ETSI GR NFV-TST 007 V2.6.1 (2020-01)

Network Functions Virtualisation (NFV) Release 2; Testing; Guidelines on Interoperability Testing for MANO

Disclaimer

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

Reference

RGR/NFV-TST007ed261

Keywords

interoperability, management, MANO, NFV, testing

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 7

Foreword 7

Modal verbs terminology 7

1 Scope 8

2 References 8
2.1 Normative references 8
2.2 Informative references 8

3 Definition of terms, symbols and abbreviations 9
3.1 Terms 9
3.2 Symbols 9
3.3 Abbreviations 9

4 Interoperability Features Statement (IFS) 10
4.1 Introduction 10
4.2 IFS for VIM 10
4.2.1 Software Image Management 10
4.2.2 VNF Package Management 11
4.2.3 VNF Lifecycle Management 11
4.2.4 Fault Management 12
4.2.5 Performance Management 13
4.2.6 NS Lifecycle Management 14
4.3 IFS for VNFM 16
4.3.1 Software Image Management 16
4.3.2 VNF Package Management 16
4.3.3 VNF Lifecycle Management 17
4.3.4 Fault Management 19
4.3.5 Performance Management 20
4.3.6 NS Lifecycle Management 21
4.4 IFS for NFVO 22
4.4.1 Software Image Management 22
4.4.2 VNF Package Management 23
4.4.3 VNF Lifecycle Management 23
4.4.4 Fault Management 24
4.4.5 Performance Management 25
4.4.6 NS Lifecycle Management 26
4.5 IFS for EM/VNF 28
4.5.1 Software Image Management 28
4.5.2 VNF Package Management 29
4.5.3 VNF Lifecycle Management 29
4.5.4 Fault Management 29
4.5.5 Performance Management 29
4.5.6 NS Lifecycle Management 29

5 System Under Test (SUT) 30
5.1 SUT Configuration 1 30
5.2 SUT Configuration 2 30
5.3 SUT Configuration 3 31
5.4 SUT Configuration 4 31

6 Test Suite Structure 32
6.1 Introduction 32
6.2 Software Image Management Test Cases Overview 32
6.3 VNF Package Management Test Cases Overview 32
6.4 VNF Lifecycle Management Test Cases Overview 33
6.5 Fault Management Test Cases Overview 33
6.6 Performance Management Test Cases Overview 34

6.7 NS Lifecycle Management Test Cases Overview 34

7 Test Descriptions 36
7.1 Introduction 36
7.2 Software Image Management 37
7.2.1 Add Software Image 37
7.2.2 Query Software Image 37
7.2.2.1 Query Software Image by NFVO 37
7.2.2.2 Query Software Image by VNFM 38
7.2.3 Update Software Image 38
7.2.4 Delete Software Image 39
7.3 VNF Package Management 39
7.3.1 On-board VNF Package 39
7.3.2 Delete VNF Package 40
7.3.3 Abort VNF Package Delete Operation 40
7.4 VNF Lifecycle Management 41
7.4.1 Instantiate VNF with an EM Request 41
7.4.2 Query VNF with an EM Request 42
7.4.3 Modify VNF Configuration Information with an EM Request 42
7.4.4 Start VNF/VNFC with an EM Request 43
7.4.5 Stop VNF/VNFC with an EM Request 44
7.4.6 VNF Healing with an EM/VNF Request 45
7.4.7 VNF Scale Out with an EM/VNF Request 46
7.4.8 VNF Scale In with an EM/VNF Request 47
7.4.9 Terminate VNF with an EM Request 48
7.5 Fault Management 49
7.5.1 Virtualised Resource Fault Management 49
7.5.1.1 Virtualised Resource Fault Alarm Notification 49
7.5.1.2 Virtualised Resource Fault Alarm Clearance Notification 50
7.5.2 VNF Fault Management 50
7.5.2.1 VNF Fault Alarm Notifications 50
7.5.2.2 VNF Fault Alarm Clearance Notifications 51
7.5.3 NS Fault Management 52
7.5.3.1 NS Fault Alarm Subscription Creation 52
7.5.3.2 NS Fault Alarm Subscription Deletion 52
7.5.3.3 NS Fault Alarm Notification 53
7.5.3.4 NS Fault Alarm Notification Clearance 54
7.5.3.5 NS Fault Alarm Query 54
7.6 Performance Management 55
7.6.1 Virtualised Resource Performance Management 55
7.6.1.1 Virtualised Resource PM Job Creation and Notification Monitoring 55
7.6.1.2 Virtualised Resource PM Job Creation and Threshold Monitoring 56
7.6.1.3 Virtualised Resource PM Job Deletion 56
7.6.1.4 Virtualised Resource PM Threshold Deletion 57
7.6.2 VNF Performance Management 58
7.6.2.1 VNF PM Job Creation and Notification Monitoring 58
7.6.2.2 VNF PM Job Creation and Threshold Monitoring 59
7.6.2.3 VNF PM Job Deletion 60
7.6.2.4 VNF PM Threshold Deletion 61
7.6.3 NS Performance Management 62
7.6.3.1 NS PM Monitoring Job Creation 62
7.6.3.2 NS PM Performance Metrics Query 62
7.6.3.3 NS PM Threshold Creation 63
7.6.3.4 NS PM Subscription Creation for Threshold Information 64
7.6.3.5 NS PM Subscription Creation for Performance Information 64
7.6.3.6 NS PM Threshold Notification 65
7.6.3.7 NS PM Monitoring Information Notification 66
7.6.3.8 NS PM Subscription Deletion 66
7.6.3.9 NS PM Monitoring Job Deletion 67
7.6.3.10 NS PM Performance Metrics Threshold Deletion 68
7.7 NS Lifecycle Management 68
7.7.1 NS Instantiation 68

7.7.1.1 Standalone NS Instantiation ... 68
7.7.1.2 Nested NS Instantiation ... 70
7.7.1.3 Multi-Site NS Instantiation ... 71
7.7.1.4 SFC NS Instantiation ... 73
7.7.2 NS Scaling ... 74
7.7.2.1 NS Scale Out ... 74
7.7.2.1.1 NS Scale out with an Operator Action ... 74
7.7.2.1.2 NS Scale out with a VNF Indicator notification ... 75
7.7.2.1.3 NS Scale out with a VIM KPI ... 76
7.7.2.1.4 NS Scale out with a query to VNF Indicator by VNFM ... 78
7.7.2.1.5 NS Scale out for Multi-Site with an Operator Action ... 79
7.7.2.2 NS Scale In ... 80
7.7.2.2.1 NS Scale In with an Operator Action ... 80
7.7.2.2.2 NS Scale in with a VNF Indicator notification ... 81
7.7.2.2.3 NS Scale in with a VIM KPI ... 82
7.7.2.2.4 NS Scale in with a query to VNF Indicator by VNFM ... 83
7.7.2.2.5 NS Scale in for Multi-Site with an Operator Action ... 85
7.7.2.3 NS VNF Scale Out ... 86
7.7.2.3.1 NS VNF Scale Out with an Operator Action ... 86
7.7.2.3.2 NS VNF Scale Out with a VNF Indicator notification ... 87
7.7.2.3.3 NS VNF Scale Out with a VIM KPI ... 88
7.7.2.3.4 NS VNF Scale Out with a query to VNF Indicator by VNFM ... 89
7.7.2.3.5 NS VNF Scale Out for Multi-Site with an Operator Action ... 90
7.7.2.4 NS VNF Scale In ... 91
7.7.2.4.1 NS VNF Scale In with an Operator Action ... 91
7.7.2.4.2 NS VNF Scale In with a VNF Indicator notification ... 92
7.7.2.4.3 NS VNF Scale In with a VIM KPI ... 93
7.7.2.4.4 NS VNF Scale In with a query to VNF indicator by VNFM ... 95
7.7.2.4.5 NS VNF Scale In for Multi-Site with an Operator Action ... 96
7.7.2.5 NS Scale to Level ... 97
7.7.2.5.1 NS Scale to Level with an Operator Action ... 97
7.7.2.5.2 NS Scale to Level from VNF Indicator notification ... 98
7.7.2.5.3 NS Scale to Level from a VIM KPI ... 99
7.7.2.5.4 NS Scale to Level with an Operator Action for Multi-Site ... 101
7.7.2.6 NS VNF Scale to Level ... 102
7.7.2.6.1 NS VNF Scale to Level with an Operator Action ... 102
7.7.2.6.2 NS VNF Scale to Level from VNF Indicator notification ... 103
7.7.2.6.3 NS VNF Scale to Level with a VIM KPI ... 105
7.7.2.6.4 NS VNF Scale to Level with an Operator Action for Multi-Site ... 106
7.7.3 NS Update ... 107
7.7.3.1 Start VNF Instance ... 107
7.7.3.2 Stop VNF Instance ... 108
7.7.3.3 Instantiate VNF and Add Instance to NS Instance ... 109
7.7.3.4 Remove VNF Instances from a NS Instance ... 110
7.7.3.5 Add Shared VNF Instances to NS Instance ... 110
7.7.3.6 Remove Shared VNF Instances from NS Instance ... 111
7.7.3.7 Change VNF Deployment Flavour ... 112
7.7.3.8 VNF Configuration update on runtime ... 113
7.7.4 NS Healing ... 114
7.7.4.1 Partial NS Healing with an Operator Action ... 114
7.7.4.2 Complete NS Healing with an Operator Action ... 115
7.7.5 NS Termination ... 116
7.7.5.1 Standalone NS Termination ... 116
7.7.5.2 Nested NS Termination ... 116
7.7.5.3 Multi-Site NS Termination ... 117
7.7.5.4 SFC NS Termination ... 118

Annex A: Technical Report Card Sample ... 119

Annex B: Document Usage Process Diagram ... 120

Annex C: Interoperability Features Statement (IFS) Diagram ... 121

C.1 Introduction 121
C.2 IFS for VIM 121
C.3 IFS for VNFM 122
C.4 IFS for NFVO 123
C.5 IFS for EM/VNF 124
History 125

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables 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 Report (GR) has been produced by ETSI Industry Specification Group (ISG) Network Functions Virtualisation (NFV).

Modal verbs terminology

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

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

1 Scope

The present document describes a set of informative interoperability test guidelines for NFV capabilities that require interactions between the components implementing NFV functionality, namely, the NFVO, VNFM, EM-VNF and VIM-NFVI (Functions under Test).

The guidelines in the present document follow the interoperability testing methodology described in ETSI GS NFV-TST 002 [i.1] and are implementation agnostic. The capabilities in scope are enabled by the interfaces on the main NFV reference points between the Functions Under Test:

- Or-Vi;
- Or-Vnfm;
- Vi-Vnfm; and
- Ve-Vnfm.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

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

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

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

[i.1] ETSI GS NFV-TST 002 (V1.1.1): "Network Functions Virtualisation (NFV); Testing Methodology; Report on NFV Interoperability Testing Methodology".

[i.2] ETSI GS NFV 003 (V1.2.1): "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

[i.3] ETSI GS NFV-IFA 005 (V2.3.1): "Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Or-Vi reference point - Interface and Information Model Specification".

[i.4] ETSI GS NFV-IFA 006 (V2.3.1): "Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Vi-Vnfm reference point - Interface and Information Model Specification".

[i.5] ETSI GS NFV-IFA 007 (V2.3.1): "Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Or-Vnfm reference point - Interface and Information Model Specification".

[i.6] ETSI GS NFV-IFA 008 (V2.3.1): "Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Ve-Vnfm reference point - Interface and Information Model Specification".

[i.7] ETSI GS NFV-IFA 010 (V2.3.1): "Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Functional requirements specification".

[i.8] ETSI GS NFV-IFA 011 (V2.3.1): "Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; VNF Packaging Specification".

[i.9] ETSI GS NFV-IFA 013 (V2.3.1): "Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Os-Ma-Nfvo reference point - Interface and Information Model Specification".

3 Definition of terms, symbols and abbreviations

3.1 Terms

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

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

3.2 Symbols

Void.

3.3 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].

CRM	Compute Resource Management
DF	Deployment Flavour
EM	Element Manager
FM	Fault Management
IFS	Interoperability Features Statement
KPI	Key Performance Indicators
MANO	Management and Orchestration
NFV	Network Function Virtualisation
NFVI	NFV Infrastructure
NFVO	NFV Orchestrator
NIC	Network Interface Controller
NS	Network Service
NSD	NS Descriptor
NRM	Network Resource Management
PM	Performance Management
SRM	Storage Resource Management
SUT	System Under Test
VIM	Virtual Infrastructure Manager
VL	Virtual Link
VLD	VL Descriptor
VM	Virtual Machine
VNF	Virtual Network Functions
VNFC	VNF Component
VNFD	VNF Descriptor
VNFFG	VNF Forwarding Graph
VNFFGD	VNFFG Descriptor
VNFM	VNF Manager
VR	Virtual Resource

4 Interoperability Features Statement (IFS)

4.1 Introduction

This clause compiles the Interoperability Feature Statement (IFS) for VIM, VNFM, EM/VNF and NFVO according to clause 4.3.3 of ETSI GS NFV-TST 002 [i.1] and following the example in ETSI GS NFV-TST 002 [i.1], annex A.

The interoperability feature statement identifier starts with a reference to the relevant functional block that produces or consumes the interface:

- IDs starting with "VIM_" designate features on interfaces produced or consumed by the VIM
- IDs starting with "VNFM_" designate features on interfaces produced or consumed by the VNFM
- IDs starting with "NFVO_" designate features on interfaces produced or consumed by the NFVO
- IDs starting with "EM_" designate features on interfaces produced or consumed by the EM/VNF

The purpose of the Interoperable Features Statement (IFS) is to identify those standardized functions which a FUT supports, including those which are optional and those which are conditional on the support of other functions. Supported interoperability feature statements should be indicated by Y in the Support column (or N if not supported). The Details column can be used to add further support details when necessary.

