. In case of forceful stop, the VNF is stopped immediately. Note that if the VNF is still in service, this may adversely impact network service, and therefore, operator policies apply to determine if forceful stop is allowed in the particular situation. In case of graceful stop, the VNFM first arranges to take the VNF out of service (by means out of scope of the present specification, e.g. involving interaction with EM, if required). Once this is successful, or after a timeout, the VNFM stops the VNF. Only applicable when changing state to stop.
gracefulStopTimeout	M	0..1	TimeDuration	The time interval to wait for the VNF to be taken out of service during graceful stop, before stopping the VNF. If not given, it is expected that the VNFM waits for the successful taking out of service of the VNF, no matter how long it takes, before stopping the VNF (see note). Minimum timeout or timeout range are specified by the VNF vendor (e.g. defined in the VNFD or communicated by other means). The parameter is not relevant in case of forceful stop.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the EM as input to the Operate VNF operation, specific to the VNF being operated as declared in the VNFD (see clause 7.1.5.8 in ETSI GS NFV-IFA 011 [4]).
NOTE: This implies that no VNF stop will be attempted if taking the VNF out of service fails or hangs.				

7.2.11.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.11.3-1.

Table 7.2.11.3-1: Operate VNF operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the VNF lifecycle operation occurrence.

7.2.11.4 Operation results

In case of success, the state of the VNF/VNFC has been changed, and the associated instance of a VnfInfo information element has been updated.

In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

The producer shall first return the lifecycleOperationOccurrenceId and second send the "start" LCM Operation Occurrence Notification before additional notifications or messages as part of this operation are issued, or operations towards the NFVO or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the VNFM shall send the "result" LCM Operation Occurrence Notification.

7.2.12 Modify VNF Information operation

7.2.12.1 Description

This operation allows updating information about a VNF instance.

This operation shall be supported for all VNFs.

Table 7.2.12.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.2.12.1-1: Modify VNF Information operation

Message	Requirement	Direction
ModifyVnfInfoRequest	Mandatory	EM → VNFM
ModifyVnfInfoResponse	Mandatory	VNFM → EM

7.2.12.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.12.2-1.

Table 7.2.12.2-1: Modify VNF Information operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance for which the writeable attributes of VnfInfo are requested to be modified.
newValues	M	1..N	KeyValuePair	Contains the set of attributes to update. The key in the KeyValuePair indicates the name of an attribute that is writable through the interface whose value is to be updated. The value in the KeyValuePair indicates the new attribute value. See note 1.
vnfcConfigurationData	M	0..N	VnfcConfigurationKvp	Configuration data related to VNFC instances. See note 2.
NOTE 1: Cardinality "0" applies if no attributes of the VNF instance, including VNF configurable properties, are requested to be modified. NOTE 2: Cardinality "0" if no configurable properties of individual VNFC instances are requested to be modified.				

7.2.12.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.12.3-1.

Table 7.2.12.3-1: Modify VNF Information operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the VNF lifecycle operation occurrence.

7.2.12.4 Operation results

In case of success:

- if the operation handles changes to the VNF or VNFC configurable properties, the configuration in the VNF or VNFC has been modified according to the input parameters specified in the operation;
- if the operation handles other changes to the VNF instance information, the VNF information has been changed according to the input parameters specified in the operation.

In case of failure, appropriate error information is provided in the "result" LCM operation occurrence notification. In particular, error information shall indicate the reason why the requested attribute has not been updated, e.g. changing the value of the attribute is not supported, input attribute name is not recognized, etc.

The producer shall first return the lifecycleOperationOccurrenceId and second send the "start" LCM operation occurrence notification before additional notifications or messages as part of this operation are issued, or operations towards the NFVO or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the VNFM shall send the "result" LCM operation occurrence notification.

7.2.13 Get Operation Status operation

7.2.13.1 Description

This operation provides the status of a VNF lifecycle management operation. This means, it is not a VNF lifecycle management operation itself, but an operation on VNF lifecycle management operations. Therefore, this operation shall be supported for all VNFs.

Table 7.2.13.1-1 lists the information flow exchanged between the EM/VNF and the VNFM.

Table 7.2.13.1-1: Get Operation Status operation

Message	Requirement	Direction
GetOperationStatusRequest	Mandatory	EM → VNFM VNF → VNFM
GetOperationStatusResponse	Mandatory	VNFM → EM VNFM → VNF

7.2.13.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.13.2-1.

Table 7.2.13.2-1: Get Operation Status operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	Identifier of the VNF lifecycle operation occurrence.

7.2.13.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.13.3-1.

Table 7.2.13.3-1: Get Operation Status operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
operationStatus	M	1	Enum	Indicates the operation status (which includes, for example: Processing, Successfully done, Failed, but can also include operation-specific states).

7.2.13.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result.

7.2.14 Subscribe operation

7.2.14.1 Description

This operation enables the consumer (EM or VNF) to subscribe with a filter for the notifications sent by the producer (VNFM) which are related to VNF lifecycle management operation occurrences as well as creation/deletion of VNF instance identifiers and the associated VnfInfo information element instances.

NOTE: Specification of filtering mechanism is left for the protocol design stage.

Table 7.2.14.1-1 lists the information flow exchanged between the consumer (EM or VNF) and the producer (VNFM).

Table 7.2.14.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	EM → VNFM VNF → VNFM
SubscribeResponse	Mandatory	VNFM → EM VNFM → VNF

7.2.14.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.14.2-1.

Table 7.2.14.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting e.g. the VNF instances of interest and the specific types of changes (see note).
NOTE: When subscribing for notifications regarding the creation of VNF identifiers and the associated VNF information object instances, selecting the VNF instances in the filter is not possible.				

7.2.14.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.14.3-1.

Table 7.2.14.3-1: Subscribe operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription realized.

7.2.14.4 Operation results

After successful subscription, the consumer (EM or VNF) is registered to receive notifications related to VNF lifecycle management operation occurrences as well as creation/deletion of VNF instance identifiers and the associated VnfInfo information element instances.

The result of the operation shall indicate if the subscription has been successful or not with a standard success/error result. For a particular subscription, only notifications matching the filter will be delivered to the consumer (EM or VNF).

7.2.15 Notify operation

7.2.15.1 Description

This operation notifies a subscriber about events related to VNF lifecycle changes as well as creation/deletion of VNF instance identifiers and the associated VnfInfo information element instances.

This operation distributes notifications to subscribers. It is a one-way operation issued by the producer (VNFM) that cannot be invoked as an operation by the consumer (EM or VNF). In order to receive notifications, the consumer (EM or VNF) has to perform an explicit Subscribe operation beforehand.

Table 7.2.15.1-1 lists the information flow exchanged between the consumer (EM or VNF) and the product (VNFM).

Table 7.2.15.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	VNFM → EM VNFM → VNF

The following notifications can be notified/sent by this operation:

- VnfLcmOperationOccurrenceNotification (see clause 9.5.2).
- VnfIdentifierCreationNotification (see clause 9.5.7).
- VnfIdentifierDeletionNotification (see clause 9.5.8).

7.2.16 Terminate Subscription operation

7.2.16.1 Description

This operation enables the EM to terminate a particular subscription.

Table 7.2.16.1-1 lists the information flow exchanged between the EM and the VNFM.

Table 7.2.16.1-1: Terminate Subscription operation

Message	Requirement	Direction
TerminateSubscriptionRequest	Mandatory	EM → VNFM
TerminateSubscriptionResponse	Mandatory	VNFM → EM

7.2.16.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.16.2-1.

Table 7.2.16.2-1: Terminate Subscription operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription to be terminated.

7.2.16.3 Output parameters

None.

7.2.16.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the EM will not receive notifications related that subscription any longer. The result of the operation shall indicate if the subscription termination has been successful or not with a standard success/error result.

7.2.17 Query Subscription Info operation

7.2.17.1 Description

This operation enables the EM to query information about subscriptions.

Table 7.2.17.1-1 lists the information flow exchanged between the EM and the VNFM.

Table 7.2.17.1-1: Query Subscription operation

Message	Requirement	Direction
QuerySubscriptionInfoRequest	Mandatory	EM → VNFM
QuerySubscriptionInfoResponse	Mandatory	VNFM → EM

7.2.17.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.17.2-1.

Table 7.2.17.2-1: Query Subscription Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filtering criteria to select one or a set of subscriptions. Details are left for the protocol design stage.

7.2.17.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.17.3-1.

Table 7.2.17.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query. Details are left for the protocol design stage.

7.2.17.4 Operation results

After successful operation, the VNFM has queried the internal subscription objects. The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular query, information about the subscriptions to notifications related to VNF lifecycle management that the EM has access to and that are matching the filter shall be returned.

7.2.18 Change External VNF Connectivity operation

7.2.18.1 Description

This operation enables changing the external connectivity of a VNF instance. The types of changes that this operation supports are:

- Disconnect the external CPs that are connected to a particular external VL, and connect them to a different external VL.

- Change the connectivity parameters of the existing external CPs, including changing addresses.

NOTE: Depending on the capabilities of the underlying VIM resources, certain changes (e.g. modifying the IP address assignment) might not be supported without deleting the resource and creating another one with the modified configuration.

VNFs shall support this operation. This operation may be service-disruptive.

Table 7.2.18.1-1 lists the information flow exchanged between the EM and the VNFM.

Table 7.2.18.1-1: Change External VNF Connectivity

Message	Requirement	Direction
ChangeExtVnfConnectivityRequest	Mandatory	EM → VNFM
ChangeExtVnfConnectivityResponse	Mandatory	VNFM → EM

7.2.18.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.18.2-1. The parameters passed for this operation override those passed at instantiation time.

Table 7.2.18.2-1: Change VNF External VLs operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.
extVirtualLink	M	0..N	ExtVirtualLinkData	Information about external VLs to change (e.g. connect the VNF to).
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the NFVO as input to the Change External VNF Connectivity operation, specific to the VNF of which the connectivity is being changed, as declared in the VNFD (see clause 7.1.5.10 in ETSI GS NFV-IFA 011 [4]).

The following behaviour applies for the changes that can be performed with this operation:

- To change the connection of external CP instances based on certain external CPDs from a "source" external VL to a different "target" external VL, the identifier of the "target" external VL shall be sent in the "extVirtualLinkId" attribute of the "extVirtualLink" parameter, and the "extCp" attributes of that parameter shall refer via the "cpdId" attribute to the external CPDs of the corresponding external connection point instances that are to be reconnected to the target external VL.

NOTE: This means that all CP instances based on a given external CPD will be reconnected. See clause A.3 for an illustration.

- To change the connectivity parameters of the external CPs connected to a particular external VL, including changing addresses, the identifier of that external VL shall be sent in the "extVirtualLinkId" attribute of the "extVirtualLink" parameter, and the "extCp" attribute of that parameter shall contain at least those entries with modified parameters.

7.2.18.3 Output parameters

None.

7.2.18.4 Operation results

In the case of success, the connectivity of the VNF has been changed according to the input parameters. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

The producer shall first return the lifecycleOperationOccurrenceId and second send the "start" LCM Operation Occurrence Notification before additional notifications or messages as part of this operation are issued, or operations towards the NFVO or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the VNFM shall send the "result" LCM Operation Occurrence Notification.

7.3 Void

7.4 VNF Performance Management interface

7.4.1 Description

This interface allows providing performance management (measurement results collection and notifications) related to VNFs. Performance information on a given VNF/VNFC results from performance information of the virtualised resources that is collected from the VIM and mapped to this VNF/VNFC instance.

Collection and reporting of performance information is controlled by a PM job that groups details of performance collection and reporting information.

When new performance information is available, the consumer is notified using the notification PerformanceInformationAvailableNotification (see clause 9.7.8). The details of the performance measurements are provided using the PerformanceReport information element (see clause 9.7.5).

NOTE: Delivery mechanism for the performance reports is left for the protocol design stage.

The following operations are defined for this interface which will be consumed by the EM:

- Create PM Job operation.
- Delete PM Jobs operation.
- Subscribe operation.
- Notify operation.
- Query PM Job operation.
- Create Threshold operation.
- Delete Thresholds operation.
- Query Threshold operation.
- Terminate Subscription operation.
- Query Subscription Info operation.

The following operations are defined for this interface which will be consumed by the VNF:

- Subscribe operation.
- Notify operation.
- Terminate Subscription operation.
- Query Subscription Info operation.

7.4.2 Create PM Job operation

7.4.2.1 Description

This operation will create a PM job, enabling an EM to specify a VNF/VNFC, that the VNFM is managing, for which it wants to receive performance information. This will allow the requesting EM to specify its performance information requirements with the VNFM.

The VNFM needs to issue a Subscribe request for PerformanceInformationAvailable notifications in order to know when new collected performance information is available.

Table 7.4.2.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.4.2.1-1: Create PM Job operation

Message	Requirement	Direction
CreatePmJobRequest	Mandatory	EM → VNFM
CreatePmJobResponse	Mandatory	VNFM →EM

7.4.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.2.2-1.

Table 7.4.2.2-1: Create PM Job operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
sourceSelector	M	1	ObjectSelection	Defines the VNFs and/or VNFCs for which performance information is requested to be collected.
performanceMetric	M	0..N	String	This defines the type of performance metric(s) for the specified VNFs. At least one of the two attributes (performance metric or group) shall be present.
performanceMetricGroup	M	0..N	String	Group of performance metrics. A metric group is a pre-defined list of metrics, known to the producer that it can decompose to individual metrics. At least one of the two attributes (performance metric or group) shall be present.
collectionPeriod	M	1	Enum	Specifies the periodicity at which the VNFM will collect performance information (see note).
reportingPeriod	M	1	Enum	Specifies the periodicity at which the VNFM will report to the EM about performance information (see note).
reportingBoundary	O	0..1	Not specified	Identifies a boundary after which the reporting will stop. The boundary shall allow a single reporting as well as periodic reporting up to the boundary.
NOTE: At the end of each reportingPeriod, the VNFM will inform EM about availability of the performance data collected for each completed collection period during this reportingPeriod. While the exact definition of the types for collectionPeriod and reportingPeriod is left for further protocol specification, it is recommended that the reportingPeriod be equal or a multiple of the collectionPeriod. In the latter case, the performance data for the collection periods within one reporting period would be reported together.				