4.2 IFS for VIM

4.2.1 Software Image Management

Table 4.2.1-1

Functional Block	VIM			
Functional Area	Software Image Management			
Observed Reference Point	Or-Vi			
Observed Interface	Software Image Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_SWIM_QUERY_IM_BY_NFVO	VIM supports software image information queries by the NFVO	M		
VIM_SWIM_ADD_IM	VIM supports "add image" operations by the NFVO	M		
VIM_SWIM_UPDATE_IM	VIM supports "update image" operations by the NFVO	M		
VIM_SWIM_DELETE_IM	VIM supports "delete image" operations by the NFVO	M		

Table 4.2.1-2

Functional Block	VIM			
Functional Area	Software Image Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Software Image Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_SWIM_QUERY_IM_BY_VNFM	VIM supports software image information queries by the VNFM	M		

4.2.2 VNF Package Management

No IFS needed for VNF Package Management on the VIM.

4.2.3 VNF Lifecycle Management

Table 4.2.3-1

Functional Block	VIM			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Compute Resource Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_CRM_ALLOCATE_BY_VNFM	VIM supports "allocate compute resource" operation requests from the VNFM	M		
VIM_CRM_INFO_TO_VNFM	VIM can send compute resource information to the VNFM	M		
VIM_CRM_OPERATE_BY_VNFM	VIM supports "operate compute resource" operation requests from the VNFM	M		
VIM_CRM_TERMINATE_BY_VNFM	VIM supports "terminate compute resource" operation requests from the VNFM	M		

Table 4.2.3-2

Functional Block	VIM			
Functional Area	Network Resource Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Network Resource Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_NRM_ALLOCATE_BY_VNFM	VIM supports "allocate network resource" operation requests from the VNFM	M		
VIM_NRM_INFO_TO_VNFM	VIM can send network resource information to the VNFM	M		
VIM_NRM_TERMINATE_BY_VNFM	VIM supports "terminate network resource" operation requests from the VNFM	M		

Table 4.2.3-3

Functional Block	VIM			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Storage Resource Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_SRM_ALLOCATE_BY_VNFM	VIM supports "allocate storage resource" operation requests from the VNFM	M		
VIM_SRM_INFO_TO_VNFM	VIM can send storage resource information to the VNFM	M		
VIM_SRM_TERMINATE_BY_VNFM	VIM supports "terminate storage resource" operation requests from the VNFM	M		

4.2.4 Fault Management

Table 4.2.4-1

Functional Block	VIM			
Functional Area	Fault Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Resources Fault Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_FM_SUBSCRIBE_BY_NFVO	VIM supports alarm notifications subscriptions by the NFVO	M		
VIM_FM_NOTIFY_BY_NFVO	VIM can generate virtualised resources fault alarm notifications to the NFVO	M		
VIM_FM_QUERY_BY_NFVO	VIM supports virtualised resource fault alarm list queries by the NFVO	M		

Table 4.2.4-2

Functional Block	VIM			
Functional Area	Fault Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Resources Fault Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_FM_SUBSCRIBE_BY_VNFM	VIM supports alarm notifications subscriptions by the VNFM	M		
VIM_FM_NOTIFY_BY_VNFM	VIM can generate virtualised resources fault alarm notifications to the VNFM	M		
VIM_FM_QUERY_BY_VNFM	VIM supports virtualised resources fault alarm list queries by the VNFM	M		

4.2.5 Performance Management

Table 4.2.5-1

Functional Block	VIM			
Functional Area	Fault Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Resources Performance Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_PM_PMJOB_CREATE_BY_NFVO	VIM supports VR PM jobs creation by the NFVO	M		
VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO	VIM supports VR PM job subscriptions from the NFVO	M		
VIM_PM_PMJOB_NOTIFY_BY_NFVO	VIM can generate VR PM notifications to the NFVO	M		
VIM_PM_PMJOB_QUERY_BY_NFVO	VIM supports VR PM job queries by the NFVO	M		
VIM_PM_PMJOB_DELETE_BY_NFVO	VIM supports VR PM jobs deletion by the NFVO	M		
VIM_PM_PMTH_CREATE_BY_NFVO	VIM supports VR PM thresholds creation by the NFVO	M		
VIM_PM_PMTH_QUERY_BY_NFVO	VIM supports VR PM threshold queries by the NFVO	M		
VIM_PM_PMTH_DELETE_BY_NFVO	VIM supports VR PM thresholds deletion by the NFVO	M		

Table 4.2.5-2

Functional Block	VIM			
Functional Area	Performance Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Resources Performance Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_PM_PMJOB_CREATE_BY_VNFM	VIM supports VR PM jobs creation by the VNFM	M		
VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM	VIM supports VR PM job subscriptions from the VNFM	M		
VIM_PM_PMJOB_NOTIFY_BY_VNFM	VIM can generate VR PM notifications to the VNFM	M		
VIM_PM_PMJOB_QUERY_BY_VNFM	VIM supports VR PM job queries by the VNFM	M		
VIM_PM_PMJOB_DELETE_BY_VNFM	VIM supports VR PM jobs deletion by the VNFM	M		
VIM_PM_PMTH_CREATE_BY_VNFM	VIM supports VR PM thresholds creation by the VNFM	M		
VIM_PM_PMTH_QUERY_BY_VNFM	VIM supports VR PM threshold queries by the VNFM	M		
VIM_PM_PMTH_DELETE_BY_VNFM	VIM supports VR PM thresholds deletion by the VNFM	M		

4.2.6 NS Lifecycle Management

Table 4.2.6-1

Functional Block	VIM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Software Image Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_SWIM_QUERY_IM_BY_NFVO	VIM supports software image information queries by the NFVO	M		

Table 4.2.6-2

Functional Block	VIM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Software Image Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_SWIM_QUERY_IM_BY_VNFM	VIM supports software image information queries by the VNFM	M		

Table 4.2.6-3

Functional Block	VIM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Compute Resource Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_CRM_ALLOCATE_BY_NFVO	VIM supports "allocate compute resource" operation requests from the NFVO	M		
VIM_CRM_TERMINATE_BY_NFVO	VIM supports "terminate compute resource" operation requests from the NFVO	M		
VIM_CRM_INFO_TO_NFVO	VIM can send compute resource information to the NFVO	M		
VIM_CRM_OPERATE_BY_NFVO	VIM supports "operate compute resource" operation requests from the NFVO	M		

Table 4.2.6-4

Functional Block	VIM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Network Resource Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_NRM_ALLOCATE_BY_NFVO	VIM supports "allocate network resource" operation requests from the NFVO	M		
VIM_NRM_TERMINATE_BY_NFVO	VIM supports "terminate network resource" operation requests from the NFVO	M		
VIM_NRM_INFO_TO_NFVO	VIM can send network resource information to the NFVO	M		

Table 4.2.6-5

Functional Block	VIM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Storage Resource Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_SRM_ALLOCATE_BY_NFVO	VIM supports "allocate storage resource" operation requests from the NFVO	M		
VIM_SRM_TERMINATE_BY_NFVO	VIM supports "terminate storage resource" operation requests from the NFVO	M		
VIM_SRM_INFO_TO_NFVO	VIM can send storage resource information to the NFVO	M		

Table 4.2.6-6

Functional Block	VIM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Resources Performance Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_PM_PMJOB_CREATE_BY_NFVO	VIM supports VR PM jobs creation by the NFVO	M		
VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO	VIM supports VR PM job subscriptions from the NFVO	M		
VIM_PM_PMJOB_NOTIFY_BY_NFVO	VIM can generate VR PM notifications to the NFVO	M		
VIM_PM_PMTH_CREATE_BY_NFVO	VIM supports VR PM thresholds creation by the NFVO	M		

Table 4.2.6-7

Functional Block	VIM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Resources Performance Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VIM_PM_PMJOB_CREATE_BY_VNFM	VIM supports VR PM jobs creation by the VNFM	M		
VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM	VIM supports VR PM job subscriptions from the VNFM	M		
VIM_PM_PMJOB_NOTIFY_BY_VNFM	VIM can generate VR PM notifications to the VNFM	M		
VIM_PM_PMTH_CREATE_BY_VNFM	VIM supports VR PM thresholds creation by the VNFM	M		

4.3 IFS for VNFM

4.3.1 Software Image Management

Table 4.3.1-1

Functional Block	VNFM			
Functional Area	Software Image Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Software Image Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_SWIM_QUERY_IM	VNFM can query software image information on the VIM	M		

4.3.2 VNF Package Management

No IFS needed for VNF Package Management on the VNFM.

4.3.3 VNF Lifecycle Management

Table 4.3.3-1

Functional Block	VNFM			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Compute Resource Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_CRM_QUERY	VNFM can query compute resource information from the VIM	M		
VNFM_CRM_ALLOCATE	VNFM can generate "allocate compute resource" operation requests to the VIM	M		
VNFM_CRM_OPERATE	VNFM can generate "operate compute resource" operation requests to the VIM	M		
VNFM_CRM_TERMINATE	VNFM can generate "terminate compute resource" operation requests to the VIM	M		

Table 4.3.3-2

Functional Block	VNFM			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Network Resource Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_NRM_QUERY	VNFM can query network resource information from the VIM	M		
VNFM_NRM_ALLOCATE	VNFM can generate "allocate network resource" operation requests to the VIM	M		
VNFM_NRM_TERMINATE	VNFM can generate "terminate network resource" operation requests to the VIM	M		

Table 4.3.3-3

Functional Block	VNFM			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Storage Resource Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_SRM_QUERY	VNFM can query storage resource information from the VIM	M		
VNFM_SRM_ALLOCATE	VNFM can generate "allocate storage resource" operation requests to the VIM	M		
VNFM_SRM_TERMINATE	VNFM can generate "terminate storage resource" operation requests to the VIM	M		

Table 4.3.3-4

Functional Block	VNFM			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Lifecycle Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	Feature	Status	Support	Details
VNFM_VNFLCM_QUERY	VNFM supports VNF information queries by the NFVO	M		

Table 4.3.3-5

Functional Block	VNFM			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Ve-Vnfm			
Observed Interface	VNF Lifecycle Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 008 [i.6]			
Interoperability Features				
Id	Feature	Status	Support	Details
VNFM_VNFLCM_VNF_INSTANTIATE	VNFM supports "instantiate VNF" requests from the EM	M		
VNFM_VNFLCM_VNF_QUERY	VNFM supports "query VNF" requests from the EM	M		
VNFM_VNFLCM_VNF_MODIFY_INFO	VNFM supports "modify VNF information" requests from the EM	M		
VNFM_VNFLCM_VNF_OPERATE	VNFM supports "operate VNF" requests from the EM	M		
VNFM_VNFLCM_VNF_HEAL	VNFM supports "heal VNF" requests from the EM/VNF	M		
VNFM_VNFLCM_VNF_SCALE_OUT	VNFM supports "scale out by adding VNFC instances" requests from the EM/VNF	M		
VNFM_VNFLCM_VNF_SCALE_IN	VNFM supports "scale in by removing VNFC instances" requests from the EM/VNF	M		
VNFM_VNFLCM_VNF_TERMINATE	VNFM supports "terminate VNF" requests from the EM	M		

Table 4.3.3-6

Functional Block	VNFM			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Lifecycle Operation Granting			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	Feature	Status	Support	Details
VNFM_VNFLCM_GRANTING	VNFM can generate "Grant VNF Lifecycle Operation" requests to the NFVO	O		

4.3.4 Fault Management

Table 4.3.4-1

Functional Block	VNFM			
Functional Area	Fault Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Resources Fault Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_FM_VR_SUBSCRIBE	VNFM can subscribe to alarm notifications on the VIM	M		
VNFM_FM_VR_NOTIFY	VNFM can process virtualised resource fault alarm notifications from by the VIM	M		
VNFM_FM_VR_GET_ALARM	VNFM can get the list of virtualised resource fault alarms from the VIM	M		

Table 4.3.4-2

Functional Block	VNFM			
Functional Area	Fault Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Fault Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_FM_VNF_SUBSCRIBE	VNFM supports alarm notifications subscriptions from by the NFVO	M		
VNFM_FM_VNF_NOTIFY	VNFM can generate VNF fault alarm notifications to the NFVO	M		
VNFM_FM_VNF_QUERY	VNFM supports VNF fault alarm list queries by the NFVO	M		

4.3.5 Performance Management

Table 4.3.5-1

Functional Block	VNFM			
Functional Area	Performance Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Resources Performance Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_PM_VR_PMJOB_CREATE	VNFM can create VR PM jobs on the VIM	M		
VNFM_PM_VR_PMJOB_SUBSCRIBE	VNFM can subscribe to VR PM jobs on the VIM	M		
VNFM_PM_VR_PMJOB_NOTIFY	VNFM supports VR PM job notifications from the VIM	M		
VNFM_PM_VR_PMJOB_QUERY	VNFM can query VR PM jobs from the VIM	M		
VNFM_PM_VR_PMJOB_DELETE	VNFM can delete VR PM jobs from the VIM	M		
VNFM_PM_VR_PMTH_CREATE	VNFM can create VR PM thresholds on the VIM	M		
VNFM_PM_VR_PMTH_QUERY	VNFM can query VR PM thresholds from the VIM	M		
VNFM_PM_VR_PMTH_DELETE	VNFM can delete VR PM thresholds from the VIM	M		

Table 4.3.5-2

Functional Block	VNFM			
Functional Area	Performance Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Performance Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_PM_VNF_PMJOB_CREATE	VNFM supports VNF PM jobs creation by the NFVO	M		
VNFM_PM_VNF_PMJOB_SUBSCRIBE	VNFM supports VNF PM job subscriptions from the NFVO	M		
VNFM_PM_VNF_PMJOB_NOTIFY	VNFM can generate VNF PM notifications to the NFVO	M		
VNFM_PM_VNF_PMJOB_QUERY	VNFM supports VNF PM job queries by the NFVO	M		
VNFM_PM_VNF_PMJOB_DELETE	VNFM supports VNF PM jobs deletion by the NFVO	M		
VNFM_PM_VNF_PMTH_CREATE	VNFM supports VNF PM thresholds creation by the NFVO	M		
VNFM_PM_VNF_PMTH_QUERY	VNFM supports VNF PM threshold queries by the NFVO	M		
VNFM_PM_VNF_PMTH_DELETE	VNFM supports VNF PM thresholds deletion by the NFVO	M		