7.4.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.2.3-1.

Table 7.4.2.3-1: Create PM Job operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
pmJobId	M	1	Identifier	Identifier of the created PM job

7.4.2.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result.

The pmJobId is only be returned when the operations has been successful.

7.4.3 Delete PM Jobs operation

7.4.3.1 Description

This operation will delete one or more PM job(s).

Table 7.4.3.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.4.3.1-1: Delete PM Jobs operation

Message	Requirement	Direction
DeletePmJobsRequest	Mandatory	EM → VNFM
DeletePmJobsResponse	Mandatory	VNFM → EM

7.4.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.3.2-1.

Table 7.4.3.2-1: Delete PM Jobs operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pmJobId	M	1..N	Identifier	Identifiers of the PM jobs to be deleted.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to delete multiple PM Jpbs in one request, or as a series of requests that delete one PM Job at a time.				

7.4.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.3.3-1.

Table 7.4.3.3-1: Delete PM Jobs operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deletedPmJobId	M	1..N	Identifier	Identifiers of the PM jobs successfully deleted.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to delete multiple PM Jpbs in one request, or as a series of requests that delete one PM Job at a time.				

7.4.3.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result.

7.4.4 Subscribe operation

7.4.4.1 Description

This operation enables the EM/VNF to subscribe with a filter for the notifications related to performance information with the VNFM.

NOTE: Specification of filtering mechanism is left for the protocol design stage.

Table 7.4.4.1-1 lists the information flow exchanged between the VNFM and the EM/VNF.

Table 7.4.4.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	EM → VNFM VNF → VNFM
SubscribeResponse	Mandatory	VNFM → EM VNFM → VNF

7.4.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.4.2-1.

Table 7.4.4.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting notifications. The filter can be on VNF/VNFC, type of notification or attribute of the notification.

7.4.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.4.3-1.

Table 7.4.4.3-1: Subscribe operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription returned.

7.4.4.4 Operation results

As a result of this operation, the VNFM shall indicate to the EM/VNF in the subscribeResponse message whether the subscription was successful or not.

For a particular subscription, only notifications matching the filter will be delivered to the consumer.

7.4.5 Notify operation

7.4.5.1 Description

This operation distributes notifications to subscribers. It is a one-way operation issued by the VNFM that cannot be invoked as an operation by the consumer (EM/VNF). In order to receive notifications, the EM/VNF shall have a subscription.

Table 7.4.5.1-1 lists the information flow exchanged between the VNFM and the EM/VNF.

Table 7.4.5.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	VNFM → EM VNFM → VNF

The following notifications can be notified/sent by this operation:

- PerformanceInformationAvailableNotification (see clause 9.7.8).
- ThresholdCrossedNotification (see clause 9.7.9).

7.4.6 Query PM Job operation

7.4.6.1 Description

This operation will enable the EM to solicit from the VNFM the details of one or more PM job(s).

This operation is not returning performance reports.

Table 7.4.6.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.4.6.1-1: Query PM Job operation

Message	Requirement	Direction
QueryPmJobRequest	Mandatory	EM → VNFM
QueryPmJobResponse	Mandatory	VNFM → EM

7.4.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.6.2-1.

Table 7.4.6.2-1: Query PM Job operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filter defining the PM Jobs on which the query applies. It can be a single identifier, multiple identifiers or a wildcard.

7.4.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.6.3-1.

Table 7.4.6.3-1: Query PM Job operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
pmJob	M	0..N	PmJob	Details of PM jobs matching the input filter.

7.4.6.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result.

7.4.7 Create Threshold operation

7.4.7.1 Description

This operation will allow the EM to create a threshold to specify threshold levels on specified performance metric and VNF/VNFC for which notifications will be generated when crossed.

Creating a threshold does not trigger collection of metrics. In order for the threshold to be active, there needs to be a PM job collecting the needed metric for the selected entities.

Table 7.4.7.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.4.7.1-1: Create Threshold operation

Message	Requirement	Direction
CreateThresholdRequest	Mandatory	EM → VNFM
CreateThresholdResponse	Mandatory	VNFM → EM

7.4.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.7.2-1.

Table 7.4.7.2-1: Create Threshold operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
sourceSelector	M	1	ObjectSelection	Defines the VNF/VNFC for which the threshold will be defined.
performanceMetric	M	1	String	Defines the performance metric on which the threshold will be defined.
thresholdType	M	1	Enum	Defines the type of threshold. The list of possible values is left for the protocol design stage and might include: single/ multi valued threshold, static/dynamic threshold, template based threshold, etc.
thresholdDetails	M	1	Not specified	Details of the threshold: value to be crossed, and direction in which it is crossed, details on the notification to be generated, etc.

7.4.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.7.3-1.

Table 7.4.7.3-1: Create Threshold operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
thresholdId	M	1	Identifier	Identifier of created threshold.

7.4.7.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result.

The thresholdId is only be returned when the operations has been successful.

7.4.8 Delete Thresholds operation

7.4.8.1 Description

This operation will allow the EM to delete one or more existing threshold(s).

Table 7.4.8.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.4.8.1-1: Delete Thresholds operation

Message	Requirement	Direction
DeleteThresholdsRequest	Mandatory	EM → VNFM
DeleteThresholdsResponse	Mandatory	VNFM → EM

7.4.8.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.8.2-1.

Table 7.4.8.2-1: Delete Thresholds operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
thresholdId	M	1..N	Identifier	Identifiers of the thresholds to be deleted.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to delete multiple thresholds in one request, or as a series of requests that delete one threshold at a time.				

7.4.8.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.8.3-1.

Table 7.4.8.3-1: Delete Thresholds operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deletedThresholdId	M	1..N	Identifier	Identifiers of the thresholds that have been deleted successfully.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to delete multiple thresholds in one request, or as a series of requests that delete one threshold at a time.				

7.4.8.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result.

7.4.9 Query Threshold operation

7.4.9.1 Description

This operation will allow the EM to query the details of an existing threshold.

Table 7.4.9.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 7.4.9.1-1: Query Threshold operation

Message	Requirement	Direction
QueryThresholdRequest	Mandatory	EM → VNFM
QueyThresholdResponse	Mandatory	VNFM → EM

7.4.9.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.9.2-1.

Table 7.4.9.2-1: Query Threshold operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filter defining the thresholds on which the query applies. It can be a single identifier, multiple identifiers or a wildcard.

7.4.9.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.9.3-1.

Table 7.4.9.3-1: Query Threshold operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
threshold	M	0..N	Threshold	List of threshold details matching the input filter.

7.4.9.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result.

7.4.10 Terminate Subscription operation

7.4.10.1 Description

This operation enables the EM/VNF to terminate a particular subscription.

Table 7.4.10.1-1 lists the information flow exchanged between the EM/VNF and the VNFM.

Table 7.4.10.1-1: Terminate Subscription operation

Message	Requirement	Direction
TerminateSubscriptionRequest	Mandatory	EM → VNFM VNF → VNFM
TerminateSubscriptionResponse	Mandatory	VNFM → EM VNFM → VNF

7.4.10.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.10.2-1.

Table 7.4.10.2-1: Terminate Subscription operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription to be terminated.

7.4.10.3 Output parameters

None.

7.4.10.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the EM/VNF will not receive notifications related that subscription any longer. The result of the operation shall indicate if the subscription termination has been successful or not with a standard success/error result.

7.4.11 Query Subscription Info operation

7.4.11.1 Description

This operation enables the EM/VNF to query information about subscriptions.

Table 7.4.11.1-1 lists the information flow exchanged between the EM/VNF and the VNFM.

Table 7.4.11.1-1: Query Subscription operation

Message	Requirement	Direction
QuerySubscriptionInfoRequest	Mandatory	EM → VNFM VNF → VNFM
QuerySubscriptionInfoResponse	Mandatory	VNFM → EM VNFM → VNF

7.4.11.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.11.2-1.

Table 7.4.11.2-1: Query Subscription Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filtering criteria to select one or a set of subscriptions. Details are left for the protocol design stage.

7.4.11.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.11.3-1.

Table 7.4.11.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query. Details are left for the protocol design stage.

7.4.11.4 Operation results

After successful operation, the VNFM has queried the internal subscription objects. The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular query, information about the subscriptions to notifications related to VNF performance management that the EM/VNF has access to and that are matching the filter shall be returned.

7.5 VNF Fault Management interface

7.5.1 Description

This interface shall allow the VNFM to provide alarms related to the VNF(s) and its VNFC(s) visible to the consumer.

Virtualised resource alarms collected by the VNFM well be filtered, correlated and modified by the VNFM and mapped to the corresponding VNF instance, resulting in alarms on the corresponding VNF and its VNFC(s).

The fault management interface provided by the VNFM and consumed by the EM/VNF shall support the following operations:

- Subscribe operation (Subscription of EM/VNF with the VNFM for the notifications related to the alarms).
- Notify operation (Notifications of alarms or alarm state change from VNFM to EM/VNF).
- Get alarm list operation (Accessing active alarms from the VNFM to EM/VNF).
- Terminate Subscription operation.
- Query Subscription Info operation.
- Escalate perceived severity operation (Sharing of perceived severity with the VNFM).

The "Escalate Perceived Severity" (or "Escalate PerSev" for short) operation shall be supported by the VNFM and may be supported by the EM/VNF.

The "Acknowledge alarms" operation (Allowing alarm acknowledgement from the EM/VNF to the VNFM) shall be supported by VNFM and may be supported by EM/VNF.

7.5.2 Subscribe operation

7.5.2.1 Description

This operation enables the EM/VNF to subscribe with a filter for the notifications related to VNF alarms sent by the VNFM.

NOTE: Specification of filtering mechanism is left for the protocol design stage.

Table 7.5.2.1-1 lists the information flow exchanged between the VNF/EM and the VNFM.

Table 7.5.2.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	EM → VNFM VNF → VNFM
SubscribeResponse	Mandatory	VNFM → EM VNFM → VNF

7.5.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.2.2-1.

Table 7.5.2.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting VNF(s) and its VNFC(s) and related alarms. This can contain the VNF/VNFC information, fault type, severity and cause of the alarm.

7.5.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.5.2.3-1.

Table 7.5.2.3-1: Subscribe operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription returned.

7.5.2.4 Operation results

As a result of this operation, the VNFM shall indicate to the EM/VNF in the SubscribeResponse message whether the subscription was successful or not.

For a particular subscription, only notifications matching the filter will be delivered to the consumer.

7.5.3 Notify operation

7.5.3.1 Description

This operation distributes notifications to subscribers. It is a one-way operation issued by the VNFM towards the EM/VNF that cannot be invoked as an operation by the consumer (EM/VNF).

In order to receive notifications, the EM/VNF shall have a subscription. Table 7.5.3.1-1 lists the information flow exchanged between the VNFM and the VNF/EM.

Table 7.5.3.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	VNFM → EM VNFM → VNF

The following notifications can be notified/sent by this operation:

- AlarmNotification. See clause 9.3.2.
- AlarmClearedNotification. See clause 9.3.3.
- AlarmListRebuiltNotification (see clause 9.3.6).

7.5.4 Get Alarm List operation

7.5.4.1 Description

This operation enables the EM/VNF to query the active alarms from the VNFM.

Table 7.5.4.1-1 lists the information flow exchanged between the VNFM and the EM/VNF.

Table 7.5.4.1-1: Get Alarm List operation

Message	Requirement	Direction
GetAlarmListRequest	Mandatory	EM → VNFM VNF → VNFM
GetAlarmListResponse	Mandatory	VNFM → EM VNFM → VNF

7.5.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.4.2-1.

Table 7.5.4.2-1: Get Alarm List operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting alarms. This can contain the list of the VNF identifier(s), VNFC identifier(s), fault type, severity and cause.

7.5.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.5.4.3-1.

Table 7.5.4.3-1: Get Alarm List operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
alarm	M	0..N	Alarm	Information about an alarm including AlarmId, affected VNF identifier, affected VNFC identifier and FaultDetails. The cardinality can be "0" to indicate that no Alarm could be retrieved based on the input Filter information (e.g. no matching alarm).

7.5.4.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular request, only alarms matching the filter will be delivered to the EM/VNF.

7.5.5 Terminate Subscription operation

7.5.5.1 Description

This operation enables the EM/VNF to terminate a particular subscription.

Table 7.5.5.1-1 lists the information flow exchanged between the EM/VNF and the VNFM.

Table 7.5.5.1-1: Terminate Subscription operation

Message	Requirement	Direction
TerminateSubscriptionRequest	Mandatory	EM → VNFM VNF → VNFM
TerminateSubscriptionResponse	Mandatory	VNFM → EM VNFM → VNF

7.5.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.5.2-1.

Table 7.5.5.2-1: Terminate Subscription operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription to be terminated.

7.5.5.3 Output parameters

None.

7.5.5.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the EM/VNF will not receive notifications related that subscription any longer. The result of the operation shall indicate if the subscription termination has been successful or not with a standard success/error result.

7.5.6 Query Subscription Info operation

7.5.6.1 Description

This operation enables the EM/VNF to query information about subscriptions.

Table 7.5.6.1-1 lists the information flow exchanged between the EM/VNF and the VNFM.

Table 7.5.6.1-1: Query Subscription operation

Message	Requirement	Direction
QuerySubscriptionInfoRequest	Mandatory	EM → VNFM VNF → VNFM
QuerySubscriptionInfoResponse	Mandatory	VNFM → EM VNFM → VNF

7.5.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.6.2-1.

Table 7.5.6.2-1: Query Subscription Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filtering criteria to select one or a set of subscriptions. Details are left for the protocol design stage.

7.5.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.5.6.3-1.