4.3.6 NS Lifecycle Management

Table 4.3.6-1

Functional Block	VNFM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Vi-Vnfm			
Observed Interface	Virtualised Resources Performance Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 006 [i.4]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_PM_VR_PMJOB_CREATE	VNFM can create VR PM jobs on the VIM	M		
VNFM_PM_VR_PMJOB_SUBSCRIBE	VNFM can subscribe to VR PM jobs on the VIM	M		
VNFM_PM_VR_PMJOB_NOTIFY	VNFM supports VR PM job notifications from the VIM	M		

Table 4.3.6-2

Functional Block	VNFM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Lifecycle Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_VNFLCM_CREATE_VNFID	VNFM supports "create VNF identifier" operation requests from the NFVO	M		
VNFM_VNFLCM_INSTANTIATE	VNFM supports "instantiate VNF" operation requests from the NFVO	M		
VNFM_VNFLCM_DELETE_VNFID	VNFM supports "delete VNF identifier" operation requests from the NFVO	M		
VNFM_VNFLCM_TERMINATE	VNFM supports "terminate VNF" operation requests from the NFVO	M		
VNFM_VNFLCM_QUERY	VNFM supports VNF information queries by the NFVO	M		
VNFM_VNFLCM_HEAL	VNFM supports VNF healing requests from the NFVO	M		
VNFM_VNFLCM_OPERATE	VNFM supports "operate VNF" operation requests from the NFVO	M		
VNFM_VNFLCM_NS_SCALE_OUT	VNFM supports "scale out by adding VNF instances" requests from the NFVO	M		
VNFM_VNFLCM_NS_SCALE_IN	VNFM supports "scale in by removing VNF instances" requests from the NFVO	M		
VNFM_VNFLCM_VNF_SCALE_OUT	VNFM supports "scale out by adding VNFC instances" requests from the NFVO	M		
VNFM_VNFLCM_VNF_SCALE_IN	VNFM supports "scale in by removing VNFC instances" requests from the NFVO	M		

Table 4.3.6-3

Functional Block	VNFM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	Indicator			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_NSVNFINDI_SUBSCRIBE	VNFM supports VNF indicator subscriptions from the NFVO	M		
VNFM_NSVNFINDI_NOTIFY	VNFM can generate VNF indicator notifications to the NFVO	M		

Table 4.3.6-4

Functional Block	VNFM			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Ve-Vnfm			
Observed Interface	Indicator			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 008 [i.6]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_VNFINDI_SUBSCRIBE	VNFM can subscribe to VNF indicators on the EM/VNF	M		
VNFM_VNFINDI_NOTIFY	VNFM supports VNF indicator notifications from the EM/VNF	M		

4.4 IFS for NFVO

4.4.1 Software Image Management

Table 4.4.1-1

Functional Block	NFVO			
Functional Area	Software Image Management			
Observed Reference Point	Or-Vi			
Observed Interface	Software Image Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_SWIM_ADD_IM	NFVO can add software images on the VIM	M		
NFVO_SWIM_QUERY_IM	NFVO can query software image information on the VIM	M		
NFVO_SWIM_UPDATE_IM	NFVO can update software image information on the VIM	M		
NFVO_SWIM_DELETE_IM	NFVO can delete software images from the VIM	M		

4.4.2 VNF Package Management

Table 4.4.2-1

Functional Block	NFVO			
Functional Area	VNF Package Management			
Observed Reference Point	Os-Ma-Nfvo			
Observed Interface	VNF Package Management			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 013 [i.9]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_VNFPM_VALIDATE	NFVO can check the integrity and validity of VNF Packages	M		
NFVO_VNFPM_ACCESS	NFVO can access the VNF catalogue where the VNF Packages information is stored	M		
NFVO_VNFPM_QUERY	NFVO supports on-boarded VNF Packages queries	M		
NFVO_VNFPM_ABORT_DELETE	NFVO supports the delete VNF Package operation abortion	M		

4.4.3 VNF Lifecycle Management

Table 4.4.3-1

Functional Block	NFVO			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Lifecycle Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_VNFLCM_QUERY	NFVO can query VNF information from the VNFM	M		

Table 4.4.3-2

Functional Block	NFVO			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Lifecycle Operation Granting			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_VNFLCM_GRANTING	NFVO supports "Grant VNF Lifecycle Operation" requests from the VNFM	O		

4.4.4 Fault Management

Table 4.4.4-1

Functional Block	NFVO			
Functional Area	Fault Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Resource Fault Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_FM_VR_SUBSCRIBE	NFVO can subscribe to virtualised resource fault alarms on the VIM	M		
NFVO_FM_VR_NOTIFY	NFVO can process virtualised resource fault alarm notifications from the VIM	M		
NFVO_FM_VR_GET_ALARMS	NFVO can get the list of virtualised resource fault alarms from the VIM	M		

Table 4.4.4-2

Functional Block	NFVO			
Functional Area	Fault Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Fault Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_FM_VNF_SUBSCRIBE	NFVO can subscribe to alarm notifications on the VNFM	M		
NFVO_FM_VNF_NOTIFY	NFVO can process VNF fault alarm notifications from by the VNFM	M		
NFVO_FM_VNF_GET_ALARMS	NFVO can get the list of VNF fault alarms from the VNFM	M		

4.4.5 Performance Management

Table 4.4.5-1

Functional Block	NFVO			
Functional Area	Performance Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Resource Performance Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_PM_VR_PMJOB_CREATE	NFVO can create VR PM jobs on the VIM	M		
NFVO_PM_VR_PMJOB_SUBSCRIBE	NFVO can subscribe to VR PM jobs on the VIM	M		
NFVO_PM_VR_PMJOB_NOTIFY	NFVO supports VR PM notifications from the VIM	M		
NFVO_PM_VR_PMJOB_QUERY	NFVO can query VR PM jobs from the VIM	M		
NFVO_PM_VR_PMJOB_DELETE	NFVO can delete VR PM jobs from the VIM	M		
NFVO_PM_VR_PMTH_CREATE	NFVO can create VR PM thresholds on the VIM	M		
NFVO_PM_VR_PMTH_QUERY	NFVO can query VR PM thresholds from the VIM	M		
NFVO_PM_VR_PMTH_DELETE	NFVO can delete VR PM thresholds from the VIM	M		

Table 4.4.5-2

Functional Block	NFVO			
Functional Area	Performance Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Performance Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_PM_VNF_PMJOB_CREATE	NFVO can create VNF PM jobs on the VNFM	M		
NFVO_PM_VNF_PMJOB_SUBSCRIBE	NFVO can subscribe to VNF PM jobs on the VNFM	M		
NFVO_PM_VNF_PMJOB_NOTIFY	NFVO supports VNF PM notifications from the VNFM	M		
NFVO_PM_VNF_PMJOB_QUERY	NFVO can query VNF PM jobs from the VNFM	M		
NFVO_PM_VNF_PMJOB_DELETE	NFVO can create VNF PM jobs on the VNFM	M		
NFVO_PM_VNF_PMTH_CREATE	NFVO can create VNF PM thresholds on the VNFM	M		
NFVO_PM_VNF_PMTH_QUERY	NFVO can query VNF PM thresholds from the VNFM	M		
NFVO_PM_VNF_PMTH_DELETE	NFVO can delete VNF PM thresholds from the VNFM	M		

4.4.6 NS Lifecycle Management

Table 4.4.6-1

Functional Block	NFVO			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Software Image Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_SWIM_QUERY_IM	NFVO can query software image information on the VIM	M		

Table 4.4.6-2

Functional Block	NFVO			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Compute Resource Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_CRM_QUERY	NFVO can query compute resource information from the VIM	M		
NFVO_CRM_ALLOCATE	NFVO can generate "allocate compute resource" operation requests to the VIM	M		
NFVO_CRM_OPERATE	NFVO can generate "operate compute resource" operation requests to the VIM	M		
NFVO_CRM_TERMINATE	NFVO can generate "terminate compute resource" operation requests to the VIM	M		

Table 4.4.6-3

Functional Block	NFVO			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Network Resource Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_NRM_QUERY	NFVO can query network resource information from the VIM	M		
NFVO_NRM_ALLOCATE	NFVO can generate "allocate network resource" operation requests to the VIM	M		
NFVO_NRM_TERMINATE	NFVO can generate "terminate network resource" operation requests to the VIM	M		

Table 4.4.6-4

Functional Block	NFVO			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Storage Resource Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_SRM_QUERY	NFVO can query storage resource information from the VIM	M		
NFVO_SRM_ALLOCATE	NFVO can generate "allocate storage resource" operation requests to the VIM	M		
NFVO_SRM_TERMINATE	NFVO can generate "terminate storage resource" operation requests to the VIM	M		

Table 4.4.6-5

Functional Block	NFVO			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vi			
Observed Interface	Virtualised Resource Performance Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 005 [i.3]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_PM_VR_PMJOB_CREATE	NFVO can create VR PM jobs on the VIM	M		
NFVO_PM_VR_PMJOB_SUBSCRIBE	NFVO can subscribe to VR PM jobs on the VIM	M		
NFVO_PM_VR_PMJOB_NOTIFY	NFVO supports VR PM notifications from the VIM	M		

Table 4.4.6-6

Functional Block	NFVO			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	VNF Lifecycle Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
NFVO_VNFLCM_CREATE_VNFID	NFVO can generate "create VNF identifier" operation requests to the VNFM	M		
NFVO_VNFLCM_INSTANTIATE	NFVO can generate "instantiate VNF" operation requests to the VNFM	M		
NFVO_VNFLCM_DELETE_VNFID	NFVO can generate "delete VNF identifier" operation requests to the VNFM	M		
NFVO_VNFLCM_TERMINATE	NFVO can generate "terminate VNF" operation requests to the VNFM	M		
NFVO_VNFLCM_QUERY	NFVO can query VNF information from the VNFM	M		
NFVO_VNFLCM_HEAL	NFVO can generate VNF healing requests to the VNFM	M		
NFVO_VNFLCM_OPERATE	NFVO can generate "operate VNF" operation requests to the VNFM	M		
NFVO_VNFLCM_NS_SCALE_OUT	NFVO can generate "scale out by adding VNF instances" requests to the VNFM	M		
NFVO_VNFLCM_NS_SCALE_IN	NFVO can generate "scale in by removing VNF instances" requests to the VNFM	M		
NFVO_VNFLCM_VNF_SCALE_OUT	NFVO can generate "scale out by adding VNFC instances" to the VNFM	M		
NFVO_VNFLCM_VNF_SCALE_IN	NFVO can generate "scale in by removing VNFC instances" to the VNFM	M		

Table 4.4.6-7

Functional Block	NFVO			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Or-Vnfm			
Observed Interface	Indicator			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 007 [i.5]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
VNFM_NSVNFINDI_SUBSCRIBE	NFVO can subscribe to VNF indicators on the VNFM	M		
VNFM_NSVNFINDI_NOTIFY	NFVO supports VNF indicator notifications from the VNFM	M		

4.5 IFS for EM/VNF

4.5.1 Software Image Management

No IFS needed for Software Image Management on the EM/VNF.

4.5.2 VNF Package Management

No IFS needed for VNF Package Management on the EM/VNF.

4.5.3 VNF Lifecycle Management

Table 4.5.3-1

Functional Block	EM/VNF			
Functional Area	VNF Lifecycle Management			
Observed Reference Point	Ve-Vnfm			
Observed Interface	VNF Lifecycle Management			
Producer/Consumer	Consumer			
References	ETSI GS NFV-IFA 008 [i.6]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
EM_VNFLCM_VNF_INSTANTIATE	EM can generate "Instantiate VNF" requests to the VNFM	M		
EM_VNFLCM_VNF_QUERY	EM can generate "query VNF" requests to the VNFM	M		
EM_VNFLCM_VNF_MODIFY_INFO	EM can generate "modify VNF information" requests to the VNFM	M		
EM_VNFLCM_VNF_OPERATE	EM can generate "operate VNF" requests to the VNFM	M		
EM_VNFLCM_VNF_SCALE_OUT	EM/VNF can generate "scale out by adding VNFC instances" requests to the VNFM	M		
EM_VNFLCM_VNF_SCALE_IN	EM/VNF can generate "scale in by removing VNFC instances" requests to the VNFM	M		
EM_VNFLCM_VNF_HEAL	EM/VNF can generate "heal VNF" requests to the VNFM	M		
EM_VNFLCM_VNF_TERMINATE	EM can generate "terminate VNF" requests to the VNFM	M		

4.5.4 Fault Management

No IFS needed for Fault Management on the EM/VNF.

4.5.5 Performance Management

No IFS needed for Performance Management on the EM/VNF.