Table 7.5.6.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query. Details are left for the protocol design stage.

7.5.6.4 Operation results

After successful operation, the VNFM has queried the internal subscription objects. The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular query, information about the subscriptions to notifications related to VNF fault management that the EM/VNF has access to and that are matching the filter shall be returned.

7.5.7 Escalate perceived severity operation

7.5.7.1 Description

This operation enables the EM/VNF to share its view of the perceived severity of an alarm with the VNFM. This operation does not directly modify the value of perceived severity attribute in the alarm information element within the VNFM. VNFM implementation (e.g. controlled by operator configuration) will determine how it should act upon receipt of the requested change in perceived severity. Some requests from the EM/VNF may be respected and applied directly by the VNFM, while others may be ignored by the VNFM.

Table 7.5.7.1-1 lists the information flow exchanged between the VNFM and the EM/VNF.

Table 7.5.7.1-1: Escalate PerSev operation

Message	Requirement	Direction
EscalatePerSevRequest	Mandatory	EM → VNFM VNF → VNFM
EscalatePerSevResponse	Mandatory	VNFM → EM VNFM → VNF
NOTE: If the EM/VNF supports the Escalate PerSev operation, then both the "EscalatePerSevRequest" and "EscalatePerSevResponse" messages shall be supported.		

7.5.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.7.2-1.

Table 7.5.7.2-1: Escalate PerSev operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
alarmId	M	1	Identifier (Reference to Alarm)	Identifier of the alarm
perceivedSeverity	M	1	Enum	New value of the alarm severity (as perceived by the consumer). Permitted values include: • Critical • Major • Minor • Warning • Indeterminate • Cleared

7.5.7.3 Output parameters

There are no output parameters defined for this operation.

7.5.7.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result. If/when the VNFM makes the change to the perceivedSeverity for the given alarm based on this operation, the VNFM will send an alarm notification to the EM/VNF (and the NFVO) containing the updated value of perceivedSeverity.

7.5.8 Acknowledge alarms operation

7.5.8.1 Description

This operation enables the EM/VNF to acknowledge alarms at VNFM.

Table 7.5.8.1-1 lists the information flow exchanged between the VNFM and the EM/VNF.

Table 7.5.8.1-1: Acknowledge alarms operation

Message	Requirement	Direction
AcknowledgeAlarmsRequest	Mandatory	EM → VNFM VNF → VNFM
AcknowledgeAlarmsResponse	Mandatory	VNFM → EM VNFM → VNF
NOTE: If the EM/VNF supports the Acknowledge alarms operation, then both AcknowledgeAlarmsRequest and AcknowledgeAlarmsResponse shall be supported.		

7.5.8.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.8.2-1.

Table 7.5.8.2-1: Acknowledge alarms operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
alarmId	M	1..N	Identifier (Reference to Alarm)	Identifier of an individual alarm to be acknowledged, or multiple identifiers of the alarms to be acknowledged. See note.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to acknowledge multiple alarms in one request, or as a series of requests that acknowledge one alarm at a time.				

7.5.8.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.5.8.3-1.

Table 7.5.8.3-1: Acknowledge alarms operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
acknowledgedAlarmId	M	1..N	Identifier (Reference to Alarm)	Identifier of an individual alarm that is acknowledged, or multiple identifiers of the alarms that are acknowledged. See note.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to acknowledge multiple alarms in one request, or as a series of requests that acknowledge one alarm at a time.				

7.5.8.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result.

7.6 Void

8 EM exposed interfaces

8.1 Introduction

This clause defines the interfaces exposed by the EM towards the VNFM over the Ve-Vnfm reference point.

NOTE: The fact that information elements and attributes are presented in tabular form does not preclude protocol designs in which these information elements and attributes are encoded in different parts of request and response messages. For example, in a RESTful interface, parts of them may be encoded in the URL, in the message header, in the message body or any combination thereof.

8.2 Indicator Interface

8.2.1 Description

This interface allows the EM to provide information on value changes of VNF related indicators. VNF related indicators are declared in the VNFD.

The following operations are defined for this interface:

- Subscribe.
- Notify.
- Get Indicator Value.
- Terminate Subscription.
- Query Subscription Info.

8.2.2 Subscribe operation

8.2.2.1 Description

This operation enables the VNFM to subscribe with a filter for the notifications related to VNF indicator value changes sent by the EM.

NOTE: Specification of filtering mechanism is left for the protocol design stage.

Table 8.2.2.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 8.2.2.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	VNFM → EM
SubscribeResponse	Mandatory	EM → VNFM

8.2.2.2 Input parameters

The input parameters sent when invoking the operation are listed in table 8.2.2.2-1.

Table 8.2.2.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting VNFs and related indicators.

8.2.2.3 Output parameters

The output parameters returned by the operation are listed in table 8.2.2.3-1.

Table 8.2.2.3-1: Subscribe operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription returned.

8.2.2.4 Operation results

As a result of this operation, the EM shall indicate to the VNFM in the SubscribeResponse message whether the subscription was successful or not.

For a particular subscription, only notifications matching the filter will be delivered to the consumer.

8.2.3 Notify operation

8.2.3.1 Description

This operation distributes notifications to subscribers. It is a one-way operation issued by the EM towards the VNFM that cannot be invoked as an operation by the consumer (VNFM).

In order to receive notifications, the VNFM shall have a subscription.

Table 8.2.3.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 8.2.3.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	EM → VNFM

The following notification can be notified/sent by this operation:

- IndicatorValueChangeNotification see clause 9.6.2.

8.2.4 Get Indicator Value operation

8.2.4.1 Description

This operation enables VNFM to request the actual value of a given indicator from the EM.

Table 8.2.4.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 8.2.4.1-1: Get Indicator Value operation

Message	Requirement	Direction
GetIndicatorValueRequest	Mandatory	VNFM → EM
GetIndicatorValueResponse	Mandatory	EM → VNFM

8.2.4.2 Input parameters

The input parameters sent when invoking the operation are listed in table 8.2.4.2-1.

Table 8.2.4.2-1: Get Indicator Value operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting VNFs and related indicators.

8.2.4.3 Output parameters

The output parameters returned by the operation are listed in table 8.2.4.3-1.

Table 8.2.4.3-1: Get Indicator Value operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
indicatorInformation	M	0..N	IndicatorInformation	The requested indicator values as a complex structure having the VNF Instance ID, Indicator and the value of the Indicator.

8.2.4.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular request, only indicators matching the filter will be delivered to the VNFM.

8.2.5 Terminate Subscription operation

8.2.5.1 Description

This operation enables the VNFM to terminate a particular subscription.

Table 8.2.5.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 8.2.5.1-1: Terminate Subscription operation

Message	Requirement	Direction
TerminateSubscriptionRequest	Mandatory	VNFM → EM
TerminateSubscriptionResponse	Mandatory	EM → VNFM

8.2.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 8.2.5.2-1.

Table 8.2.5.2-1: Terminate Subscription operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifier of the subscription to be terminated.

8.2.5.3 Output parameters

None.

8.2.5.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the VNFM will not receive notifications related that subscription any longer. The result of the operation shall indicate if the subscription termination has been successful or not with a standard success/error result.

8.2.6 Query Subscription Info operation

8.2.6.1 Description

This operation enables the VNFM to query information about subscriptions.

Table 8.2.6.1-1 lists the information flow exchanged between the VNFM and the EM.

Table 8.2.6.1-1: Query Subscription operation

Message	Requirement	Direction
QuerySubscriptionInfoRequest	Mandatory	VNFM → EM
QuerySubscriptionInfoResponse	Mandatory	EM → VNFM

8.2.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 8.2.6.2-1.

Table 8.2.6.2-1: Query Subscription Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filtering criteria to select one or a set of subscriptions. Details are left for the protocol design stage.

8.2.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 8.2.6.3-1.

Table 8.2.6.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query. Details are left for the protocol design stage.

8.2.6.4 Operation results

After successful operation, the EM has queried the internal subscription objects. The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular query, information about the subscriptions to notifications related to VNF indicator value changes that the VNFM has access to and that are matching the filter shall be returned.

9 Information elements exchanged over reference point Ve-Vnfm

9.1 Introduction

This clause defines, or references, definitions of information elements used in the interfaces defined in the present document.

The specification of the following information elements is left for the protocol design stage:

- String.
- Integer.
- Identifier.
- Filter.
- DateTime.
- Value.
- Version.
- KeyValuePair.

9.2 Information elements and notifications related to VNF Configuration Management

9.2.1 Introduction

This clause defines information elements and notifications related to VNF configuration management.

9.2.2 VnfConfiguration information element

9.2.2.1 Description

This data type provides the list of attributes for the configuration of a VNF instance.

9.2.2.2 Attributes

The VnfConfiguration information element shall follow the indications provided in table 9.2.2.2-1.

Table 9.2.2.2-1: Attributes of the VnfConfiguration information element

Attribute	Qualifier	Cardinality	Content	Description
cpConfiguration	M	0..N	CpConfiguration	External CPs See clause 9.2.4.
dhcpServer	M	0..1	Not specified	Identifies a DHCP server that the VNF can use to obtain IP addresses to be assigned to its external CPs.
vnfSpecificData	M	0..N	KeyValuePair	Configuration object containing values of VNF configurable properties (see VnfConfigurableProperties in clause 7.1.12 of ETSI GS NFV-IFA 011 [4]) applicable to whole VNF instance. See notes 1 and 2.
NOTE 1: Cardinality "0" applies to information elements that are not requested to be modified. NOTE 2: These data can be generated by LCM scripts available in the VNF package and executed by the VNFM, or be provided by the EM or NFVO.				

9.2.3 VnfcConfiguration information element

9.2.3.1 Description

This data type provides the list of attributes for the configuration of a VNFC instance.

9.2.3.2 Attributes

The VnfcConfiguration information element shall follow the indications provided in table 9.2.3.2-1.

Table 9.2.3.2-1: Attributes of the VnfcConfiguration information element

Attribute	Qualifier	Cardinality	Content	Description
vnfcId	M	1	Identifier	Uniquely identifies a VNFC instance within the namespace of a specific VNF instance.
cp	M	0..N	CpConfiguration	Internal CPs. See clause 9.2.4.
dhcpServer	M	0..1	Not specified	Identifies a DHCP server that the VNF can use to obtain IP addresses to be assigned to its CPs.
vnfcSpecificData	M	0..N	KeyValuePair	Configuration object containing values of VNFC configurable properties (see VnfcConfigurableProperties in clause 7.1.6.7 of ETSI GS NFV-IFA 011 [4] applicable to a specific VNFC instance. See note.
NOTE: These data can be generated by LCM scripts available in the VNF package and executed by the VNFM, or be provided by the EM or NFVO.				

9.2.4 CpConfiguration information element

9.2.4.1 Description

This data type provides the list of attributes for the configuration of a CP instance.

9.2.4.2 Attributes

The CpConfiguration information element shall follow the indications provided in table 9.2.4.2-1.

Table 9.2.4.2-1: Attributes of the CpConfiguration information element

Attribute	Qualifier	Cardinality	Content	Description
cpId	M	1	Identifier	Uniquely identifies a CP instance within the namespace of a specific VNF instance or VNFC instance.
cpLabel	M	1	Not specified	Uniquely identifies a type of CP instance within the namespace of a VNFD.
address	M	1..N	CpAddress	Address and Port assigned to the CP. See clause 9.2.5.

9.2.5 CpAddress information element

9.2.5.1 Description

This data type provides the list of attributes associated to a CP instance address.

9.2.5.2 Attributes

The CpAddress information element shall follow the indications provided in table 9.2.5.2-1.

Table 9.2.5.2-1: Attributes of the CpAddress information element

Attribute	Qualifier	Cardinality	Content	Description
address	M	0..1		The address assigned to the CP instance (e.g. IP address, MAC address, etc.). It shall be provided for configuring a fixed address.
useDynamicAddress	M	0..1	Boolean	It determines whether an address shall be assigned dynamically. It shall be provided if a dynamic address needs to be configured on the CP.
port	M	0..1	Not specified	The port assigned to the CP instance (e.g. IP port number, Ethernet port number, etc.).

9.2.6 VnfcConfigurationKvp information element

9.2.6.1 Description

This information element provides the list of key value pairs for the configuration of a VNFC instance.

9.2.6.2 Attributes

The VnfcConfigurationKvp information element shall follow the indications provided in table 9.2.6.2-1.

Table 9.2.6.2-1: Attributes of the VnfcConfigurationKvp information element

Attribute	Qualifier	Cardinality	Content	Description
vnfcId	M	1	Identifier	Uniquely identifies a VNFC instance within the namespace of a specific VNF instance.
vnfcConfigKvp	M	1..N	KeyValuePair	Configuration data for the VNFC instance. Configuration data can include values for the properties that are declared as configurable in the VNFD (see VnfcConfigurableProperties in clause 7.1.6.7 of ETSI GS NFV-IFA 011 [4]).

9.2.7 Void

9.3 Information elements and notifications related to VNF Fault Management

9.3.1 Introduction

This clause defines information elements and notifications related to VNF Fault Management.

9.3.2 AlarmNotification

9.3.2.1 Description

This notification informs the receiver of alarms related to the VNFs and their constituent VNFCs managed by the VNFM. Alarms are created in response to:

- faults detected by the VNFM; and
- faults generated due to changes in the state of virtualised resources used by the VNFs and their constituent VNFC instances managed by the VNFM.

The notification is mandatory.

9.3.2.2 Trigger conditions

- An alarm has been created.
- An alarm has been updated, e.g. if the severity of the alarm has changed.

9.3.2.3 Attributes

The AlarmNotification shall follow the indications provided in table 9.3.2.3-1.

Table 9.3.2.3-1: Attributes of the AlarmNotification

Attribute	Qualifier	Cardinality	Content	Description
alarm	M	1	Alarm	Information about an alarm including AlarmId, affected VNF identifier, affected VNFC identifiers, and FaultDetails. For notifications related to changes in the state of virtualised resources (indicated using the attribute faultType), the alarm shall indicate: • the cause for the state change of the virtualised resource using the attribute probableCause, with possible values such as: maintenance of NFVI component, evacuation of NFVI component, etc. • the identifier of the origin (VIM) responsible for the management of the virtualised resource with state change using the attribute faultDetails.

9.3.3 AlarmClearedNotification

9.3.3.1 Description

This notification informs the receiver of the clearing of an alarm related to the VNFs and their constituent VNFCs managed by the VNFM, e.g. the alarm's perceived severity is set to "cleared" since the corresponding fault has been solved.

The notification is mandatory.

9.3.3.2 Trigger conditions

- An alarm has been cleared.

9.3.3.3 Attributes

The AlarmClearedNotification shall follow the indications provided in table 9.3.3.3-1.

Table 9.3.3.3-1: Attributes of the AlarmClearedNotification

Attribute	Qualifier	Cardinality	Content	Description
alarmId	M	1	Identifier (Reference to Alarm)	Alarm identifier.
alarmClearedTime	M	1	DateTime	Timestamp indicating when the alarm was cleared.

9.3.4 Alarm information element

9.3.4.1 Description

The Alarm information element encapsulates information about an alarm.

9.3.4.2 Attributes

The Alarm information element shall follow the indications provided in table 9.3.4.2-1.

Table 9.3.4.2-1: Attributes of the Alarm information element

Attribute	Qualifier	Cardinality	Content	Description
alarmId	M	1	Identifier	Identifier of this Alarm information element.
managedObjectId	M	1	Identifier	Identifier of the affected managed object. The managed objects for this information element will be VNF instances.
vnfcId	M	1..N	Identifier	Identifier of the affected VNFC(s).
rootCauseFaultyResource	M	1	FaultyResourceInfo	The virtualised resources that are causing the VNF/VNFC fault.
alarmRaisedTime	M	1	DateTime	Timestamp indicating when the alarm is raised by the managed object.
alarmChangedTime	M	0..1	DateTime	Timestamp indicating when the alarm was last changed. It shall be present if the alarm has been updated.
alarmClearedTime	M	0..1	DateTime	Timestamp indicating when the alarm was cleared. It shall be present if the alarm has been cleared.
ackState	M	1	Enum	State of the alarm, permitted values include: • Acknowledged • Unacknowledged
perceivedSeverity	M	1	Enum	Perceived severity of the managed object failure. Permitted values include: • Critical • Major • Minor • Warning • Indeterminate • Cleared
eventTime	M	1	DateTime	Timestamp indicating when the fault was observed.
eventType	M	1	Enum	Type of the event. The allowed values for the eventType attribute use the event type defined in [5]: • Communication Alarm • Processing Alarm • Environment Alarm • QoS Alarm • Equipment Alarm
faultType	M	0..1	String	Additional information related to the type of the fault.
probableCause	M	1	String	Information about the probable cause of the fault
isRootCause	M	1	Boolean	Attribute indicating if this fault is the root for other correlated alarms. If TRUE, then the alarms listed in the Attribute CorrelatedAlarmId are caused by this fault.
correlatedAlarmId	M	0..N	Identifier (Reference to Alarm)	List of identifiers of other alarms correlated to this fault.
faultDetails	M	0..N	Not specified	Provides additional information about the fault.

9.3.5 FaultyResourceInfo information element

9.3.5.1 Description

The FaultyResourceInfo information element encapsulates information about faulty resource that has a negative impact on a VNF.

9.3.5.2 Attributes

The FaultyResourceInfo information element shall follow the indications provided in table 9.3.5.2-1.

Table 9.3.5.2-1: Attributes of the FaultyResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
faultyResource	M	1	ResourceHandle	Information that identifies the faulty resource instance and its managing entity. See clause 9.4.7.
faultyResourceType	M	1	Enum	Type of the faulty resource. Values: • COMPUTE. • STORAGE. • NETWORK.

9.3.6 AlarmListRebuiltNotification

9.3.6.1 Description

This notification informs the receiver that the active alarm list has been rebuilt by the VNFM. Upon receipt of this notification, the receiver need to use the "Get Alarm List" operation to synchronize its view on current active alarms with that of the VNFM.

The notification is mandatory.

9.3.6.2 Trigger conditions

- Active alarm list has been rebuilt by the VNFM, e.g. if the VNFM detects its storage holding the alarm list is corrupted.

9.3.6.3 Attributes

The AlarmListRebuiltNotification does not contain any attributes.

9.4 Information elements related to VNF Lifecycle Management

9.4.1 Introduction

This clause defines information elements related to VNF Lifecycle Management.

9.4.2 VnfInfo information element

9.4.2.1 Description

The VnfInfo information element provides run-time information about a VNF instance.

NOTE: In ETSI GS NFV-MAN 001 [i.3], the concept of the VNF record (VNFR) was introduced which is a model for the totality of information managed by the VNFM regarding a running VNF instance. VNFR is not used in the present document.

9.4.2.2 Attributes

The VnfInfo information element shall follow the indications provided in table 9.4.2.2-1.

Table 9.4.2.2-1: Attributes of the VnfInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance that is represented by this VnfInfo information element.
vnfInstanceName	M	0..1	String	VNF instance name. See note 1.
vnfInstanceDescription	M	0..1	String	Human-readable description of the VNF instance. See note 1.
vnfdId	M	1	Identifier	Identifier of the VNFD on which the VNF instance is based. See note 2 and note 3.
vnfProvider	M	1	String	See note 3.
vnfProductName	M	1	String	See note 3.
vnfSoftwareVersion	M	1	Version	See note 3.
vnfdVersion	M	1	Version	See note 3.
onboardedVnfPkgInfoId	M	1	Identifier	Identifier of information held by the NFVO about the specific VNF Package on which the VNF is based. This identifier was allocated by the NFVO. See notes 1 and 4.
vnfConfigurableProperty	M	0..N	KeyValuePair	Current values of the configurable properties of the VNF instance. Configurable properties referred in this attribute are declared in the VNFD (see clause 7.1.12 in ETSI GS NFV-IFA 011 [4]). They include those set as initial configuration, and/or those that modify a running configuration. See note 1 and note 5.
instantiationState	M	1	Enum	The instantiation state of the VNF instance. Possible values: NOT_INSTANTIATED (VNF instance is terminated or not instantiated, and the identifier of the VNF instance exists), INSTANTIATED (VNF instance is instantiated).
instantiatedVnfInfo	M	0..1	InstantiatedVnfInfo	Information specific to an instantiated VNF instance. Shall be present if the VNF is in INSTANTIATED instantiation state.
metadata	M	0..N	KeyValuePair	Additional VNF-specific metadata describing the VNF instance. Metadata that are writeable are declared in the VNFD (see clause 7.1.14.2 in ETSI GS NFV-IFA 011 [4]).See note 1.
extension	M	0..N	KeyValuePair	VNF-specific attributes that affect the lifecycle management of this VNF instance by the VNFM, or the lifecycle management scripts. Extensions that are writeable are declared in the VNFD (see clause 7.1.14.2 in ETSI GS NFV-IFA 011 [4]). See note 1.

NOTE 1: This attribute in the VnfInfo shall be writable through the Modify VNF information operation (refer to clause 7.2.12).

NOTE 2: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way.

NOTE 3: See ETSI GS NFV-IFA 011 [4], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package which was used to instantiate the VNF instance.

NOTE 4: Modifying the value of this attribute can be performed when no conflicts exist between the previous and the newly referred VNF Package, e.g. when the new VNFD is not changed with respect to the previous VNFD apart from referencing to other VNF software image(s). In order to avoid misalignment of the VnfInfo with the current VNF's on-boarded VNF Package, the values copied from the VNFD of the on-boarded VNF Package (see note 3) need to be kept in sync.

NOTE 5: VNF configurable properties are sometimes also referred to as configuration parameters applicable to a VNF. Some of these are set prior to instantiation and cannot be modified if the VNF is instantiated, some are set prior to instantiation (are part of initial configuration) and can be modified later, and others can be set only after instantiation. The applicability of certain configuration may depend on the VNF and the required operation of the VNF at a certain point in time.

9.4.3 InstantiatedVnfInfo information element

9.4.3.1 Description

This information element provides run-time information specific to an instantiated VNF instance.

Annex A provides examples illustrating the relationship among the different run-time information elements (CP, VL and link ports) used to represent the connectivity of a VNF.

9.4.3.2 Attributes

The InstantiatedVnfInfo information element shall follow the indications provided in table 9.4.3.2-1.

Table 9.4.3.2-1: Attributes of the InstantiatedVnfInfo information element

Attribute	Qualifier	Cardinality	Content	Description
flavourId	M	1	Identifier (Reference to VnfDf)	Identifier of the VNF DF applied to this VNF instance. See note 1.
vnfState	M	1	Enum	The state of the VNF instance. Permitted values include: STARTED, STOPPED.
scaleStatus	M	0..N	ScaleInfo	Scale status of the VNF, one entry per aspect. Shall be present if the VNF supports scaling. Represents for every scaling aspect how "big" the VNF has been scaled w.r.t. that aspect. See note 2.
extCpInfo	M	1..N	VnfExtCpInfo	External CPs exposed by the VNF instance.
extVirtualLinkInfo	M	0..N	ExtVirtualLinkInfo	External VLs the VNF instance is connected to.
extManagedVirtualLinkInfo	M	0..N	ExtManagedVirtualLinkInfo	Externally-managed internal VLs of the VNF instance.
monitoringParameter	M	0..N	Not specified	Performance metrics tracked by VNFM (e.g. for auto-scaling purposes) and their current (as known to the VNFM) values. See note 3.
localizationLanguage	M	0..1	Not specified	Information about localization language of the VNF (includes e.g. strings in the VNFD). The localization languages supported by a VNF can be declared in the VNFD, and localization language selection can take place at instantiation time.
vnfcResourceInfo	M	0..N	VnfcResourceInfo	Information on the virtualised compute and storage resource(s) used by the VNFCs of the VNF instance.
vnfVirtualLinkResourceInfo	M	0..N	VnfVirtualLinkResourceInfo	Information on the virtualised network resource(s) used by the VLs of the VNF instance.
virtualStorageResourceInfo	M	0..N	VirtualStorageResourceInfo	Information on the virtualised storage resource(s) used as storage for the VNF instance.
vnfcInfo	M	0..N	VnfcInfo	The information items about the selected VNFC instance(s).

NOTE 1: The VnfDf information element is defined in ETSI GS NFV-IFA 011 [4], clause 7.1.8.2.
NOTE 2: For every scaling aspect, together with the information provided by the "maxScaleLevel" attribute of the "ScalingAspect" information element in the VNFD, this allows an external entity to derive how many scaling steps are possible for scaling in or scaling out a VNF instance. Per aspect, the number of steps possible to scale in corresponds to the "scaleLevel" attribute for that aspect in the "scaleStatus" information element, and the possible number of steps to scale out corresponds to the difference between "maxScaleLevel" for that aspect, and the "scaleLevel" attribute for that aspect in the "scaleStatus" information element.
NOTE 3: The monitoring parameters to be tracked by VNFM are identified by VNF provider in the VNFD. The VNFM collects the values of identified performance metrics using one or more locally initiated PM Jobs.

9.4.4 VnfcResourceInfo information element

9.4.4.1 Description

This information element provides information on virtualised compute and storage resources used by a VNFC in a VNF instance.

9.4.4.2 Attributes

The VnfcResourceInfo information element shall follow the indications provided in table 9.4.4.2-1.

Table 9.4.4.2-1: Attributes of the VnfcResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfcInstanceId	M	1	Identifier	Identifier of this VNFC instance.
vduId	M	1	Identifier (Reference to Vdu)	Reference to the applicable Vdu information element in the VNFD.
computeResource	M	1	ResourceHandle	Reference to the VirtualCompute resource.
storageResource	M	0..N	Identifier (Reference to VirtualStorageResourceInfo)	Reference(s) to the VirtualStorage resource(s).
vnfcCpInfo	M	1..N	VnfcCpInfo	CP(s) of the VNFC instance.
metadata	M	0..N	KeyValuePair	Metadata about this resource.

9.4.5 VnfVirtualLinkResourceInfo information element

9.4.5.1 Description

This information element provides information on virtualised network resources used by an internal VL instance in a VNF.

9.4.5.2 Attributes

The VnfVirtualLinkResourceInfo information element shall follow the indications provided in table 9.4.5.2-1.

Table 9.4.5.2-1: Attributes of the VnfVirtualLinkResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
virtualLinkInstanceId	M	1	Identifier	Identifier of this VL instance.
virtualLinkDescId	M	1	Identifier (Reference to VnfVirtualLinkDesc)	Identifier of the VNF Virtual Link Descriptor (VLD) in the VNFD.
networkResource	M	1	ResourceHandle	Reference to the VirtualNetwork resource.
vnfLinkPort	M	0..N	VnfLinkPort	Links ports of this VL.
metadata	M	0..N	KeyValuePair	Metadata about this resource.

9.4.6 VirtualStorageResourceInfo information element

9.4.6.1 Description

This information element provides information on virtualised storage resources used by a storage instance in a VNF.

9.4.6.2 Attributes

The VirtualStorageResourceInfo information element shall follow the indications provided in table 9.4.6.2-1.

Table 9.4.6.2-1: Attributes of the VirtualStorageResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
virtualStorageInstanceId	M	1	Identifier	Identifier of this virtual storage resource instance.
virtualStorageDescId	M	1	Identifier (Reference to VirtualStorageDesc)	Identifier of the VirtualStorageDesc in the VNFD.
storageResource	M	1	ResourceHandle	Reference to the VirtualStorage resource(s).
metadata	M	0..N	KeyValuePair	Metadata about this resource.

9.4.7 ResourceHandle information element

9.4.7.1 Description

This information element provides information that allows addressing a resource that is used by a VNF instance.