4.5.6 NS Lifecycle Management

Table 4.5.6-1

Functional Block	EM/VNF			
Functional Area	NS Lifecycle Management			
Observed Reference Point	Ve-Vnfm			
Observed Interface	Indicator			
Producer/Consumer	Producer			
References	ETSI GS NFV-IFA 008 [i.6]			
Interoperability Features				
Id	**Feature**	**Status**	**Support**	**Details**
EM_VNFINDI_SUBSCRIBE	EM/VNF supports VNF indicator subscriptions from the VNFM	M		
EM_VNFINDI_NOTIFY	EM/VNF can generate VNF indicator notifications to the VNFM	M		

5 System Under Test (SUT)

5.1 SUT Configuration 1

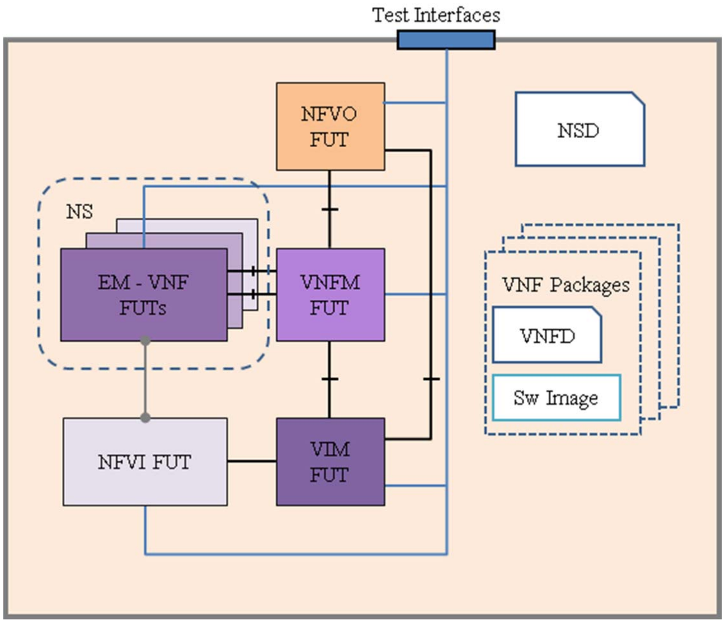

Figure 5.1-1: SUT Configuration 1

5.2 SUT Configuration 2

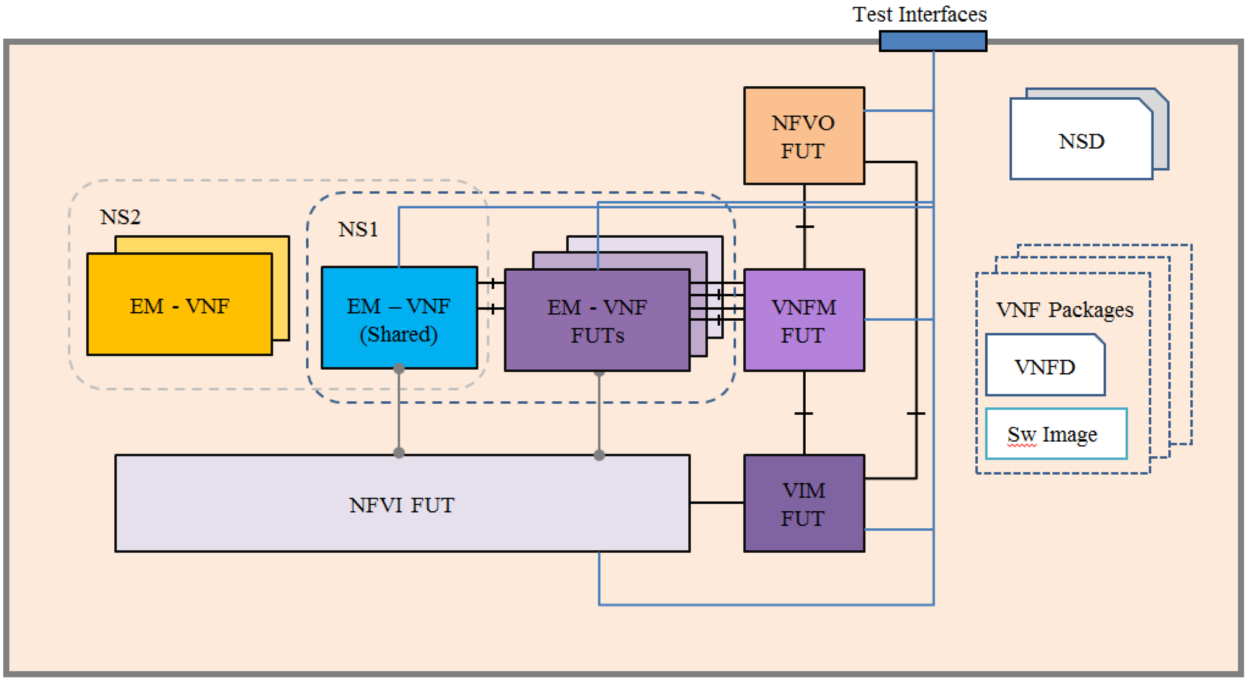

Figure 5.2-1: SUT Configuration 2

5.3 SUT Configuration 3

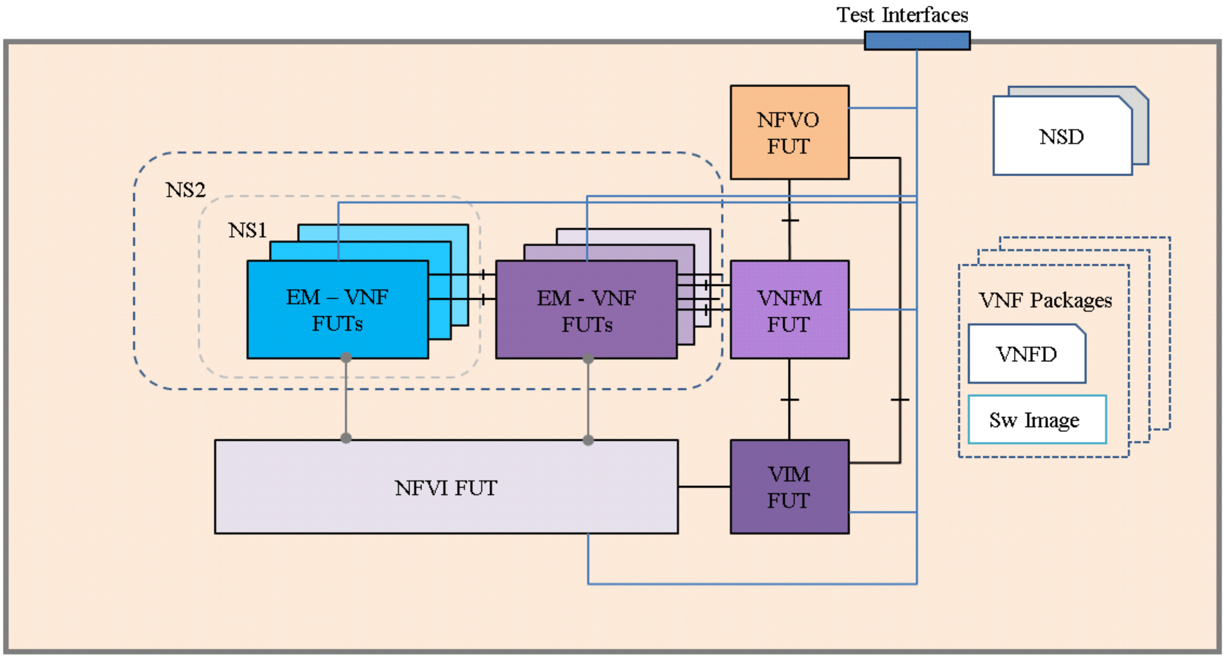

Figure 5.3-1: SUT Configuration 3

5.4 SUT Configuration 4

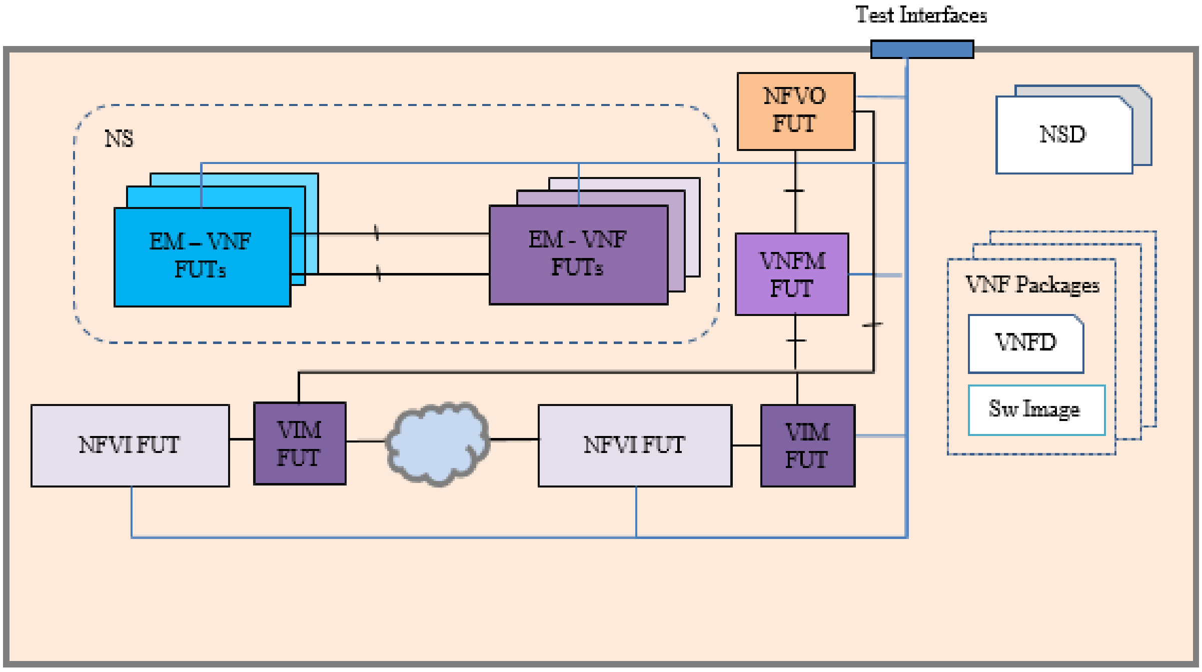

Figure 5.4-1: SUT Configuration 4

6 Test Suite Structure

6.1 Introduction

This clause describes the test suite structure and naming convention for the test descriptions according to ETSI GS NFV-TST 002 [i.1], clause 4.3.5.

The test cases in this test suite are grouped by the major functional areas they cover. The test identifiers use a common prefix of TD_NFV_ with an additional unique string for each area. Each test has a unique test identifier and a running number of the test case (in case of additional test case that cover similar test purpose).

The test case groups and their corresponding identifier naming prefix are described in table 6.1-1.

Table 6.1-1

Test Case Group	Identifier Name Prefix
Software Image Management	TD_NFV_SWIM_
VNF Package Management	TD_NFV_VNFPM_
VNF Lifecycle Management	TD_NFV_VNFLCM_
Fault Management	TD_NFV_FM_
Performance Management	TD_NFV_PM_
NS Lifecycle Management	TD_NFV_NSLCM_

6.2 Software Image Management Test Cases Overview

Table 6.2-1

Test description identifier	Test purpose
TD_NFV_SWIM_ADD_001	Verify that the NFVO can add a software image to the image repository managed by the VIM
TD_NFV_SWIM_QUERY_001	Verify that the NFVO can retrieve the information of a software image from the image repository managed by the VIM
TD_NFV_SWIM_QUERY_002	Verify that the VNFM can retrieve the information of a software image from the image repository managed by the VIM
TD_NFV_SWIM_UPDATE_001	Verify that the NFVO can update the metadata of a software image in the image repository managed by the VIM
TD_NFV_SWIM_DELETE_001	Verify that the NFVO can delete a software image from the image repository managed by the VIM

6.3 VNF Package Management Test Cases Overview

Table 6.3-1

Test case Identifier	Test case purpose
TD_NFV_VNFPM_ONBOARD_001	To verify that a VNF Package can be successfully on-boarded to the VNF catalogue managed by the NFVO
TD_NFV_VNFPM_DELETE_001	To verify that an on-boarded VNF Package can be successfully deleted from the VNF catalogue managed by the NFVO
TD_NFV_VNFPM_ABORTDEL_001	To verify that an on-boarded VNF Package that has been in deletion pending state can be successfully aborted the deletion by the NFVO

6.4 VNF Lifecycle Management Test Cases Overview

Table 6.4-1

Test case Identifier	Test case purpose
TD_NFV_VNFLCM_INSTANTIATE_VNF_001	Verify that a VNF be successfully instantiated when a "instantiate VNF" operation is triggered by the EM
TD_NFV_VNFLCM_QUERY_VNF_001	Verify that the VNF instance's information can be queried successfully by the EM
TD_NFV_VNFLCM_MODIFY_VNF_INFO_001	Verify that the VNF information can be updated successfully by the EM
TD_NFV_VNFLCM_START_VNF_001	Verify that a VNF/VNFC instance be successfully started when an "operate VNF" operation is triggered by the EM
TD_NFV_VNFLCM_STOP_VNF_001	Verify that a VNF/VNFC instance be successfully stopped when an "operate VNF" operation is triggered by the EM
TD_NFV_VNFLCM_HEAL_VNF_001	Verify that a VNF can be successfully healed when VNF healing request is sent out by the EM/VNF
TD_NFV_VNFLCM_SCALE_OUT_VNF_001	Verify that a VNF can be successfully scaled out by adding VNFC instances triggered by an EM/VNF request
TD_NFV_VNFLCM_SCALE_IN_VNF_001	Verify that a VNF can be successfully scaled in by removing VNFC instances triggered by an EM/VNF request
TD_NFV_VNFLCM_TERMINATE_VNF_001	Verify that a VNF be successfully terminated when a "terminate VNF" operation is triggered by the EM

6.5 Fault Management Test Cases Overview

Table 6.5-1

Test case Identifier	Test case purpose
TD_NFV_FM_VR_NOTIFY_001	Verify that a fault alarm notification propagates to the NFVO when a virtualised resource that is required for the NS connectivity fails
TD_NFV_FM_VR_CLEAR_001	Verify that a fault clearance notification propagates to the NFVO when a failed virtualised resource that is required for the NS connectivity is recovered
TD_NFV_FM_VNF_NOTIFY_001	Verify that a VNF fault alarm notification propagates via the VNFM to the NFVO when a VNF fault is triggered by a failed virtualised resource
TD_NFV_FM_VNF_CLEAR_001	Verify that a VNF fault alarm clearance notification propagates via the VNFM to the NFVO when a VNF fault is cleared by resolving a failed virtualised resource
TD_NFV_FM_NS_ALARM_SUBSCRIPTION_CREATE_001	Verify that it is possible to subscribe to alarm notifications related to faults on the NS instance
TD_NFV_FM_NS_ALARM_SUBSCRIPTION_DELETE_001	Verify that the subscription to NS faults alarm notifications can be deleted
TD_NFV_FM_NS_ALARM_NOTIFICATION_001	Verify that a fault alarm notification is exposed by the NFVO when a NS related resource fails
TD_NFV_FM_NS_ALARM_CLEAR_NOTIFICATION_001	Verify that an alarm clear notification is dispatched by the NFVO when a NS related resource failure is cleared
TD_NFV_FM_NS_ALARM_QUERY_001	Verify that an alarm clear notification is dispatched by the NFVO when a NS related resource failure is cleared

6.6 Performance Management Test Cases Overview

Table 6.6-1

Test case Identifier	Test case purpose
TD_NFV_PM_VR_CREATE_NOTIFY_001	Verify that the performance metrics of a virtualised resource that is required for a NS instance connectivity can be monitored using PM jobs and notifications
TD_NFV_PM_VR_CREATE_THRESHOLD_001	Verify that the performance metrics of a virtualised resource that is required for a NS instance connectivity can be monitored using PM jobs and thresholds
TD_NFV_PM_VR_DELETE_MONITOR_001	Verify that the monitoring of performance metrics of a virtualised resource that is required for a NS instance connectivity can be stopped by deleting PM jobs
TD_NFV_PM_VR_DELETE_THRESHOLD_001	Verify that a threshold created for a virtualised resource that is required for a NS instance connectivity can be deleted
TD_NFV_PM_VNF_CREATE_NOTIFY_001	Verify that the performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be monitored using PM jobs and notifications
TD_NFV_PM_VNF_CREATE_THRESHOLD_001	Verify that the performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be monitored using PM jobs and thresholds
TD_NFV_PM_VNF_DELETE_MONITOR_001	Verify that the monitoring of performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be stopped by deleting PM jobs
TD_NFV_PM_VNF_DELETE_THRESHOLD_001	Verify that a threshold created for a virtualised resource that is allocated to a VNF instance inside a NS instance can be deleted
TD_NFV_PM_NS_MONITORING_JOB_CREATE_001	Verify that performance monitoring job for monitoring NS related metrics can be created to start monitoring NS performance metrics
TD_NFV_PM_NS_PERFORMANCE_METRICS_QUERY_001	Verify that NS related performance monitoring metrics can be retrieved in the form of reports by querying the NFVO
TD_NFV_PM_NS_THRESHOLD_CREATE_001	Verify that performance monitoring thresholds can be created for one or more NS related performance metrics on the NFVO
TD_NFV_PM_NS_SUBSCRIPTION_CREATE_001	Verify that it is possible to subscribe to NS performance metrics notifications related to threshold crossed notifications
TD_NFV_PM_NS_SUBSCRIPTION_CREATE_002	Verify that it is possible to subscribe to NS performance metrics notifications related to availability of performance information
TD_NFV_PM_NS_MONITORING_INFO_NOTIFICATION_001	Verify that a monitoring information availability notification is exposed by the NFVO when new and updated NS performance monitoring metrics are available
TD_NFV_PM_NS_SUBSCRIPTION_DELETE_001	Verify that the subscription to NS performance monitoring notifications can be deleted
TD_NFV_PM_NS_MONITORING_JOB_DELETE_001	Verify that performance monitoring job for monitoring NS related metrics can be deleted to stop monitoring NS performance metrics
TD_NFV_PM_NS_THRESHOLD_DELETE_001	Verify that performance monitoring thresholds can be deleted for one or more NS related performance metrics on the NFVO

6.7 NS Lifecycle Management Test Cases Overview

Table 6.7-1

Test case Identifier	Test case purpose
TD_NFV_NSLCM_INSTANTIATE_001	To verify that a standalone NS can be successfully instantiated
TD_NFV_NSLCM_INSTANTIATE_NEST_NS_001	To verify that a NS referencing an existing nested NS can be successfully instantiated
TD_NFV_NSLCM_INSTANTIATE_MULTISITE_001	To verify that an NS can be successfully instantiated across different sites
TD_NFV_NSLCM_INSTANTIATE_SFC_001	To verify that an NS with NSH based SFC can be successfully instantiated
TD_NFV_NSLCM_SCALE_OUT_001	Verify that the NS can be successfully scaled out by adding VNF instances triggered by an operator action

Test case Identifier	Test case purpose
TD_NFV_NSLCM_SCALE_OUT_002	Verify that the NS can be successfully scaled out by adding VNF instances triggered automatically by a VNF indicator notification
TD_NFV_NSLCM_SCALE_OUT_003	Verify that the NS can be successfully scaled out by adding VNF instances triggered automatically by a VIM KPI
TD_NFV_NSLCM_SCALE_OUT_004	Verify that the NS can be successfully scaled out by adding VNF instances triggered automatically in VNFM by querying VNF Indicator
TD_NFV_NSLCM SCALE_OUT_MULTISITE_001	To verify that a multi-site NS can be successfully scaled out (by adding VNF instances) if triggered by a MANO operator
TD_NFV_NSLCM_SCALE_IN_001	Verify that the NS can be successfully scaled in by removing VNF instances triggered by an operator action
TD_NFV_NSLCM_SCALE_IN_002	Verify that the NS can be successfully scaled in by removing VNF instances triggered automatically by a VNF indicator notification
TD_NFV_NSLCM_SCALE_IN_003	Verify that the NS can be successfully scaled in by removing VNF instances triggered automatically by a VIM KPI
TD_NFV_NSLCM_SCALE_IN_004	Verify that the NS can be successfully scaled in by removing VNF instances triggered automatically in VNFM by querying VNF Indicator
TD_NFV_NSLCM_SCALE_IN_MULTISITE_001	To verify that a multi-site NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator
TD_NFV_NSLCM_SCALE_OUT_VNF_001	To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered by a NFVO operator
TD_NFV_NSLCM_SCALE_OUT_VNF_002	To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered automatically by a VNF indicator notification
TD_NFV_NSLCM_SCALE_OUT_VNF_003	To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered automatically by a VIM KPI
TD_NFV_NSLCM_SCALE_OUT_VNF_004	To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered automatically in VNFM by querying VNF Indicator
TD_NFV_NSLCM_SCALE_OUT_VNF_MULTISITE_001	To verify that a VNF in a multi-site NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered by a MANO operator
TD_NFV_NSLCM_SCALE_IN_VNF_001	Verify that a VNF in a NS can be successfully scaled in by removing VNFC instances from an existing VNF triggered by an operator action
TD_NFV_NSLCM_SCALE_IN_VNF_002	Verify that a VNF in a NS can be successfully scaled in by removing VNFC instances triggered automatically by a VNF indicator notification
TD_NFV_NSLCM_SCALE_IN_VNF_003	Verify that a VNF in a NS can be successfully scaled in by removing VNFC instances triggered automatically by a VIM KPI
TD_NFV_NSLCM_SCALE_IN_VNF_004	Verify that a VNF in a NS can be successfully scaled in by removing VNFC instances triggered automatically in VNFM by querying VNF Indicator
TD_NFV_NSLCM_SCALE_IN_VNF_MULTISITE_001	To verify that a VNF in a multi-site NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered by a MANO operator
TD_NFV_NSLCM_SCALE_TO_LEVEL_001	Verify that the NS can be successfully scaled to another existing instantiation level by an operator action
TD_NFV_NSLCM_SCALE_TO_LEVEL_002	Verify that the NS can be successfully scaled to another existing instantiation level automatically by a VNF indicator notification
TD_NFV_NSLCM_SCALE_TO_LEVEL_003	Verify that the NS can be successfully scaled to another existing instantiation level automatically by a VIM KPI
TD_NFV_NSLCM_SCALE_TO_LEVEL_ MULTISITE_001	Verify that a multi-site NS can be successfully scaled to another existing instantiation level (Scale NS to level) by an operator
TD_NFV_NSLCM_SCALE_TO_LEVEL_VNF_001	To verify that a VNF in a NS can be successfully scaled to another existing instantiation level by a NFVO operator

Test case Identifier	Test case purpose
TD_NFV_NSLCM_SCALE_TO_LEVEL_VNF_002	To verify that a VNF in a NS can be successfully scaled to another existing instantiation level by a VNF indicator notification
TD_NFV_NSLCM_SCALE_TO_LEVEL_VNF_003	To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered automatically by a VIM KPI
TD_NFV_NSLCM_SCALE_TO_LEVEL_VNF_MULTISITE_001	Verify that a VNF in a multi-site NS can be successfully scaled to another existing instantiation level (Scale VNF to Level) by an operator
TD_NFV_NSLCM_UPDATE_START_001	Verify the capability to start a VNF instance inside a NS instance
TD_NFV_NSLCM_UPDATE_STOP_001	Verify the capability to stop a VNF instance inside a NS instance
TD_NFV_NSLCM_UPDATE_INST_ADD_VNF_001	To verify that one or more VNFs can be instantiated and the instances added to a running NS instance
TD_NFV_NSLCM_UPDATE_REM_VNF_001	To verify that one or more VNF instances can be removed from a running NS instance
TD_NFV_NSLCM_UPDATE_ADD_SHVNF_001	To verify that one or more shared VNF instances can be added to a running NS instance
TD_NFV_NSLCM_UPDATE_REM_SHVNF_001	To verify that one or more shared VNF instances can be removed from a running NS instance
TD_NFV_NSLCM_UPDATE_ADD_VL_001	To verify that one or more virtual links (VL) can be added to a running NS instance
TD_NFV_NSLCM_UPDATE_REM_VL_001	To verify that one or more virtual links (VL) can be removed from a running NS instance
TD_NFV_NSLCM_UPDATE_VNF_DF_001	To verify that the deployment flavour of one or more VNF instances in a NS instance can be changed
TD_NFV_NSLCM_UPDATE_VNF_CONFIG_001	To verify that the configuration of a VNF running in a NS can be successfully updated
TD_NFV_NSLCM_HEAL_001	Verify that VNF instances inside the NS can be successfully healed when partial NS healing (VNF healing) is triggered by an operator action
TD_NFV_NSLCM_HEAL_002	Verify that a NS can be successfully healed when complete NS healing is triggered by an operator action
TD_NFV_NSLCM_TERMINATE_001	To verify that a standalone NS instance can be successfully terminated
TD_NFV_NSLCM_TERMINATE_NESTED_NS_001	To verify that a NS instance referencing an existing nested NS can be successfully terminated
TD_NFV_NSLCM_TERMINATE_MULTISITE_001	To verify that a multi-Site NS can be successfully terminated
TD_NFV_NSLCM_TERMINATE_SFC_001	To verify that a NS can be successfully terminated

7 Test Descriptions

7.1 Introduction

This clause contains the test descriptions for the NFV functional areas. The results of IOP Checks and IOP Verdict inside each test description form should be indicated by OK/NOK according to clause 4.3.7 of ETSI GS NFV-TST 002 [i.1]. Non-applicable test descriptions can be marked as N/A. Collective results for functional areas can be reported according to the sample report card in annex A of the present document.

7.2 Software Image Management

7.2.1 Add Software Image

Test Description: add software image				
Identifier	TD_NFV_SWIM_ADD_001			
Test Purpose	Verify that the NFVO can add a software image to the image repository managed by the VIM			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3]			
Applicability	• NFVO can add software images on the VIM (NFVO_SWIM_ADD_IM) • VIM supports "add image" operations by the NFVO (VIM_SWIM_ADD_IM) • VIM supports software image information queries by the NFVO (VIM_SWIM_QUERY_IM_BY_NFVO)			
Pre-test conditions	• Software image information (e.g. name, metadata, location) is defined			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger an "add image" operation on the NFVO	
	2	IOP Check	Verify the image information by querying the list of images on the image repository managed by the VIM	
	3	IOP Check	Verify that the NFVO shows no "add image" operation errors	
IOP Verdict				

7.2.2 Query Software Image

7.2.2.1 Query Software Image by NFVO

Test Description: query software image by NFVO				
Identifier	TD_NFV_SWIM_QUERY_001			
Test Purpose	Verify that the NFVO can retrieve the information of a software image from the image repository managed by the VIM			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3]			
Applicability	• NFVO can query software image information on the VIM (NFVO_SWIM_QUERY_IM) • VIM supports software image information queries by the NFVO (VIM_SWIM_QUERY_IM_BY_NFVO)			
Pre-test conditions	• The software image to be queried is added to the image repository managed by the VIM (TD_NFV_SWIM_ADD_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the query of the relevant image information on the NFVO using the unique image identifier	
	2	IOP Check	Verify the image information by querying the list of images on the image repository managed by the VIM	
	3	IOP Check	Verify that the image information obtained by the NFVO matches the image information in step 2	
IOP Verdict				

7.2.2.2 Query Software Image by VNFM

Test Description: query software image by VNFM				
Identifier	TD_NFV_SWIM_QUERY_002			
Test Purpose	Verify that the VNFM can retrieve the information of a software image from the image repository managed by the VIM			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 006 [i.4]			
Applicability	• VNFM can query software image information on the VIM (VNFM_SWIM_QUERY_IM) • VIM supports software image information queries by the VNFM (VIM_SWIM_QUERY_IM_BY_VNFM)			
Pre-test conditions	• The software image to be queried is added to the image repository managed by the VIM (TD_NFV_SWIM_ADD_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the query of the relevant image information on the VNFM using the unique image identifier	
	2	IOP Check	Verify the image information by querying the list of images on the image repository managed by the VIM	
	3	IOP Check	Verify that the image information obtained by the VNFM matches the image information in step 2	
IOP Verdict				

7.2.3 Update Software Image

Test Description: update software image				
Identifier	TD_NFV_SWIM_UPDATE_001			
Test Purpose	Verify that the NFVO can update the metadata of a software image in the image repository managed by the VIM			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3]			
Applicability	• NFVO can update software image information on the VIM (NFVO_SWIM_UPDATE_IM) • VIM supports "update image" operations by the NFVO (VIM_SWIM_UPDATE_IM) • VIM supports software image information queries by the NFVO (VIM_SWIM_QUERY_IM_BY_NFVO)			
Pre-test conditions	• The software image to be updated is added to the image repository managed by the VIM (TD_NFV_SWIM_ADD_001) • The updated software image metadata is defined			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger an "update image" operation on the NFVO using the unique image identifier and including the updated metadata	
	2	IOP Check	Verify the updated image information by querying the list of images on the image repository managed by the VIM	
	3	IOP Check	Verify that the NFVO shows no "update image" operation errors	
IOP Verdict				

7.2.4 Delete Software Image

Test Description: delete software image				
Identifier	TD_NFV_SWIM_DELETE_001			
Test Purpose	Verify that the NFVO can delete a software image from the image repository managed by the VIM			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3]			
Applicability	• NFVO can delete software images from the VIM (NFVO_SWIM_DELETE_IM) • VIM supports "delete image" operations by the NFVO (VIM_SWIM_DELETE_IM) • VIM supports software image information queries by the NFVO (VIM_SWIM_QUERY_IM_BY_NFVO)			
Pre-test conditions	• The software image to be deleted has been added to the image repository managed by the VIM (TD_NFV_SWIM_ADD_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a "delete image" operation on the NFVO using the unique image identifier	
	2	IOP Check	Verify that the affected image has been deleted from the image repository managed by the VIM by querying the list of images	
	3	IOP Check	Verify that the NFVO shows no "delete image" operation errors	
IOP Verdict				