Information about the resource is available to the VNFM from the corresponding Virtualised Compute/Storage/Network Resource Management interfaces. Table 9.4.7.1-1 shows the relationship between the resourceId attribute of ResourceHandle specified in the present document and the resource identifiers used in the aforementioned interfaces specified in ETSI GS NFV-IFA 006 [1].

Table 9.4.7.1-1: Relationship between resource identifiers

Attribute in Ve-Vnfm-em	Type, Interface, information element and attribute in ETSI GS NFV-IFA 006 [1]		
	Type	Interface	Information element and attribute
ResourceHandle:resourceId	Compute	Virtualised Compute Resource Management	- VirtualCompute:computeId
	Storage	Virtualised Storage Resource Management	- VirtualStorage:storageId
	Network	Virtualised Network Resource Management	- VirtualNetwork:networkResourceId

9.4.7.2 Attributes

The ResourceHandle information element shall follow the indications provided in table 9.4.7.2-1.

Table 9.4.7.2-1: Attributes of the ResourceHandle information element

Attribute	Qualifier	Cardinality	Content	Description
vimConnectionId	CM	0..1	Identifier (Reference to VimConnectionInfo)	Reference to the identifier of the VimConnectionInfo information element defining the VIM connection to manage this resource. This attribute shall be supported when VNF-related Resource Management in direct mode is applicable. See note 1.
resourceProviderId	CM	0..1	Identifier	Identifies the entity responsible for the management of the virtualised resource. This attribute shall be supported when VNF-related Resource Management in indirect mode is applicable.
resourceId	M	1	Identifier	Identifier of the resource in the scope of the VIM or the resource provider.
vimLevelResourceType	M	0..1	Not specified	Type of the resource in the scope of the VIM or the resource provider. See note 2.

NOTE 1: The VimConnectionInfo is a data structure known to the VNFM. The identifier of the VIM connection provides scope to the resourceId.
NOTE 2: The value set of the "vimLevelResourceType" attribute is within the scope of the VIM or the resource provider and can be used as information that complements the ResourceHandle.

9.4.8 ScaleInfo information element

9.4.8.1 Description

This information element provides information about the scale level of a VNF instance w.r.t. one scaling aspect.

9.4.8.2 Attributes

The ScaleInfo information element shall follow the indications provided in table 9.4.8.2-1.

Table 9.4.8.2-1: Attributes of the ScaleInfo information element

Attribute	Qualifier	Cardinality	Content	Description
aspectId	M	1	Identifier (Reference to ScalingAspect)	Reference to the scaling aspect.
scaleLevel	M	1	Integer	The scale level for that aspect. Minimum value 0, maximum value maxScaleLevel as declared in the VNFD (see ETSI GS NFV-IFA 011 [4], clause 7.1.10).

9.4.9 ExtVirtualLinkInfo information element

9.4.9.1 Description

This information element provides a reference to an external VL.

9.4.9.2 Attributes

The ExtVirtualLinkInfo information element shall follow the indications provided in table 9.4.9.2-1.

Table 9.4.9.2-1: Attributes of the ExtVirtualLinkInfo information element

Attribute	Qualifier	Cardinality	Content	Description
extVirtualLinkId	M	1	Identifier	Identifier of this external VL.
resourceHandle	M	1	ResourceHandle	Reference to the resource realizing this VL.
linkPort	M	0..N	ExtLinkPort	Link ports of this VL.

9.4.10 ExtManagedVirtualLinkInfo information element

9.4.10.1 Description

This information element provides a reference to an externally-managed internal VL.

9.4.10.2 Attributes

The ExtManagedVirtualLinkInfo information element shall follow the indications provided in table 9.4.10.2-1.

Table 9.4.10.2-1: Attributes of the ExtManagedVirtualLinkInfo information element

Attribute	Qualifier	Cardinality	Content	Description
extManagedVirtualLinkId	M	1	Identifier	Identifier of this externally-managed internal VL.
vnfVirtualLinkDescId	M	1	Identifier (Reference to VnfVirtualLinkDesc)	Identifier of the VNF Virtual Link Descriptor (VLD) in the VNFD.
networkResource	M	1	ResourceHandle	Reference to the VirtualNetwork resource.
linkPort	M	0..N	VnfLinkPort	Link ports of this VL.

9.4.11 VnfLinkPort information element

9.4.11.1 Description

This information element provides information about a port of a VNF's internal VL. See also VnfVirtualLinkResourceInfo in clause 9.4.5.

9.4.11.2 Attributes

The attributes of the VnfLinkPort information element shall follow the indications provided in table 9.4.11.2-1.

Table 9.4.11.2-1: Attributes of the VnfLinkPort information element

Attribute	Qualifier	Cardinality	Content	Description
vnfLinkPortId	M	1	Identifier	Identifier of this link port as provided by the entity that has created the link port.
resourceHandle	M	1	ResourceHandle	Reference to the virtualised network resource realizing this link port.
cpInstanceId	M	0..1	Identifier (Reference to VnfExtCpInfo or VnfcCpInfo))	When the link port is used for external connectivity by the VNF, the external CP of the VNF to be connected to this link port. When the link port is used for internal connectivity in the VNF, the VNFC CP to be connected to this link port. See note.
NOTE: There shall be at most one link port associated with any external connection point instance or internal connection point (i.e. VNFC CP) instance.				

9.4.12 ExtManagedVirtualLinkData information element

9.4.12.1 Description

This information element provides the information of an externally-managed internal VL to be used as a parameter passed to VNF lifecycle management operations.

9.4.12.2 Attributes

The ExtManagedVirtualLinkData information element shall follow the indications provided in table 9.4.12.2-1.

Table 9.4.12.2-1: Attributes of the ExtManagedVirtualLinkData information element

Attribute	Qualifier	Cardinality	Content	Description
extManagedVirtualLinkId	M	1	Identifier	Identifier of this externally-managed internal VL instance.
virtualLinkDescId	M	1	Identifier (Reference to VnfVirtualLinkDesc)	Identifier of the VLD in the VNFD for this VL.
vimConnectionId	CM	0..1	Identifier (Reference to VimConnectionInfo)	Identifier of the VIM connection to manage this resource. This attribute shall be supported and present if VNF-related resource management in direct mode is applicable. See note.
resourceProviderId	CM	0..1	Identifier	Identifies the entity responsible for the management of the resource. This attribute shall be supported and present when VNF-related Resource Management in indirect mode is applicable.
resourceId	M	1	Identifier	Identifier of the resource in the scope of the VIM or the resource provider.
NOTE: The VimConnectionInfo is a data structure known to the VNFM. The identifier of the VIM connection provides scope to the resourceId.				

9.4.13 VnfcInfo information element

9.4.13.1 Description

This information element provides information about a VNFC instance.

9.4.13.2 Attributes

The VnfcInfo information element shall follow the indications provided in table 9.4.13.2-1.

Table 9.4.13.2-1: Attributes of the VnfcInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfcInstanceId	M	1	Identifier	Identifier of the VNFC instance that is represented by this VnfcInfo information element.
vduId	M	1	Identifier (Reference to Vdu)	Identifier of the VDU in the VNFD.
vnfcState	M	1	Enum	The state of the VNFC instance. Permitted values include: STARTED, STOPPED.
vnfcConfigurableProperty	M	0..N	KeyValuePair	Current values of the configurable properties of the VNFC instance. Configurable properties referred in this attribute are declared in the VNFD (see clause 7.1.6.7 in ETSI GS NFV-IFA 011 [4]). See note.
NOTE: This attribute in the VnfInfo shall be writable through the Modify VNF information operation (refer to clause 7.2.12).				

9.4.14 ExtLinkPort information element

9.4.14.1 Description

This information element provides information about a port of an external VL, i.e. a port providing connectivity for the VNF to an NS VL.

9.4.14.2 Attributes

The attributes of the ExtLinkPort information element shall follow the indications provided in table 9.4.14.2-1.

Table 9.4.14.2-1: Attributes of the ExtLinkPort information element

Attribute	Qualifier	Cardinality	Content	Description
extLinkPortId	M	1	Identifier	Identifier of this link port as provided by the entity that has created the link port.
resourceHandle	M	1	ResourceHandle	Reference to the virtualised network resource realizing this link port.
cpInstanceId	M	0..1	Identifier (Reference to VnfExtCpInfo)	External CP of the VNF to be connected to this link port. See note.
NOTE: There shall be at most one link port associated with any external connection point instance.				

9.4.15 VnfcCpInfo information element

9.4.15.1 Description

This information element provides information related to a CP of a VNFC.

9.4.15.2 Attributes

The VnfcCpInfo information element shall follow the indications provided in table 9.4.15.2-1.

Table 9.4.15.2-1: Attributes of the VnfcCpInfo information element

Attribute	Qualifier	Cardinality	Content	Description
cpInstanceId	M	1	Identifier	Identifier of this VnfcCpInfo information element.
cpdId	M	1	Identifier (Reference to VduCpd)	Identifier of the VDU CPD, cpdId, in the VNFD.
vnfExtCpId	M	0..1	Identifier (Reference to VnfExtCpInfo)	When the VNFC CP is exposed as external CP of the VNF, the identifier of this external VNF CP.
address	M	0..N	Not specified.	List of network addresses that have been configured (statically or dynamically) on the CP.

9.5 Information elements and notifications related to VNF Lifecycle Changes

9.5.1 Introduction

This clause defines notifications related to VNF lifecycle changes and update of VNF information.

9.5.2 VnfLcmOperationOccurrenceNotification

9.5.2.1 Description

This notification informs the receiver of changes in the VNF lifecycle caused by VNF lifecycle management operation occurrences. The support of the notification is mandatory.

9.5.2.2 Trigger conditions

This notification is produced when there is a change in the VNF lifecycle caused by VNF lifecycle management operation occurrences, including:

- Instantiation of the VNF.
- Scaling of the VNF instance (including auto-scaling).
- Healing of the VNF instance (including auto-healing).
- Change of the state of the VNF instance (i.e. Operate VNF).
- Change of the DF of the VNF instance.
- Changing the external virtual links of the VNF instance.
- Termination of the VNF instance.
- Modification of VNF instance information and/or VNF/VNFC configurable properties explicitly through Modify VNF Information operation.

If this is a notification about the start of an LCM operation occurrence, the notification shall be sent before any action (including sending the grant request) is taken, however, after acknowledging the LCM operation request to the consumer.

If this is a notification about the result of an LCM operation occurrence, the notification shall be sent after all other actions of the LCM operation have been executed.

9.5.2.3 Attributes

The VnfLcmOperationOccurrenceNotification shall follow the indications provided in table 9.5.2.3-1.

Table 9.5.2.3-1: Attributes of the VnfLcmOperationOccurrenceNotification

Attribute	Qualifier	Cardinality	Content	Description
status	M	1	Enum	Indicates whether this notification reports about the start of a lifecycle management operation occurrence or the result of a lifecycle management operation occurrence.
vnfInstanceId	M	1	Identifier	The identifier of the VNF instance affected.
operation	M	1	String	The lifecycle management operation.
isAutomaticInvocation	M	1	Boolean	Set to true if this VNF LCM operation occurrence has been triggered by an automated procedure inside the VNFM (i.e. ScaleVnf / ScaleVnfToLevel triggered by auto-scale, or HealVnf triggered by auto-heal). Set to false otherwise.
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the VNF lifecycle management operation occurrence associated to the notification.

Attribute	Qualifier	Cardinality	Content	Description
affectedVnfc	M	0..N	AffectedVnfc	Information about VNFC instances that were affected during the execution of the lifecycle management operation, if this notification represents the result of a lifecycle management operation occurrence.
affectedVirtualLink	M	0..N	AffectedVirtualLink	Information about VL instances that were affected during the execution of the lifecycle management operation, if this notification represents the result of a lifecycle management operation occurrence.
affectedVirtualStorage	M	0..N	AffectedVirtualStorage	Information about virtualised storage instances that were affected during the execution of the lifecycle management operation, if this notification represents the result of a lifecycle management operation occurrence.
changedExtConnectivity	M	0..N	ExtVirtualLinkInfo	Information about changed external connectivity, if this notification represents the result of a lifecycle management operation occurrence. Only relevant for the "Change external VNF connectivity" operation.
changedInfo	M	0..1	Not specified	Information about the changed VNF information, including changed VNF configurable properties, if this notification represents the result of a lifecycle management operation occurrence.
NOTE: If this notification represents the result of a lifecycle management operation occurrence that was not successful, the notification shall contain appropriate error information.				

9.5.3 AffectedVnfc information element

9.5.3.1 Description

This information element provides information about added, deleted, modified and temporary VNFCs.

9.5.3.2 Attributes

The AffectedVnfc information element shall follow the indications provided in table 9.5.3.2-1.

Table 9.5.3.2-1: Attributes of the AffectedVnfc information element

Attribute	Qualifier	Cardinality	Content	Description
vnfcInstanceId	M	1	Identifier (Reference to VnfcResourceInfor)	Identifier of the VNFC instance.
vduId	M	1	Identifier (Reference to Vdu)	Identifier of the VDU in the VNFD.
changeType	M	1	Enum	Signals the type of change (added, removed, modified, temporary). For a temporary resource, an AffectedVnfc IE exists as long as the temporary resource exists.
computeResource	M	0..1	ResourceHandle	Reference to the VirtualCompute resource.
addedStorageResourceIds	M	0..N	Identifier	Reference(s) to VirtualStorage resource(s) that were added. Each value refers to a VirtualStorageResourceInfo item in the VnfInfo that was added to the VNFC. It shall be provided if at least one storage resource was added to the VNFC.
removedStorageResourceIds	M	0..N	Identifier	Reference(s) to VirtualStorage resource(s) that were removed. The value contains the identifier of a VirtualStorageResourceInfo item that has been removed from the VNFC, and might no longer exist in the VnfInfo. It shall be provided if at least one storage resource was removed from the VNFC.

9.5.4 AffectedVirtualLink information element

9.5.4.1 Description

This information element provides information about added, deleted, modified and temporary VLs, as well as about link port changes.