7.3 VNF Package Management

7.3.1 On-board VNF Package

Test Description: on-board VNF package				
Identifier	TD_NFV_VNFPM_ONBOARD_001			
Test Purpose	To verify that a VNF Package can be successfully on-boarded to the VNF catalogue managed by the NFVO			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 013 [i.9] and ETSI GS NFV-IFA 011 [i.8]			
Applicability	• NFVO can check the integrity and validity of VNF Packages (NFVO_VNFPM_VALIDATE) • NFVO can access the VNF catalogue where the VNF Packages information is stored (NFVO_VNFPM_ACCESS)			
Pre-test conditions	• The VNF Package contains all the necessary information including VNFD, software image and artefacts			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the on-boarding of a VNF Package on the NFVO	
	2	IOP Check	Verify that the VNF Package is stored in the VNF catalogue managed by the NFVO	
	3	IOP Check	Verify that a unique identifier for the VNF Package is generated by querying the VNF Package information on the NFVO	
IOP Verdict				

7.3.2 Delete VNF Package

Test Description: delete VNF package				
Identifier	TD_NFV_VNFPM_DELETE_001			
Test Purpose	To verify that an on-boarded VNF Package can be successfully deleted from the VNF catalogue managed by the NFVO			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO can access the VNF catalogue where the VNF Packages information is stored (NFVO_VNFPM_ACCESS) • NFVO supports on-boarded VNF Packages queries (NFVO_VNFPM_QUERY)			
Pre-test conditions	• The VNF Package is on-boarded, not in a "used" state, and not in a "deletion pending" state • All the VNF instances associated to the VNF Package are terminated			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the deletion of a VNF Package on the NFVO	
	2	IOP Check	Verify that VNF Package has been deleted by querying the NFVO	
IOP Verdict				

7.3.3 Abort VNF Package Delete Operation

Test Description: abort VNF package delete operation				
Identifier	TD_NFV_VNFPM_ABORTDEL_001			
Test Purpose	To verify that an on-boarded VNF Package that is in deletion pending state can be successfully aborted from deletion by the NFVO			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO supports the delete VNF Package operation abortion (NFVO_VNFPM_ABORT_DELETE) • NFVO supports on-boarded VNF Packages queries (NFVO_VNFPM_QUERY)			
Pre-test conditions	• The VNF Package is on-boarded and in "deletion pending" state			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the abortion of the deletion operation of an on-boarded VNF Package on NFVO	
	2	IOP Check	Verify that the VNF Package is not deleted by querying the NFVO	
	3	IOP Check	Verify the VNF Package is in disabled state by querying the NFVO	
IOP Verdict				

7.4 VNF Lifecycle Management

7.4.1 Instantiate VNF with an EM Request

Test Description: instantiate VNF with an EM request	
Identifier	TD_NFV_VNFLCM_INSTANTIATE_VNF_001
Test Purpose	Verify that a VNF be successfully instantiated when an "instantiate VNF" operation is triggered by the EM
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 008 [i.6]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can query software image information from the VIM (NFVO_SWIM_QUERY_IM or VNFM_SWIM_QUERY_IM) • VIM supports software image information queries by the NFVO/VNFM (VIM_SWIM_QUERY_IM_BY_NFVO or VIM_SWIM_QUERY_IM_BY_VNFM) • NFVO/VNFM can query compute resource information from the VIM (NFVO_CRM_QUERY or VNFM_CRM_QUERY) • NFVO/VNFM can query network resource information from the VIM (NFVO_NRM_QUERY or VNFM_NRM_QUERY) • NFVO/VNFM can query storage resource information from the VIM (NFVO_SRM_QUERY or VNFM_SRM_QUERY) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • EM can generate "Instantiate VNF" requests to the VNFM (EM_VNFLCM_VNF_INSTANTIATE) • VNFM supports "instantiate VNF" requests from the EM (VNFM_VNFLCM_VNF_INSTANTIATE) • Optionally, VNFM can generate "Grant VNF Lifecycle Operation" requests to the NFVO • (VNFM_VNFLCM_GRANTING) and NFVO supports "Grant VNF Lifecycle Operation" requests from the VNFM (NFVO_VNFLCM_GRANTING)
Pre-test conditions	• Any constraints required for the virtualised resource such as affinity/anti-affinity rules and resource locations are defined in the VNFD • The software image repository is reachable by the VIM • The required resources are available on the NFVI

Test Description: instantiate VNF with an EM request				
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Trigger an "instantiate VNF" operation on the EM	
	2	IOP Check	If VNFM_VNFLCM_GRANTING and NFVO_VNFLCM_GRANTING are supported, verify that the requested grant for the "instantiate VNF" operation has been approved by the NFVO	
	3	IOP Check	Verify that the software images have been successfully added to the image repository managed by the VIM	
	4	IOP Check	Verify that the requested virtualised resources have been allocated by the VIM according to the VNFD	
	5	IOP Check	Verify that virtualised resource allocation constraints have been met by querying the VIM	
	6	IOP Check	Verify that any existing virtualised resources have not been affected by the allocation of the new virtualised resources by querying the VIM	
	7	IOP Check	Verify that the VNF instance resources are visible on the VNFM	
	8	IOP Check	Verify that the VNF instance resources are visible on the NFVO	
	9	IOP Check	Verify that the VNF instance is reachable via the management network	
	10	IOP Check	Verify that the VNF instance has been configured according to the VNFD by querying the VNFM	
	11	IOP Check	Verify that the EM shows no "instantiate VNF" operation errors	
IOP Verdict				

7.4.2 Query VNF with an EM Request

Test Description: query VNF information with an EM request				
Identifier	TD_NFV_VNFLCM_QUERY_VNF_001			
Test Purpose	Verify that the VNF instance's information can be queried successfully by the EM			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 008 [i.6]			
Applicability	• EM can generate "query VNF" requests to the VNFM (EM_VNFLCM_VNF_QUERY) • VNFM supports "query VNF" requests from the EM (VNFM_VNFLCM_VNF_QUERY)			
Pre-test conditions	• The VNF is instantiated (TD_NFV_VNFLCM_INSTANTIATE_VNF_001)			
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Trigger the EM to query the VNF information using the unique VNF instance identifier from the VNFM	
	2	IOP Check	Verify that the EM has received the VNF information from the VNFM	
	3	IOP Check	Verify that VNF information in Step 2 matches the VNF information on the VNFM	
	4	IOP Check	Verify that the EM indicates the "query VNF" operation result as successful	
IOP Verdict				

7.4.3 Modify VNF Configuration Information with an EM Request

Test Description: update VNF with an EM request	
Identifier	TD_NFV_VNFLCM_MODIFY_VNF_INFO_001
Test Purpose	Verify that the VNF information can be updated successfully by the EM
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 008 [i.6]
Applicability	• EM can generate "modify VNF information" requests to the VNFM (EM_VNFLCM_VNF_MODIFY_INFO) • VNFM supports "modify VNF information" requests from the EM (VNFM_VNFLCM_VNF_MODIFY_INFO) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY)

Test Description: update VNF with an EM request				
Pre-test conditions	• The VNF is instantiated (TD_NFV_VNFLCM_INSTANTIATE_VNF_001) • The original VNF instance information is available for comparison (TD_NFV_VNFLCM_QUERY_VNF_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the EM to modify the VNF instance information on the VNFM	
	2	IOP Check	Verify that the VNF instance information has been updated on the VNFM	
	3	IOP Check	Verify that the VNF instance information has been updated by querying the NFVO	
	4	IOP Check	Verify that the EM indicates the "modify VNF information" operation result as successful	
IOP Verdict				

7.4.4 Start VNF/VNFC with an EM Request

Test Description: start VNF/VNFC with an EM request	
Identifier	TD_NFV_VNFLCM_START_VNF_001
Test Purpose	Verify that a VNF/VNFC instance be successfully started when an "operate VNF" operation is triggered by the EM
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 008 [i.6]
Applicability	• EM can generate "operate VNF" requests to the VNFM (EM_VNFLCM_VNF_OPERATE) • VNFM supports "operate VNF" requests from the EM (VNFM_VNFLCM_VNF_OPERATE) • NFVO/VNFM can generate "operate compute resource" operation requests to the VIM (NFVO_CRM_OPERATE or VNFM_CRM_OPERATE) • VIM supports "operate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_OPERATE_BY_NFVO or VIM_CRM_OPERATE_BY_VNFM) • NFVO/VNFM can query compute resource information from the VIM (NFVO_CRM_QUERY or VNFM_CRM_QUERY) • VIM can send compute resource information to the NFVO/VNFM (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO/VNFM (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO/VNFM (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY) • Optionally, VNFM can generate "Grant VNF Lifecycle Operation" requests to the NFVO (VNFM_VNFLCM_GRANTING) and NFVO supports "Grant VNF Lifecycle Operation" requests from the VNFM (NFVO_VNFLCM_GRANTING)
Pre-test conditions	• The VNF is instantiated (TD_NFV_VNFLCM_INSTANTIATE_VNF_001) • Target VNF/VNFC instance is in a "stopped" operational state. See note.

Test Description: start VNF/VNFC with an EM request				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the EM to start the target VNF/VNFC instance	
	2	IOP Check	If VNFM_VNFLCM_GRANTING and NFVO_VNFLCM_GRANTING are supported, verify that the requested grant for the "operate VNF" operation has been approved by the NFVO	
	3	IOP Check	Verify that the compute resources allocated to the target VNF/VNFC instance have been started by querying the VIM	
	4	IOP Check	Verify that other existing compute resources have not been affected by the performed operation by querying the VIM	
	5	IOP Check	Verify that the VNF/VNFC instance operational state on the VNFM is indicated as "started"	
	6	IOP Check	Verify that the EM indicates the "operate VNF" operation result as successful	
	7	IOP Check	Verify that the VNF/VNFC instance operates successfully by running the end-to-end functional test	
IOP Verdict				
NOTE: In the state STOPPED, the virtualised container(s), where the VNFC instance(s) of the VNF run, are shut down but not terminated.				

7.4.5 Stop VNF/VNFC with an EM Request

Test Description: stop VNF/VNFC with an EM request	
Identifier	TD_NFV_VNFLCM_STOP_VNF_001
Test Purpose	Verify that a VNF/VNFC instance be successfully stopped when an "operate VNF" operation is triggered by the EM
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 008 [i.6]
Applicability	• EM can generate "operate VNF" requests to the VNFM (EM_VNFLCM_VNF_OPERATE) • VNFM supports "operate VNF" requests from the EM (VNFM_VNFLCM_VNF_OPERATE) • NFVO/VNFM can generate "operate compute resource" operation requests to the VIM (NFVO_CRM_OPERATE or VNFM_CRM_OPERATE) • VIM supports "operate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_OPERATE_BY_NFVO or VIM_CRM_OPERATE_BY_VNFM) • NFVO/VNFM can query compute resource information from the VIM (NFVO_CRM_QUERY or VNFM_CRM_QUERY) • VIM can send compute resource information to the NFVO/VNFM (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO/VNFM (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO/VNFM (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY) • Optionally, VNFM can generate "Grant VNF Lifecycle Operation" requests to the NFVO (VNFM_VNFLCM_GRANTING) and NFVO supports "Grant VNF Lifecycle Operation" requests from the VNFM (NFVO_VNFLCM_GRANTING)
Pre-test conditions	• The VNF is instantiated (TD_NFV_VNFLCM_INSTANTIATE_VNF_001) • Target VNF/VNFC instance is in a "started" operational state

Test Description: stop VNF/VNFC with an EM request				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the EM to stop the target VNF/VNFC instance	
	2	IOP Check	If VNFM_VNFLCM_GRANTING and NFVO_VNFLCM_GRANTING are supported, verify that the requested grant for the "operate VNF" operation has been approved by the NFVO	
	3	IOP Check	Verify that the compute resources allocated to the target VNF/VNFC instance have been stopped by querying the VIM. See note.	
	4	IOP Check	Verify that other existing compute resources have not been affected by the performed operation by querying the VIM	
	5	IOP Check	Verify that the VNF/VNFC instance operational state on the VNFM is indicated as "stopped"	
	6	IOP Check	Verify that the EM indicates the "operate VNF" operation result as successful	
IOP Verdict				
NOTE: In the state STOPPED, the virtualised container(s), where the VNFC instance(s) of the VNF run, are shut down but not terminated.				