9.5.4.2 Attributes

The AffectedVirtualLink information element shall follow the indications provided in table 9.5.4.2-1.

Table 9.5.4.2-1: Attributes of the AffectedVirtualLink information element

Attribute	Qualifier	Cardinality	Content	Description
virtualLinkInstanceId	M	1	Identifier (Reference to VirtualLinkResourceInfo)	Identifier of the VL instance.
virtualLinkDescId	M	1	Identifier (Reference to VnfVirtualLinkDesc)	Identifier of the VLD in the VNFD.
changeType	M	1	Enum	Signals the type of change including, not limited to, changes made to the characteristics of the existing VL, new VL added, existing VL removed, temporary VL exists, link port added, link port removed. For a temporary resource, an AffectedVirtualLink IE exists as long as the temporary resource exists.
networkResource	M	1	ResourceHandle	Reference to the VirtualNetwork resource.

9.5.5 AffectedVirtualStorage information element

9.5.5.1 Description

This information element provides information about added, deleted, modified and temporary virtual storage resources.

9.5.5.2 Attributes

The AffectedVirtualStorage information element shall follow the indications provided in table 9.5.5.2-1.

Table 9.5.5.2-1: Attributes of the AffectedVirtualStorage information element

Attribute	Qualifier	Cardinality	Content	Description
virtualStorageInstanceId	M	1	Identifier (Reference to VirtualStorageResourceInfo)	Identifier of the virtual storage instance.
virtualStorageDescId	M	1	Identifier (Reference to VirtualStorageDesc)	Identifier of the VirtualStorageDesc in the VNFD.
changeType	M	1	Enum	Signals the type of change (added, removed, modified, temporary). For a temporary resource, an AffectedVirtualStorage IE exists as long as the temporary resource exists.
storageResource	M	1	ResourceHandle	Reference to the VirtualStorage resource.

9.5.6 Void

9.5.7 VnfIdentifierCreationNotification

9.5.7.1 Description

This notification informs the receiver of the creation of a new VNF instance identifier and the associated instance of a VnfInfo information element, identified by that identifier. The support of the notification is mandatory.

9.5.7.2 Trigger conditions

- Creation of a VNF instance identifier and the associated instance of a VnfInfo information element.

9.5.7.3 Attributes

The VnfIdentifierCreationNotification shall follow the indications provided in table 9.5.7.3-1.

Table 9.5.7.3-1: Attributes of the VnfIdentifierCreationNotification

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	The newly created VNF instance identifier

9.5.8 VnfIdentifierDeletionNotification

9.5.8.1 Description

This notification informs the receiver of the deletion of a VNF instance identifier and the associated instance of a VnfInfo information element identified by that identifier. The support of the notification is mandatory.

9.5.8.2 Trigger conditions

- Deletion of a VNF instance identifier and the associated instance of a VnfInfo information element.

9.5.8.3 Attributes

The VnfIdentifierDeletionNotification shall follow the indications provided in table 9.5.8.3-1.

Table 9.5.8.3-1: Attributes of the VnfIdentifierDeletionNotification

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	The VNF instance identifier that has been deleted.

9.6 Information elements and notifications related to VNF indicators

9.6.1 Introduction

The clauses below define information elements which represent indicator values, and notifications about changes of these.

9.6.2 IndicatorValueChangeNotification

9.6.2.1 Description

This notification informs the receiver of a value change of an indicator related to the VNF.

The notification is mandatory.

9.6.2.2 Trigger conditions

- The value of an indicator has changed.

9.6.2.3 Attributes

The IndicatorValueChangeNotification information element shall follow the indications provided in table 9.6.2.3-1.

Table 9.6.2.3-1: Attributes of the IndicatorValueChangeNotification

Attribute	Qualifier	Cardinality	Content	Description
indicatorInformation	M	1	IndicatorInformation	This is to provide the indicator, the value of the indicator and the VNF instance the indicator is related to.

9.6.3 IndicatorInformation information element

9.6.3.1 Description

This information element provides the indicator values of a VNF instance.

9.6.3.2 Attributes

The IndicatorInformation information element shall follow the indications provided in table 9.6.3.2-1.

Table 9.6.3.2-1: Attributes of the IndicatorInformation information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifies the VNF instance which provides the indicator value(s).
indicatorId	M	1	Identifier (Reference to VnfIndicator)	Identifier of the indicator.
indicatorValue	M	1	Value	Provides the value of the indicator. The value format is described in the VNFD (see ETSI GS NFV-IFA 011 [4]).
indicatorName	M	0..1	String	Human readable name of the indicator. Shall be present if defined in the VNFD according to clause 7.1.2 of ETSI GS NFV-IFA 011 [4].

9.7 Information elements and notifications related to VNF Performance Management

9.7.1 Introduction

This clause defines information elements and notifications related to VNF Performance management.

9.7.2 ObjectSelection information element

9.7.2.1 Description

This information element allows specifying VNF and/or VNFC instances on which performance information will be provided.

The ObjectSelection is a pattern to select object instances. The pattern is used in multiple interfaces.

In the present interface, the ObjectSelection pattern is used to select VNF and/or VNFC instances.

The pattern proposes 2 exclusive options:

1) Provide a list of object types and a filter to specify object properties.

2) Provide a list of object instances.

In the present interface, the object type will be VNF (represented by VNFD), and the filter will be based on some VNF and/or VNFC properties.

9.7.2.2 Attributes

The ObjectSelection information element shall follow the indications provided in table 9.7.2.2-1.

Table 9.7.2.2-1: Attributes of the ObjectSelection information element

Attribute	Qualifier	Cardinality	Content	Description
objectType	M	0..N	String	Defines the object types. The object types for this information element will be the VNFDs. One of the two attributes (objectType + objectFilter or objectInstanceId) shall be present.
objectFilter	M	0..1	Filter	The filter will apply on the object types to specify on which object instances the performance information is requested to be collected. One of the two attributes (objectType + objectFilter or objectInstanceId) shall be present.
objectInstanceId	M	0..N	Identifier	Identifies the object instances for which performance information is requested to be collected. The object instances for this information element will be either VNF or VNFC instances. One of the two attributes (objectType+ objectFilter or objectInstanceId) shall be present.

9.7.3 PmJob information element

9.7.3.1 Description

This information element provides the details of the PM Job.

The object instances for this information element will be VNF or VNFC instances.

9.7.3.2 Attributes

The PmJob information element shall follow the indications provided in table 9.7.3.2-1.

Table 9.7.3.2-1: Attributes of the PmJob information element

Attribute	Qualifier	Cardinality	Content	Description
pmJobId	M	1	Identifier	Identifier of this PM Job.
objectSelector	M	1	ObjectSelection	Defines the object instances for which performance information is requested to be collected. The object instances for this information element will be VNF and/or VNFC instances.
performanceMetric	M	0..N	String	This defines the type(s) of performance metric(s) for the specified object instances. At least one of the two attributes (performance metric or group) shall be present.
performanceMetricGroup	M	0..N	String	Group of performance metrics. A metric group is a pre-defined list of metrics, known to the producer that it can decompose to individual metrics. At least one of the two attributes (performance metric or group) shall be present.

Attribute	Qualifier	Cardinality	Content	Description
collectionPeriod	M	1	Enum	Specifies the periodicity at which the producer will collect performance information. See note.
reportingPeriod	M	1	Enum	Specifies the periodicity at which the producer will report to the consumer about performance information. See note.
reportingBoundary	O	0..1	Not specified	Identifies a boundary after which the reporting will stop. The boundary shall allow a single reporting as well as periodic reporting up to the boundary.
NOTE: At the end of each reportingPeriod, the producer will inform the consumer about availability of the performance data collected for each completed collection period during this reportingPeriod. While the exact definition of the types for collectionPeriod and reportingPeriod is left for further specification, it is recommended that the reportingPeriod be equal or a multiple of the collectionPeriod. In the latter case, the performance data for the collection periods within one reporting period would be reported together.				

9.7.4 Threshold information element

9.7.4.1 Description

This information element provides the details of a threshold.

The object instances for this information element will be VNF or VNFC instances.

9.7.4.2 Attributes

The Threshold information element shall follow the indications provided in table 9.7.4.2-1.

Table 9.7.4.2-1: Attributes of the Threshold information element

Attribute	Qualifier	Cardinality	Content	Description
thresholdId	M	1	Identifier	Identifier of this Threshold information element.
objectSelector	M	1	ObjectSelection	Defines the object instances associated with the threshold. The object instances for this information element will be VNF and/or VNFC instances.
performanceMetric	M	1	String	Defines the performance metric associated with the threshold.
thresholdType	M	1	Enum	Type of threshold. The list of possible values is left for the protocol design stage and might include: single/ multi valued threshold, static/dynamic threshold, template based threshold, etc.
thresholdDetails	M	1	Not specified	Details of the threshold: value to be crossed, details on the notification to be generated, etc.

9.7.5 PerformanceReport information element

9.7.5.1 Description

This information element defines the format of a performance report provided by the producer to the consumer on a specified object instance or a set of them.

The object instances for this information element will be VNF or VNFC instances.

9.7.5.2 Attributes

The PerformanceReport information element shall follow the indications provided in table 9.7.5.2-1.

Table 9.7.5.2-1: Attributes of the PerformanceReport information element

Attribute	Qualifier	Cardinality	Content	Description
performanceReport	M	1..N	PerformanceReportEntry	List of performance information entries.

9.7.6 PerformanceReportEntry information element

9.7.6.1 Description

This information element defines a single performance report entry. This performance report entry is for a given metric of a given object instance, but can include multiple collected values.

The object instances for this information element will be VNF or VNFC instances.

9.7.6.2 Attributes

The PerformanceReportEntry information element shall follow the indications provided in table 9.7.6.2-1.

Table 9.7.6.2-1: Attributes of the PerformanceReportEntry information element

Attribute	Qualifier	Cardinality	Content	Description
objectType	M	1	String	Defines the object type. The object types for this information element will be the VNFDs.
objectInstanceId	M	1	Identifier	The object instance for which the performance metric is reported. The object instances for this information element will be VNF or VNFC instances.
performanceMetric	M	1	String	Name of the metric collected.
performanceValue	M	1..N	PerformanceValueEntry	List of performance values with associated timestamp.

9.7.7 PerformanceValueEntry information element

9.7.7.1 Description

This information element defines a single performance value with its associated time stamp.

9.7.7.2 Attributes

The PerformanceValueEntry information element shall follow the indications provided in table 9.7.7.2-1.

Table 9.7.7.2-1: Attributes of the PerformanceValueEntry information element

Attribute	Qualifier	Cardinality	Content	Description
timeStamp	M	1	DateTime	Timestamp indicating when the data was collected.
performanceValue	M	1	Value	Value of the metric collected.

9.7.8 PerformanceInformationAvailableNotification

9.7.8.1 Description

This notification informs the receiver that performance information is available. Delivery mechanism for the performance reports is left for later specification.

The object instances for this information element will be VNF or VNFC instances.

9.7.8.2 Trigger Conditions

- New performance information is available.

9.7.8.3 Attributes

The PerformanceInformationAvailableNotification shall follow the indications provided in table 9.7.8.3-1.

Table 9.7.8.3-1: Attributes of the PerformanceInformationAvailableNotification

Attribute	Qualifier	Cardinality	Content	Description
objectInstanceId	M	1..N	Identifier	Object instance(s) for which performance information is available. The object instances for this information element will be VNF or VNFC instances.

9.7.9 ThresholdCrossedNotification

9.7.9.1 Description

This notification informs the receiver that a threshold value has been crossed.

The object instances for this information element will be VNF or VNFC instances.

9.7.9.2 Trigger Condition

A Threshold has been crossed. Depending on threshold type, there might be a single or multiple crossing values.

9.7.9.3 Attributes

The ThresholdCrossedNotification shall follow the indications provided in table 9.7.9.3-1.

Table 9.7.9.3-1: Attributes of the ThresholdCrossedNotification

Attribute	Qualifier	Cardinality	Content	Description
thresholdId	M	1	Identifier (Reference to Threshold)	Threshold which has been crossed.
crossingDirection	M	1	Enum	An indication of whether the threshold was crossed in upward or downward direction. Permitted values: UP, DOWN.
objectInstanceId	M	1	Identifier	Object instance for which the threshold has been crossed. The object instances for this information element will be VNF or VNFC instances.
performanceMetric	M	1	String	Performance metric associated with the threshold.
performanceValue	M	1	Value	Value of the metric that resulted in threshold crossing.

9.8 Information elements and notifications related to multiple interfaces

9.8.1 Introduction

This clause defines information elements that are referenced by other information elements related to multiple interfaces.

9.8.2 VnfExtCpInfo information element

9.8.2.1 Description

This information element provides information related to an external CP.

9.8.2.2 Attributes

The VnfExtCpInfo information element shall follow the indications provided in table 9.8.2.2-1.

Table 9.8.2.2-1: Attributes of the VnfExtCpInfo information element

Attribute	Qualifier	Cardinality	Content	Description
cpInstanceId	M	1	Identifier	Identifier of this external CP instance and of this VnfExtCpInfo information element.
cpdId	M	1	Identifier (Reference to VnfExtCpd)	Identifier of the external Connection Point Descriptor (CPD), VnfExtCpd, in the VNFD.
address	M	0..N	Not specified	List of network addresses that have been configured (statically or dynamically) on the CP.

9.8.3 ExtVirtualLinkData information element

9.8.3.1 Description

This information element provides the information of an external VL to be used as a parameter passed to multiple interfaces.

9.8.3.2 Attributes

The ExtVirtualLinkData information element shall follow the indications provided in table 9.8.3.2-1.

Table 9.8.3.2-1: Attributes of the ExtVirtualLinkData information element

Attribute	Qualifier	Cardinality	Content	Description
extVirtualLinkId	M	1	Identifier	Identifier of this external VL instance.
vimConnectionId	CM	0..1	Identifier (Reference to VimConnectionInfo)	Identifier of the VIM connection to manage this resource. This attribute shall be supported and present if VNF-related resource management in direct mode is applicable. See note.
resourceProviderId	CM	0..1	Identifier	Identifies the entity responsible for the management of the resource. This attribute shall be supported and present when VNF-related Resource Management in indirect mode is applicable.
resourceId	M	1	Identifier	Identifier of the resource in the scope of the VIM or the resource provider.
extCp	M	1..N	VnfExtCpData	External CPs of the VNF to be connected to this external VL.
NOTE: The VimConnectionInfo is a data structure known to the VNFM. The identifier of the VIM connection provides scope to the resourceId.				

9.8.4 VnfExtCpData information element

9.8.4.1 Description

This information element provides input information related to an external CP.

9.8.4.2 Attributes

The VnfExtCpData information element shall follow the indications provided in table 9.8.4.2-1.

Table 9.8.4.2-1: Attributes of the VnfExtCpData information element

Attribute	Qualifier	Cardinality	Content	Description
cpdId	M	1	Identifier	Identifier of the connection point descriptor (CPD) in the VNFD.
fixedAddresses	M	0..N	Not specified	List of (fixed) network addresses that need to be configured on the CP. It shall be provided for configuring fixed addresses.
dynamicAddresses	M	0..N	Not specified	List of parameter sets for the assignment of dynamic addresses. It should be possible to define per parameter set the number of network addresses to be assigned dynamically. Other parameters could be, e.g. valid address ranges or subnets. It shall be provided if dynamic addresses need to be configured on the CP.

9.8.5 Void

Annex A (informative): Examples of VNF connectivity patterns

A.1 Introduction

This annex illustrates examples of possible connectivity patterns for a VNF. The purpose is to illustrate the relationship among the different information elements specified in clause 9.4 that are used to describe the connectivity of and within a VNF instance.This annex also illustrates the use of the "Change External VNF Connectivity" to re-connect external CPs of a VNF instance to a different external VL.

NOTE: The information related to connectivity as shown in the Annex A is to be understood in the context of the present document, i.e. availability of certain information on the Ve-Vnfm reference point follows the conditions that are detailed in the respective attribute descriptions and notes in the present document.

A.2 Example of a VNF with two different types of external connection points

The present example shows a regular connectivity pattern of a VNF where the two external CPs of the VNF use different connectivity patterns. Figure A.2-1 illustrates the example, from which it is highlighted the following:

- An external CP of the VNF instance (see VnfExtCp #1) that maps to an internal CP, i.e. a CP of a specific VNFC.
- An external CP of the VNF instance (see VnfExtCp #2) that refers to a link port of an internal VL of the VNF (see VnfLinkPort #2.2).
- An internal VL of the VNF instance (see VnfVirtualLink #1) that is only used for connectivity of VNFCs within the VNF.
- An internal VL of the VNF instance (see VnfVirtualLink #2) that is used as provider of a link port for connectivity of external CPs of the VNF.
- Link ports of internal VL(s) of the VNF instance (see VnfLinkPort #1.1 to #1.3 and VnfLinkPort #2.1), which are exposed on Ve-Vnfm reference point.
- Internal CPs, i.e. CPs of specific VNFCs (see VNFC CPs), which are exposed on the Ve-Vnfm reference point.

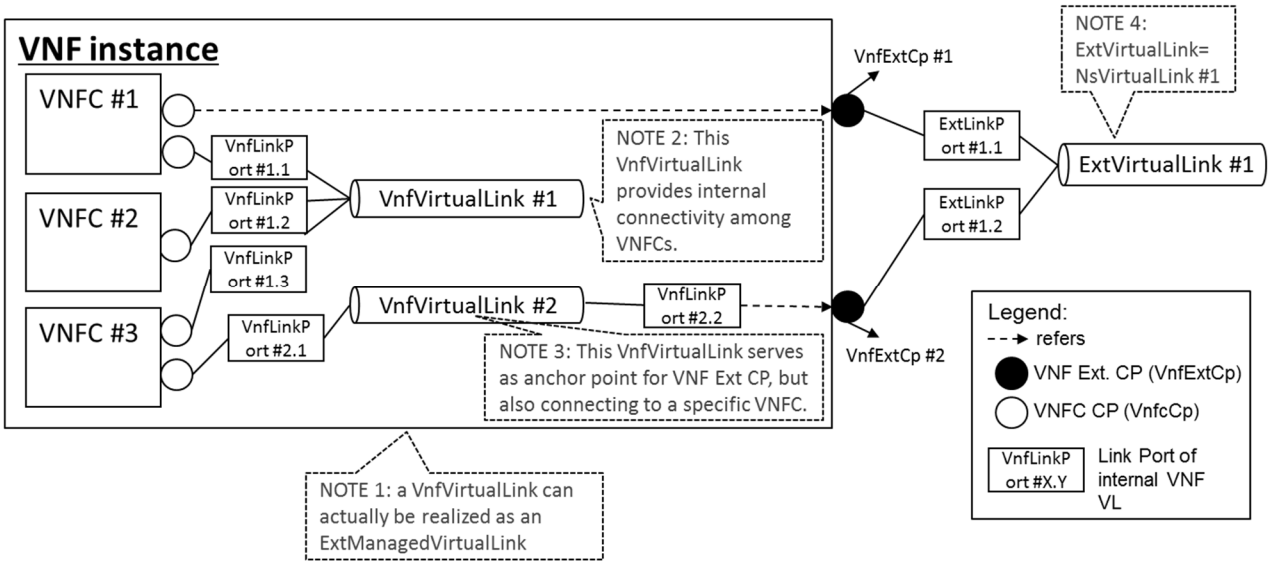

Figure A.2-1: Example of a VNF with two different types of external connections points

A.3 Example of changing VNF connectivity

This example illustrates the operation "Change external VNF connectivity" (clause 7.2.18). The scenario depicted disconnects all external CP instances that were created based on a particular CPD from a "source" external VL and connects them to a "target" external VL.

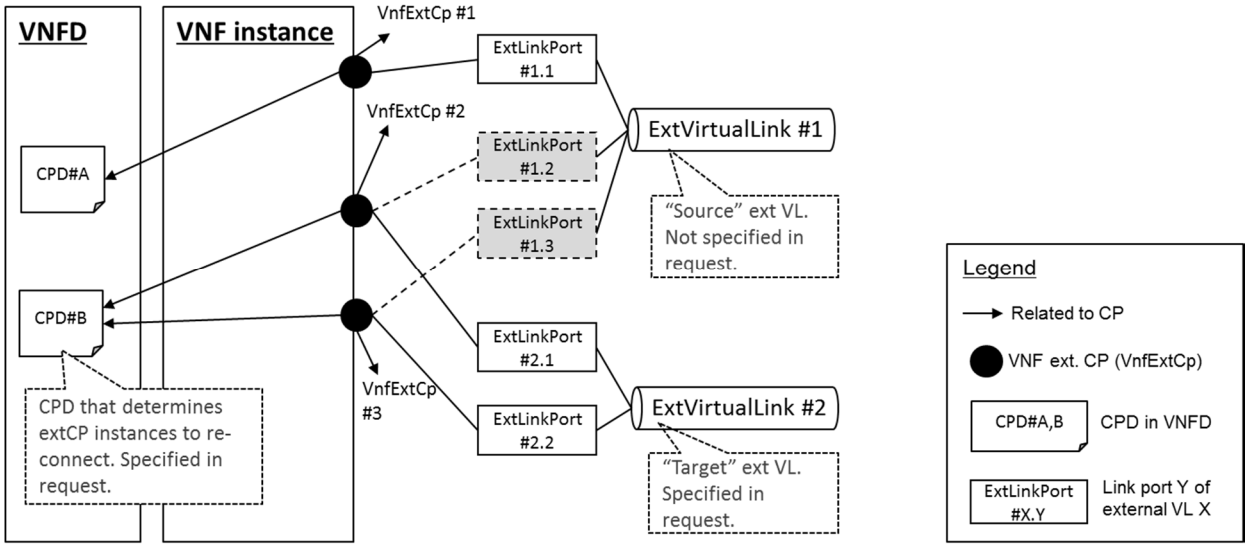

Figure A.3-1: Illustration of disconnecting external CPs from one external VL and connecting them to another external VL

Annex B (informative): Example VNF Configuration flows

B.1 Explicit change of VNF Configurable Properties

Figure B.1-1 illustrates two alternative non-exhaustive examples of VNF configuration triggered by explicit change of VNF Configurable Properties. The first alternative (steps 10a - 11a) shows a scenario where VNF Configuration interface is being used, the second alternative (steps 10b - 12b) shows a scenario where VNF Configuration interface is not being used and VNF is capable of self-managing. Other alternatives for passing the configuration changes to the VNF instance are possible.

```
@startuml
skinparam monochrome true
skinparam sequenceActorBackgroundColor #FFFFFF
skinparam sequenceParticipantBackgroundColor #FFFFFF
skinparam noteBackgroundColor #FFFFFF

participant "NFVO" as nfvo
participant "VNFM" as vnfm
participant "Instance:VNF" as vnf

autonumber "#'.'"

    ref over nfvo: triggering of VNF instantiation
        nfvo ->> vnfm: InstantiateVNF()
    activate vnfm
        vnfm ->> nfvo: VnfLcmOperationOccurrenceNotification(START, lcmOpOccId, ...)
        create vnf
        vnfm -->> vnf: Create VNF instance
        vnfm ->> vnf: Subscribe(VNF Indicator Value Change Notifications)
        vnfm ->> nfvo: VnfLcmOperationOccurrenceNotification(RESULT, lcmOpOccId, changes)
    deactivate vnfm
    vnf ->> vnfm: Subscribe(VNF LCM Notifications)
    ref over nfvo, vnfm, vnf: "normal" operation...
    nfvo ->> vnfm: ModifyVnfInfo(vnfConfigurableProperties)
    activate vnfm
        vnfm ->> nfvo: VnfLcmOperationOccurrenceNotification(start, lcmOpOccId, ...)
        vnfm ->> vnf: VnfLcmOperationOccurrenceNotification(start, lcmOpOccId, ...)
        alt VNF configuration interface used
            autonumber 10 "#'a.'"
            vnfm ->> vnf: SetConfigurationRequest(configuration data)
            ref over vnf: Apply config
            vnf -->> vnfm: SetConfigurationResponse(configuration data)
        else VNF configuration interface is not used and VNF is self-managing
            autonumber 10 "#'b.'"
            vnf -> vnfm: GetOperationStatusRequest(lcmOpOccId)
            vnfm --> vnf: GetOperationStatusResponse(operation parameters)
            ref over vnf: Self-manage
            vnf ->> vnfm: IndicatorValueChangeNotification(self-manage, (lcmOpOccId, result))
        end
        autonumber 13 "#'.'"
        ref over vnfm: Update VnfInfo IE
        vnfm ->> vnf: VnfLcmOperationOccurrenceNotification(RESULT, lcmOpOccId, ...)
        vnfm ->> nfvo: VnfLcmOperationOccurrenceNotification(RESULT, lcmOpOccId, ...)
    deactivate vnfm
@enduml
```

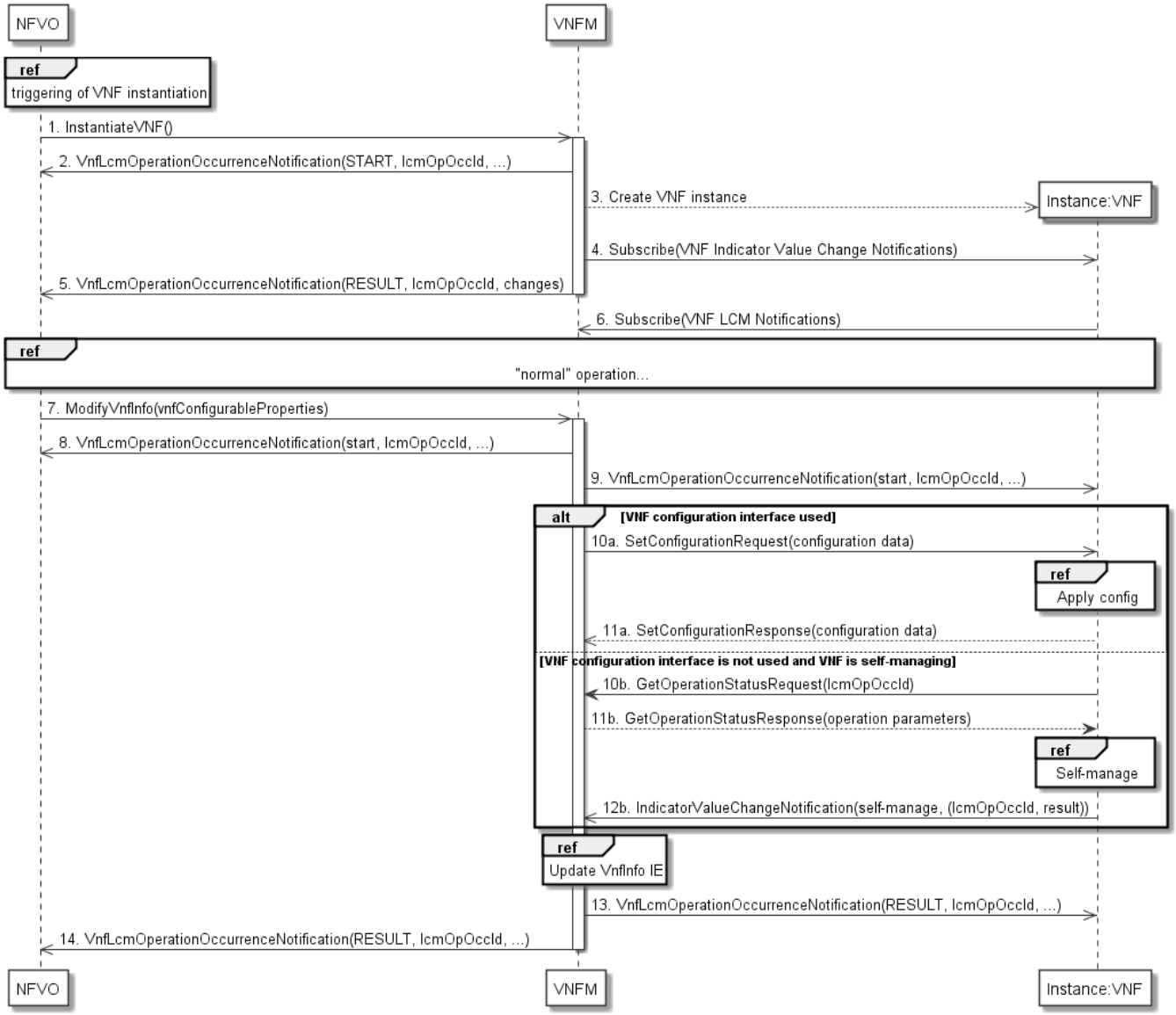

Figure B.1-1: Explicit change of VNF Configurable Properties

1. NFVO requests VNF instantiation (the trigger for VNF instantiation is out of scope of this flow). The step of VNF identifier creation is omitted for simplicity.