7.4.6 VNF Healing with an EM/VNF Request

Test Description: VNF healing with an EM/VNF Request	
Identifier	TD_NFV_VNFLCM_HEAL_VNF_001
Test Purpose	Verify that a VNF can be successfully healed when a VNF healing request is sent out by the EM/VNF
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • VNFM supports "heal VNF" requests from EM/VNF (VNFM_VNFLCM_VNF_HEAL) • EM/VNF can generate "heal VNF" requests to the VNFM (EM_VNFLCM_VNF_HEAL) • Optionally, VNFM can generate "Grant VNF Lifecycle Operation" requests to the NFVO (VNFM_VNFLCM_GRANTING) and NFVO supports "Grant VNF Lifecycle Operation" requests from the VNFM (NFVO_VNFLCM_GRANTING)
Pre-test conditions	• VNF is in a failed state (for example a virtualised resource needed by the impacted VNF instance has been terminated directly on the VIM)

Test Description: VNF healing with an EM/VNF Request				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the EM/VNF to send a VNF healing request to the VNFM	
	2	IOP Check	If VNFM_VNFLCM_GRANTING and NFVO_VNFLCM_GRANTING are supported, verify that the requested grant for the "heal VNF" operation has been approved by the NFVO	
	3	IOP Check	Verify that any additional resources required for the healing process have been allocated by the VIM according to the descriptors	
	4	IOP Check	Verify that healed VNF instance is running and reachable via the management network	
	5	IOP Check	Verify that the healed VNF instance has been configured according to the descriptors by querying the VNFM	
	6	IOP Check	Verify that any failed resources have been terminated and released by the VIM	
	7	IOP Check	Verify that the EM/VNF indicates the VNF healing operation result as successful	
	8	IOP Check	Verify that the NFVO indicates the VNF healing operation result as successful	
	9	IOP Check	Verify that the VNF has been healed by running an end-to-end functional test factoring in the functionality of the healed VNF instance	
IOP Verdict				

7.4.7 VNF Scale Out with an EM/VNF Request

Test Description: scale out VNF with an EM/VNF request	
Identifier	TD_NFV_VNFLCM_SCALE_OUT_VNF_001
Test Purpose	Verify that a VNF can be successfully scaled out by adding VNFC instances triggered by the EM/VNF
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • EM/VNF can generate "scale out by adding VNFC instances" requests to the VNFM (EM_VNFLCM_VNF_SCALE_OUT) • VNFM supports "scale out by adding VNFC instances" requests from the EM/VNF (VNFM_VNFLCM_VNF_SCALE_OUT) • Optionally, VNFM can generate "Grant VNF Lifecycle Operation" requests to the NFVO (VNFM_VNFLCM_GRANTING) and NFVO supports "Grant VNF Lifecycle Operation" requests from the VNFM (NFVO_VNFLCM_GRANTING)
Pre-test conditions	• VNFM is configured to trigger scale out when scale out request is received from EM/VNF • NFVI has the required amount of consumable virtual resources to run the scaled-out VNF • The current scale level of the VNF instance is not at the maximum scale level

Test Description: scale out VNF with an EM/VNF request				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the EM/VNF to send a "scale out by adding VNFC instance(s)" request to the VNFM	
	2	IOP Check	If VNFM_VNFLCM_GRANTING and NFVO_VNFLCM_GRANTING are supported, verify that the requested grant for the "VNF scale out" operation has been approved by the NFVO	
	3	IOP Check	Verify that the "scale out by adding VNFC instance(s)" procedure has been started on the NFVO	
	4	IOP Check	Verify that the additional VNFC instance(s) have been deployed by querying the VNFM	
	5	IOP Check	Verify that the additional resources have been allocated by the VIM according to the VNFD	
	6	IOP Check	Verify that the additional VNFC instance(s) are running and are reachable through their management network	
	7	IOP Check	Verify that the VNF configuration has been updated to include the additional VNFC instances according to the VNFD by querying the VNFM	
	8	IOP Check	Verify that the additional VNFC instances(s) are connected to the VL(s) according to the VNFD	
	9	IOP Check	Verify that the EM/VNF indicates the scaling operation result as successful	
	10	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	11	IOP Check	Verify that VNF has been scaled out by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.4.8 VNF Scale In with an EM/VNF Request

Test Description: VNF scale in with an EM/VNF request	
Identifier	TD_NFV_VNFLCM_SCALE_IN_VNF_001
Test Purpose	Verify that a VNF can be successfully scaled in by removing VNFC instances triggered by the EM/VNF
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • EM/VNF can generate "scale in by removing VNFC instances" to the VNFM (EM_VNFLCM_VNF_SCALE_IN) • VNFM supports "scale in by removing VNFC instances" requests from the EM/VNF (VNFM_VNFLCM_VNF_SCALE_IN) • Optionally, VNFM can generate "Grant VNF Lifecycle Operation" requests to the NFVO (VNFM_VNFLCM_GRANTING) and NFVO supports "Grant VNF Lifecycle Operation" requests from the VNFM (NFVO_VNFLCM_GRANTING)
Pre-test conditions	• The current NS deployment size allows scaling in • The current scale level of the VNF instance is not at the minimum scale level

Test Description: VNF scale in with an EM/VNF request				
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Trigger the EM/VNF to send a "scale in by removing VNFC instance(s)" request to the VNFM	
	2	IOP Check	If VNFM_VNFLCM_GRANTING and NFVO_VNFLCM_GRANTING are supported, verify that the requested grant for the "VNF scale in" operation has been approved by the NFVO	
	3	IOP Check	Verify that the "scale in by removing VNFC instance(s)" procedure has been started in NFVO	
	4	IOP Check	Verify that the impacted VNFC instance(s) inside the VNF have been terminated by querying the VNFM	
	5	IOP Check	Verify that the impacted VNFC instance(s) resources have been released by the VIM	
	6	IOP Check	Verify that the remaining VNFC instance(s) are still running and reachable via their management network	
	7	IOP Check	Verify that the VNF configuration has been updated to exclude the removed VNFC instances according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the remaining VNFC instances(s) and VL(s) are still connected according to the descriptors	
	9	IOP Check	Verify that the EM/VNF indicates the scaling operation result as successful	
	10	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	11	IOP Check	Verify that the VNF has been scaled in by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.4.9 Terminate VNF with an EM Request

Test Description: terminate VNF with an EM request	
Identifier	TD_NFV_VNFLCM_TERMINATE_VNF_001
Test Purpose	Verify that a VNF is successfully terminated when a "terminate VNF" operation is triggered by the EM
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 008 [i.6]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • NFVO/VNFM can query compute resource information from the VIM (NFVO_CRM_QUERY or VNFM_CRM_QUERY) • NFVO/VNFM can query network resource information from the VIM (NFVO_NRM_QUERY or VNFM_NRM_QUERY) • NFVO/VNFM can query storage resource information from the VIM (NFVO_SRM_QUERY or VNFM_SRM_QUERY) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • EM can generate "terminate VNF" requests to the VNFM (EM_VNFLCM_VNF_TERMINATE) • VNFM supports "terminate VNF" requests from the EM (VNFM_VNFLCM_VNF_TERMINATE)

Test Description: terminate VNF with an EM request				
	• Optionally, VNFM can generate "Grant VNF Lifecycle Operation" requests to the NFVO (VNFM_VNFLCM_GRANTING) and NFVO supports "Grant VNF Lifecycle Operation" requests from the VNFM (NFVO_VNFLCM_GRANTING)			
Pre-test conditions	• The target VNF is instantiated			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger an "terminate VNF" operation on the EM	
	2	IOP Check	If VNFM_VNFLCM_GRANTING and NFVO_VNFLCM_GRANTING are supported, verify that the requested grant for the "VNF terminate" operation has been approved by the NFVO	
	3	IOP Check	Verify that the impacted virtualised resources have been terminated by the VIM	
	4	IOP Check	Verify that other allocated virtualised resources have not been affected by the termination of the virtualised resources of the VNF instance by querying the VIM	
IOP Verdict				

7.5 Fault Management

7.5.1 Virtualised Resource Fault Management

7.5.1.1 Virtualised Resource Fault Alarm Notification

Test Description: virtualised resource fault alarm notification				
Identifier	TD_NFV_FM_VR_NOTIFY_001			
Test Purpose	Verify that a fault alarm notification propagates to the NFVO when a virtualised resource that is required for the NS connectivity fails			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO can subscribe to virtualised resource fault alarms on the VIM (NFVO_FM_VR_SUBSCRIBE) • VIM supports alarm notifications subscriptions from the NFVO (VIM_FM_SUBSCRIBE_BY_NFVO) • VIM can generate virtualised resources fault alarm notifications to the NFVO (VIM_FM_NOTIFY_BY_NFVO) • NFVO can process virtualised resource fault alarm notifications from the VIM (NFVO_FM_VR_NOTIFY) • NFVO can get the list of virtualised resource fault alarms from the VIM (NFVO_FM_VR_GET_ALARMS) • VIM supports virtualised resource fault alarm list queries by the NFVO (VIM_FM_QUERY_BY_NFVO)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVO is subscribed to virtualised resources fault alarms on the VIM			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a fault on a virtualised resource that is required for the NS instance connectivity on the NFVI (e.g. disable the NIC allocated to a network resource)	
	2	IOP Check	Verify that a virtualised resource fault alarm has been created on the VIM by querying the list of virtualised resource fault alarms	
	3	IOP Check	Verify that a NS fault alarm has been created on the NFVO by querying the list of NS fault alarms	
IOP Verdict				

7.5.1.2 Virtualised Resource Fault Alarm Clearance Notification

Test Description: virtualised resource fault alarm clearance notification				
Identifier	TD_NFV_FM_VR_CLEAR_001			
Test Purpose	Verify that a fault clearance notification propagates to the NFVO when a failed virtualised resource that is required for the NS connectivity is recovered			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO can subscribe to virtualised resource fault alarms on the VIM (NFVO_FM_VR_SUBSCRIBE) • VIM supports virtualised resource fault alarms subscriptions from the NFVO (VIM_FM_SUBSCRIBE_BY_NFVO) • VIM can generate virtualised resource fault alarm notifications to the NFVO (VIM_FM_NOTIFY_BY_NFVO) • NFVO can process virtualised resource fault alarm notifications from the VIM (NFVO_FM_VR_NOTIFY) • NFVO can get the list of virtualised resource fault alarms from the VIM (NFVO_FM_VR_GET_ALARMS) • VIM supports virtualised resource fault alarm list queries by the NFVO (VIM_FM_QUERY_BY_NFVO)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVO is subscribed to virtualised resources fault alarms on the VIM • NS fault alarm is created on the NFVO by failing a virtualised resource that is required for the NS connectivity (TD_NFV_FM_VR_NOTIFY_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Resolve the failure of the virtualised resource that is required for the NS instance connectivity (e.g. reconnect the NIC allocated to a network resource)	
	2	IOP Check	Verify that the relevant virtualised resource fault alarm has been cleared on the VIM by querying the list of virtualised resource fault alarms	
	3	IOP Check	Verify that the relevant NS fault alarm has been cleared on the NFVO by querying the list of NS fault alarms	
IOP Verdict				

7.5.2 VNF Fault Management

7.5.2.1 VNF Fault Alarm Notifications

Test Description: VNF fault alarm notification	
Identifier	TD_NFV_FM_VNF_NOTIFY_001
Test Purpose	Verify that a VNF fault alarm notification propagates via the VNFM to the NFVO when a VNF fault is triggered by a failed virtualised resource
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO can subscribe to alarm notifications on the VNFM (NFVO_FM_VNF_SUBSCRIBE) • VNFM supports alarm notifications subscriptions from the NFVO (VNFM_FM_VNF_SUBSCRIBE) • VNFM can subscribe to alarm notifications on the VIM (VNFM_FM_VR_SUBSCRIBE) • VIM supports alarm notifications subscriptions by the VNFM (VIM_FM_SUBSCRIBE_BY_VNFM) • VNFM can generate VNF fault alarm notifications to the NFVO (VNFM_FM_VNF_NOTIFY) • NFVO can process VNF fault alarm notifications from the VNFM (NFVO_FM_VNF_NOTIFY) • VIM can generate virtualised resources fault alarm notifications to the VNFM (VIM_FM_NOTIFY_BY_VNFM) • VNFM can process virtualised resource fault alarm notifications by the VIM (VNFM_FM_VR_NOTIFY) • NFVO can get the list of VNF fault alarms from the VNFM (NFVO_FM_VNF_GET_ALARMS) • VNFM supports VNF fault alarm list queries by the NFVO (VNFM_FM_VNF_QUERY) • VNFM can get the list of virtualised resource fault alarms from the VIM (VNFM_FM_VR_GET_ALARM)

Test Description: VNF fault alarm notification				
	• VIM supports virtualised resources fault alarm list queries by the VNFM (VIM_FM_QUERY_BY_VNFM)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVO is subscribed to VNF fault alarms on the VNFM • VNFM is subscribed to virtualised resources fault alarms on the VIM			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a failure on a virtualised resource allocated to the relevant VNF instance (e.g. terminate the virtualised resource directly on the VIM)	
	2	IOP Check	Verify that a virtualised resource fault alarm has been created on the VIM by querying the list of virtualised resource fault alarms	
	3	IOP Check	Verify that a VNF fault alarm has been created for the affected VNF instance on the VNFM by querying the list of VNF fault alarms	
	4	IOP Check	Verify that a NS fault alarm has been created on the NFVO by querying the list of NS fault alarms	
IOP Verdict				

7.5.2.2 VNF Fault Alarm Clearance Notifications

Test Description: VNF fault alarm clearance notification	
Identifier	TD_NFV_FM_VNF_CLEAR_001
Test Purpose	Verify that a VNF fault alarm clearance notification propagates via the VNFM to the NFVO when a VNF fault is cleared by resolving a failed virtualised resource
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO can subscribe to alarm notifications on the VNFM (NFVO_FM_VNF_SUBSCRIBE) • VNFM supports alarm notifications subscriptions from the NFVO (VNFM_FM_VNF_SUBSCRIBE) • VNFM can subscribe to alarm notifications on the VIM (VNFM_FM_VR_SUBSCRIBE) • VIM supports alarm notifications subscriptions by the VNFM (VIM_FM_SUBSCRIBE_BY_VNFM) • VNFM can generate VNF fault alarm clearance notifications to the NFVO (VNFM_FM_VNF_NOTIFY) • NFVO can process VNF fault alarm clearance notifications from the VNFM (NFVO_FM_VNF_NOTIFY) • VIM can generate virtualised resources fault alarm clearance notifications to the VNFM (VIM_FM_NOTIFY_BY_VNFM) • VNFM can process virtualised resource fault alarm clearance notifications from the VIM (VNFM_FM_VR_NOTIFY) • NFVO can get the list of VNF fault alarms from the VNFM (NFVO_FM_VNF_GET_ALARM) • VNFM supports VNF fault alarm list queries by the NFVO (VNFM_FM_VNF_QUERY) • VIM supports virtualised resources fault alarm list queries by the VNFM (VIM_FM_QUERY_BY_VNFM)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVO is subscribed to VNF fault alarms on the VNFM • VNFM is subscribed to virtualised resources fault alarms on the VIM • NS fault alarm is created on the NFVO by failing a virtualised resource that is allocated to the relevant VNF instance (TD_NFV_FM_VNF_NOTIFY_001)

Test Description: VNF fault alarm clearance notification				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Resolve the failure of the virtualised resource allocated to the relevant VNF (e.g. restart the virtualised resource directly on the VIM)	
	2	IOP Check	Verify that the relevant virtualised resource fault alarm has been cleared on the VIM by querying the list of virtualised resource fault alarms	
	3	IOP Check	Verify that the relevant VNF fault alarm has been cleared on the VNFM by querying the list of VNF fault alarms	
	4	IOP Check	Verify that the relevant NS fault alarm has been cleared on the NFVO by querying the list of NS fault alarms	
IOP Verdict				