2. VNFM begins the VNF instantiation and sends the `VnfLcmOperationOccurrenceNotification` to NFVO (and all subscribers) indicating the start of LCM operation, operation ID, etc.

3. VNFM creates the new VNF instance (the interactions between VNFM and VIM are omitted for simplicity).

4. VNFM subscribes for VNF indicator value change notifications.

5. VNFM completes the VNF instantiation and sends the `VnfLcmOperationOccurrenceNotification` to NFVO (and all subscribers) indicating the result of LCM operation, operation ID, etc.

6. VNF instance subscribes for VNF LCM notifications.

7. NFVO requests VNFM to change certain VNF Configurable Properties with the `ModifyVnfInfo` operation and by passing `vnfConfigurableProperties` as a parameter.

8. 9. VNFM begins the operation and sends the `VnfLcmOperationOccurrenceNotification` to NFVO, VNF (and all subscribers) indicating the start of LCM operation, operation ID, etc.

If VNF Configuration interface (as defined in clause 6.2) is used:

10a. VNFM requests setting the configuration with `SetConfigurationRequest` operation and passes configuration data to be applied as a parameter.

VNF applies configuration:

11a. VNF returns the result of setting the configuration with the `SetConfigurationResponse` operation and passes the applied configuration data as a parameter.

If VNF Configuration interface (as defined in clause 6.2) is not used and VNF is self-managing:

10b. VNF requests the LCM operation details with the `GetOperationStatusRequest` operation and passes the `lcmOpOccId` as a parameter.

11b. VNFM returns the LCM operation details (including its parameters) with the `GetOperationStatusResponse` operation and passes the "LCM operation parameters" as a parameter.

VNF "self-manages" (determines what configuration changes need to be applied and applies them):

12b. VNF sends VNF Indicator value change notification `IndicatorValueChangeNotification` with information about successful completion of self-managing and indication that configuration changes have been applied. The format and values of the VNF indicator are declared by the VNF provider in the VNFD.

VNFM updates the VnfInfo IE accordingly:

13. 14. VNFM completes the operation and sends the `VnfLcmOperationOccurrenceNotification` to NFVO, VNF (and all subscribers) indicating the result of LCM operation, operation ID, etc.

Annex C (informative): Authors & contributors

The following people have contributed to the present document:

Rapporteur:

- Mr. shitao li, Huawei

Other contributors:

- Jong-Hwa Yi, ETRI
- Jihyun Lee, ETRI
- Joan Triay, DOCOMO Communications Lab.
- Kazuaki Obana, DOCOMO Communications Lab.
- Ashiq Khan, DOCOMO Communications Lab.
- Myung-Ki Shin, ETRI
- Zou Lan, Huawei
- Kai ZHANG, Huawei
- Bruno, Chatras, Orange
- Gerald Kunzmann, DOCOMO Communications Lab.
- Marc Flauw, Hewlett-Packard Enterprise
- Anatoly Andrianov, Nokia Networks
- Zarrar Yousaf, NEC Europe Ltd
- Dan Druta , AT&T
- Shitao Li, Huawei
- Uwe Rauschenbach, Nokia Networks
- Aijuan Feng, Huawei
- Yu Fang, Huawei
- Bertrand Souville, DOCOMO Communications Lab.
- Janusz Pieczerak, Orange
- Gyula Bodog, Nokia Networks
- Tommy Lindgren, Ericsson
- Nicola Santinelli, TELECOM ITALIA S.p.A.
- Zhou Yan, Huawei
- Chu Junsheng, ZTE Corporation
- Chen Liping, ZTE Corporation
- Xie Yunpeng, China Telecommunications Corporation

- Amanda Xiang, Huawei
- Junyi Jiang, Huawei
- Haitao Xia, Huawei
- Peng Zhao, China Mobile
- Harshad Tanna, Ericsson
- Chirag Parekh, Ericsson
- Jeremy Fuller, GENBAND Ireland Ltd
- Mehmet Ersue, Nokia Networks
- Haibin Chu, Ericsson LM
- Ernie Bayha, Ericsson LM

Annex D (informative): Change History

Date	Version	Information about changes
December	V0.0.1	Skeleton and scope
March 2015	V0.2.0	Implemented NFVIFA(15)000357r1
May 2015	V0.2.1	Implemented contribution agreed in Sanya
May 2015	V0.2.2	Implemented contribution agreed in San Jose and the Disclaimer Implemented CR: NFVIFA(15)000799r5, NFVIFA(15)000615r6, NFVIFA(15)000622r4, NFVIFA(15)000623r4, NFVIFA(15)000523r2, NFVIFA(15)000067r3
August 2015	V0.2.3	Implemented NFVIFA(15)000563r4, NFVIFA(15)000930r3
September 2015	V0.3.0	Implemented NFVIFA(15)0001141r1, NFVIFA(15)000978r2, NFVIFA(15)000929r6
October 2015	V0.3.1	Implemented NFVIFA(15)000994r1, NFVIFA(15)000953r4
November 2015	V0.4.0	Implemented NFVIFA(15)0001151r3, NFVIFA(15)00065r3, NFVIFA(15)0001150r2, NFVIFA(15)0001152r2, NFVIFA(15)0001139r3, NFVIFA(15)0001196r1, NFVIFA(15)0001317r1
December 2015	V0.5.0	Implemented NFVIFA(15)0001529, NFVIFA(15)0001495r2, NFVIFA(15)0001561, NFVIFA(15)0001509, NFVIFA(15)0001493r2, NFVIFA(15)0001323r6, NFVIFA(15)0001498r1, NFVIFA(15)0001455r2, NFVIFA(15)0001596, NFVIFA(15)0001608r2, NFVIFA(15)0001444
February 2016	V0.6.0	Implemented NFVIFA(15)0001321r4, NFVIFA(16)000098r1, NFVIFA(16)000107r1, NFVIFA(16)000108r1, NFVIFA(16)000109r1, NFVIFA(16)000110r1, NFVIFA(16)000112r1, NFVIFA(16)000113r1, NFVIFA(16)000114r2, NFVIFA(16)000115r2, NFVIFA(16)000116r1, NFVIFA(16)000120r3, NFVIFA(16)000127r4, NFVIFA(16)000106r1, NFVIFA(16)000119r1
March 2016	V0.7.0	Implemented NFVIFA(16)000257r1, NFVIFA(15)000779r9, NFVIFA(16)000281r1, NFVIFA(16)000253r1, NFVIFA(16)000264r1, NFVIFA(16)000322r1, NFVIFA(16)000252r1, NFVIFA(16)000192, NFVIFA(16)000326, NFVIFA(16)000324r1, NFVIFA(16)000289, NFVIFA(16)000291r1, NFVIFA(16)000290r1, NFVIFA(16)000210r2, NFVIFA(16)000246r2, NFVIFA(16)000249r3, NFVIFA(16)000224, NFVIFA(16)000197, NFVIFA(16)000217, NFVIFA(16)000267r4, NFVIFA(16)000142r6
March 2016	V0.7.1	ToC Updated
April 2016	V0.8.0	Implemented NFVIFA(16)000394r2, NFVIFA(16)000429r2, NFVIFA(16)000430, NFVIFA(16)000469r1, NFVIFA(16)000485, NFVIFA(16)000508r1, NFVIFA(16)000526r1, NFVIFA(16)000530, NFVIFA(16)000552, NFVIFA(16)000562, NFVIFA(16)000571, NFVIFA(16)000558r1, NFVIFA(16)000421r2, NFVIFA(16)000425r2 NFVIFA(16)000382r2, NFVIFA(16)000397r3, NFVIFA(16)000418r2, NFVIFA(16)000422, NFVIFA(16)000444r2, NFVIFA(16)000497r2, NFVIFA(16)000499r1, NFVIFA(16)000543r2, NFVIFA(16)000549r1, NFVIFA(16)000550r1, NFVIFA(16)000554, NFVIFA(16)000556r1, NFVIFA(16)000557r1, NFVIFA(16)000560r3, NFVIFA(16)000563, NFVIFA(16)000575r1, NFVIFA(16)000576r3 NFVIFA(16)000420r2, NFVIFA(16)000445r1, NFVIFA(16)000449r3, NFVIFA(16)000450r3, NFVIFA(16)000462, NFVIFA(16)000466r5, NFVIFA(16)000509r1, NFVIFA(16)000547r1, NFVIFA(16)000551r4, NFVIFA(16)000553, NFVIFA(16)000565r2, NFVIFA(16)000592, NFVIFA(16)000601r4, NFVIFA(16)000652, NFVIFA(16)000656r1, NFVIFA(16)000667r1, NFVIFA(16)000722 NFVIFA(16)000461r7, NFVIFA(16)000720r2, NFVIFA(16)000484r9

Date	Version	Information about changes
June 2016	V0.9.0	Implemented NFVIFA(16)000453r2, NFVIFA(16)000769r2, NFVIFA(16)000783, NFVIFA(16)000784, NFVIFA(16)000785r3, NFVIFA(16)000786r1, NFVIFA(16)000788, NFVIFA(16)000794r1, NFVIFA(16)000831, NFVIFA(16)000836r2, NFVIFA(16)000837, NFVIFA(16)000838r2, NFVIFA(16)000839, NFVIFA(16)000852 NFVIFA(16)000791r5, NFVIFA(16)000795r3, NFVIFA(16)000856, NFVIFA(16)000893r1, NFVIFA(16)0001001r3, NFVIFA(16)0001011r2, NFVIFA(16)0001016 NFVIFA(16)000857r2, NFVIFA(16)000858, NFVIFA(16)000860, NFVIFA(16)000862r2, NFVIFA(16)000864, NFVIFA(16)000905r1
July 2016	V0.9.1	Implemented NFVIFA(16)0001042r2, NFVIFA(16)0001049, NFVIFA(16)0001033r1, NFVIFA(16)0001038r1, NFVIFA(16)000806r11, NFVIFA(16)0001066r1, NFVIFA(16)0001063r1 Fix conventions for stage 3 data type and the use of abbreviation
August 2016	V0.9.2	Implemented NFVIFA(16)0001088r1, NFVIFA(16)0001078r2, NFVIFA(16)0001077 ToC alignment with IFA007: • move VNF Lifecycle Change Notification interface from clauses 7.5 to 7.3 • move VNF Performance Management interface from clauses 7.6 to 7.4 • move VNF Fault Management interface from clauses 7.3 to 7.5 • move VNF Configuration Management interface from clauses 7.4 to 7.6 Delete editor's notes in clauses 4.2 and 5.3.1 as proposed in NFVIFA(16)0001122 Implemented NFVIFA(16)0001109r2 Alignment of the sequence and description of the LCM operations between IFA007 & IFA008 in clause 7.2 Alignment of the sequence and description of the Information elements related to VNF Lifecycle Management between IFA007 & IFA008 in clause 9.4 Fixed some typos and editorial inconsistencies
August 2016	V0.9.3	Implemented NFVIFA(16)0001131, NFVIFA(16)0001126r3
August 2016	V0.9.4	Implemented NFVIFA(16)000770_Replace_primitive_type_TimeStamp_by_DateTime Extended the implementation of change 5 in NFVIFA(16)001126r3_IFA007_IFA008_inconsistency_fixes to all places where the text is applicable (Description of input/output parameters of type ExtVirtualLink and ExtManagedVirtualLink starts with "Information about", rather than "Reference to")
22 August 2016	V0.9.4b	Alignment of the Operation Result clauses: result of an operation use past tense and return parameter use passive present tense and avoid future tense (will be). Output parameter mentioned in attribute descriptions are also changed to use "is returned" or "are returned" Editorial change in Table 9.4.2.2-1, change "NVFO" to "NFVO"
7 September 2016	V0.9.5	Implemented editorial CR in NFVIFA(16)0001219
February 2017	V2.1.2	Implemented NFVIFA(17)000032
March 2017	V2.1.3	Implemented NFVIFA(17)000062r4
May 2017	V2.1.4	Implemented NFVIFA(17)000095r3, NFVIFA(17)000238r1, NFVIFA(17)000257r2 after NFVIFA#48-F2F-Piscataway NFVIFA(17)000285r4, NFVIFA(17)000293r2, NFVIFA(17)000354, NFVIFA(17)000463r1, NFVIFA(17)000478, NFVIFA(17)000479 after NFV#18
June 2017	V2.1.5	Implemented NFVIFA(17)000104r2, NFVIFA(17)000298r1, NFVIFA(17)000391r2, NFVIFA(17)000428r2, NFVIFA(17)000451r2, NFVIFA(17)000506r1, NFVIFA(17)000548r3, NFVIFA(17)000558r1, NFVIFA(17)000564r1, NFVIFA(17)000567r3, NFVIFA(17)000568r1, NFVIFA(17)000589r3, NFVIFA(17)000601r1, NFVIFA(17)000275r1, NFVIFA(17)000274r2

History

Document history		
V2.1.1	October 2016	Publication
V2.3.1	August 2017	Publication