7.5.3 NS Fault Management

7.5.3.1 NS Fault Alarm Subscription Creation

Test Description: NS Fault Alarm Subscription Creation				
Identifier	TD_NFV_FM_NS_ALARM_SUBSCRIPTION_CREATE_001			
Test Purpose	Verify that it is possible to subscribe to alarm notifications related to faults on the NS instance			
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO supports receiving VNF faults/alarms from VNFM (notifications) • VNFM exposes VNF alarms towards NFVO (notifications) • VNFM supports receiving virtualised resource faults/alarms from VIM			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Subscribe to the alarm notifications related to NS instance faults from NFVO (i.e. create new subscription resource)	
	2	IOP Check	Verify that a new subscription resource is created in the NFVO	
	3	IOP Check	Verify that the NFVO is subscribed on the VNFM to the related VNF resources faults alarm notifications	
	4	IOP Check	If applicable, verify that the VNFM is subscribed to the related VNF or virtualised resources faults alarm notifications	
IOP Verdict				

7.5.3.2 NS Fault Alarm Subscription Deletion

Test Description: NS Fault Alarm Subscription Deletion	
Identifier	TD_NFV_FM_NS_ALARM_SUBSCRIPTION_DELETE_001
Test Purpose	Verify that the subscription to NS faults alarm notifications can be deleted
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO supports receiving VNF faults/alarms from VNFM (notifications) • VNFM exposes VNF alarms towards NFVO (notifications) • VNFM supports receiving virtualised resource faults/alarms from VIM

Test Description: NS Fault Alarm Subscription Deletion				
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVO is subscribed to the NS faults alarm notifications (TD_NFV_FM_ALARM_SUBSCRIPTION_CREATE_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Delete the subscription to related to NS instance faults in the NFVO	
	2	IOP Check	Verify that the subscription related to NS instance fault alarm notifications is deleted in the NFVO	
	3	IOP Check	Verify that the NFVO deletes the subscription on the VNFM to VNF related resources faults alarm notifications	
	4	IOP Check	If applicable, verify that the VNFM in turn deletes its subscription to VNF or virtualised resource related resources faults alarm notifications	
IOP Verdict				

7.5.3.3 NS Fault Alarm Notification

Test Description: NS Fault Alarm Notification				
Identifier	TD_NFV_FM_NS_ALARM_NOTIFICATION_001			
Test Purpose	Verify that a fault alarm notification is exposed by the NFVO when a NS related resource fails			
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO supports receiving VNF faults/alarms from VNFM (notifications) • VNFM exposes VNF alarms towards NFVO (notifications) • VNFM supports receiving virtualised resource faults/alarms from VIM			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVO is subscribed to the NS faults alarm notifications (TD_NFV_FM_ALARM_SUBSCRIPTION_CREATE_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a fault on a NS related resource matching an existing alarm notification subscription	
	2	IOP Check	If applicable, verify that a VNF fault alarm notification has been received by the VNFM (e.g. by querying the NS fault alarms database)	
	3	IOP Check	Verify that a VNF fault alarm notification is generated and dispatched by the VNFM to the NFVO	
	4	IOP Check	Verify that an NS fault alarm notification is generated and dispatched by the NFVO (e.g. by checking the GUI where applicable)	
IOP Verdict				

7.5.3.4 NS Fault Alarm Notification Clearance

Test Description: NS Fault Alarm Notification Clearance				
Identifier	TD_NFV_FM_NS_ALARM_CLEAR_NOTIFICATION_001			
Test Purpose	Verify that an alarm clear notification is dispatched by the NFVO when a NS related resource failure is cleared			
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO supports receiving VNF faults/alarms from VNFM (notifications) • VNFM exposes VNF alarms towards NFVO (notifications) • VNFM supports receiving virtualised resource faults/alarms from VIM			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS fault alarm is created on the NFVO by failing a NS related resource (TD_NFV_FM_VR_ALARM_NOTIFY_001) • NFVO is subscribed to the NS faults alarm notifications (TD_NFV_FM_ALARM_SUBSCRIPTION_CREATE_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Clear the failure of the NS related resource that triggered the alarm notification	
	2	IOP Check	If applicable, verify that a VNF clear alarm notification has been received by the VNFM (e.g. by querying the NS fault alarms database)	
	3	IOP Check	Verify that a VNF clear alarm notification is generated and dispatched by the VNFM to the NFVO	
	4	IOP Check	Verify that an NS clear alarm notification is generated and dispatched by the NFVO (e.g. by checking the GUI where applicable)	
IOP Verdict				

7.5.3.5 NS Fault Alarm Query

Test Description: NS Fault Alarm Query	
Identifier	TD_NFV_FM_NS_ALARM_QUERY_001
Test Purpose	Verify that NS related fault alarms can be queried from the NFVO
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO supports receiving VNF faults/alarms from VNFM (query) • VNFM exposes VNF alarms towards NFVO (query response) • VNFM supports receiving virtualised resource faults/alarms from VIM
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)

Test Description: NS Fault Alarm Query				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a fault on a NS related resource	
	2	IOP Check	If applicable, verify that a VNF fault alarm has been collected by the VNFM (e.g. by querying its VNF fault alarms database)	
	3	IOP Check	Verify that the VNF fault alarms are queried by NFVO from the VNFM and are properly exposed as NS fault alarms upon explicit NS fault alarms query on the NFVO (e.g. by checking the GUI where applicable)	
IOP Verdict				

7.6 Performance Management

7.6.1 Virtualised Resource Performance Management

7.6.1.1 Virtualised Resource PM Job Creation and Notification Monitoring

Test Description: VR PM job creation and notification monitoring				
Identifier	TD_NFV_PM_VR_CREATE_NOTIFY_001			
Test Purpose	Verify that the performance metrics of a virtualised resource that is required for a NS instance connectivity can be monitored using PM jobs and notifications			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO can create VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the NFVO (VIM_PM_PMJOB_CREATE_BY_NFVO) • NFVO can subscribe to VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the NFVO (VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO) • VIM can generate VR PM notifications to the NFVO (VIM_PM_PMJOB_NOTIFY_BY_NFVO) • NFVO supports VR PM job notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY) • NFVO can query VR PM jobs from the VIM (NFVO_PM_VR_PMJOB_QUERY) • VIM supports VR PM job queries by the NFVO (VIM_PM_PMJOB_QUERY_BY_NFVO)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • Monitoring parameters (e.g. metrics, metric groups, collection and reporting periods) are defined			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the NFVO to create a VR PM job for a virtualised resource that is allocated to the target NS instance	
	2	IOP Check	Verify that a VR PM job has been created on the VIM according to the monitoring parameters by querying the VR PM jobs	
	3	Stimulus	Trigger the NFVO to subscribe to the VR PM job created in step 1	
	4	IOP Check	Verify that a "performance information available" notification for the monitored virtualised resource was generated by the VIM to the NFVO by monitoring the VR PM notifications. See note.	
IOP Verdict				
NOTE: Metric collection and reporting periods should be taken into account.				

7.6.1.2 Virtualised Resource PM Job Creation and Threshold Monitoring

Test Description: VR PM job creation and threshold monitoring				
Identifier	TD_NFV_PM_VR_CREATE_THRESHOLD_001			
Test Purpose	Verify that the performance metrics of a virtualised resource that is required for a NS instance connectivity can be monitored using PM jobs and thresholds			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO can create VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the NFVO (VIM_PM_PMJOB_CREATE_BY_NFVO) • NFVO can create VR PM thresholds on the VIM (NFVO_PM_VR_PMTH_CREATE) • VIM supports VR PM thresholds creation by the NFVO (VIM_PM_PMTH_CREATE_BY_NFVO) • VIM can generate VR PM notifications to the NFVO (VIM_PM_PMJOB_NOTIFY_BY_NFVO) • NFVO supports VR PM job notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY) • NFVO can query VR PM jobs from the VIM (NFVO_PM_VR_PMJOB_QUERY) • VIM supports VR PM job queries by the NFVO (VIM_PM_PMJOB_QUERY_BY_NFVO) • NFVO can query VR PM thresholds from the VIM (NFVO_PM_VR_PMTH_QUERY) • VIM supports VR PM threshold queries by the NFVO (VIM_PM_PMTH_QUERY_BY_NFVO)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • Monitoring parameters (e.g. metrics, metric groups, thresholds) are defined			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the NFVO to create a VR PM job for a virtualised resource that is allocated to the target NS instance	
	2	IOP Check	Verify that a VR PM job has been created on the VIM according to the monitoring parameters by querying the VR PM jobs	
	3	Stimulus	Trigger the NFVO to create a VR PM threshold for the virtualised resource monitored in step 1	
	4	Stimulus	Trigger the NFVO to subscribe to the threshold crossing notification for the VR PM threshold created in step 3	
	5	Stimulus	Trigger the virtualised resource to cross the specified threshold (e.g. by increasing resource utilization levels in the virtualisation container)	
	6	IOP Check	Verify that the "threshold crossed" notification for the monitored virtualised resource was generated by the VIM to the NFVO by monitoring the VR PM notifications	
IOP Verdict				

7.6.1.3 Virtualised Resource PM Job Deletion

Test Description: VR PM job deletion	
Identifier	TD_NFV_PM_VR_DELETE_MONITOR_001
Test Purpose	Verify that the monitoring of performance metrics of a virtualised resource that is required for a NS instance connectivity can be stopped by deleting PM jobs
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO can delete VR PM jobs from the VIM (NFVO_PM_VR_PMJOB_DELETE) • VIM supports VR PM jobs deletion by the NFVO (VIM_PM_PMJOB_DELETE_BY_NFVO) • NFVO can subscribe to VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the NFVO (VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO) • VIM can generate VR PM notifications to the NFVO (VIM_PM_PMJOB_NOTIFY_BY_NFVO) • NFVO supports VR PM job notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY) • NFVO can query VR PM jobs from the VIM (NFVO_PM_VR_PMJOB_QUERY) • VIM supports VR PM job queries by the NFVO (VIM_PM_PMJOB_QUERY_BY_NFVO)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • A virtualised resource that is required for the NS connectivity is monitored by the NFVO (TD_NFV_PM_VR_CREATE_NOTIFY_001)

Test Description: VR PM job deletion				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the NFVO to delete a VR PM job of a virtualised resource that is allocated to the target NS instance	
	2	IOP Check	Verify that the relevant VR PM job has been deleted on the VIM by querying the VR PM jobs	
	3	IOP Check	Verify that no "performance information available" notification for the monitored virtualised resource has been generated by the VIM to the NFVO by monitoring the VR PM notifications. See note.	
IOP Verdict				
NOTE: Virtualised resource metric collection and reporting periods should be taken into account.				

7.6.1.4 Virtualised Resource PM Threshold Deletion

Test Description: VR PM threshold deletion	
Identifier	TD_NFV_PM_VR_DELETE_THRESHOLD_001
Test Purpose	Verify that a threshold created for a virtualised resource that is required for a NS instance connectivity can be deleted
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO can delete VR PM thresholds from the VIM (NFVO_PM_VR_PMTH_DELETE) • VIM supports VR PM thresholds deletion by the NFVO (VIM_PM_PMTH_DELETE_BY_NFVO) • NFVO can subscribe to VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the NFVO (VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO) • VIM can generate VR PM notifications to the NFVO (VIM_PM_PMJOB_NOTIFY_BY_NFVO) • NFVO supports VR PM job notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY) • NFVO can query VR PM thresholds from the VIM (NFVO_PM_VR_PMTH_QUERY) • VIM supports VR PM threshold queries by the NFVO (VIM_PM_PMTH_QUERY_BY_NFVO)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • A threshold for a virtualised resource that is required for the NS connectivity is created by the NFVO (TD_NFV_PM_VR_CREATE_THRESHOLD_001)

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the NFVO to delete a VR PM threshold of a virtualised resource that is allocated to the target NS instance	
	2	IOP Check	Verify that the relevant VR PM threshold has been deleted on the VIM by querying the VR PM thresholds	
	3	Stimulus	Trigger the virtualised resource to cross the specified threshold (e.g. by increasing resource utilization levels in the virtualisation container)	
	4	IOP Check	Verify that no "threshold crossed" notification for the monitored virtualised resource has been generated by the VIM to the NFVO by monitoring the VR PM notifications. See note.	
IOP Verdict				
NOTE: Virtualised resource metric collection and reporting periods should be taken into account.				

7.6.2 VNF Performance Management

7.6.2.1 VNF PM Job Creation and Notification Monitoring

Test Description: VNF PM job creation and notification monitoring				
Identifier	TD_NFV_PM_VNF_CREATE_NOTIFY_001			
Test Purpose	Verify that the performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be monitored using PM jobs and notifications			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO can create VNF PM jobs on the VNFM (NFVO_PM_VNF_PMJOB_DELETE) • VNFM supports VNF PM jobs creation by the NFVO (VNFM_PM_VNF_PMJOB_CREATE) • VNFM can create VR PM jobs on the VIM (VNFM_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the VNFM (VIM_PM_PMJOB_CREATE_BY_VNFM) • NFVO can subscribe to VNF PM jobs on the VNFM (NFVO_PM_VNF_PMJOB_SUBSCRIBE) • VNFM supports VNF PM job subscriptions from the NFVO (VNFM_PM_VNF_PMJOB_SUBSCRIBE) • VNFM can subscribe to VR PM jobs on the VIM (VNFM_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM) • VIM can generate VR PM notifications to the VNFM (VIM_PM_PMJOB_NOTIFY_BY_VNFM) • VNFM supports VR PM job notifications from the VIM (VNFM_PM_VR_PMJOB_NOTIFY) • VNFM can generate VNF PM notifications to the NFVO (VNFM_PM_VNF_PMJOB_NOTIFY) • NFVO supports VNF PM job notifications from the VNFM (NFVO_PM_VNF_PMJOB_NOTIFY) • VNFM can query VR PM jobs from the VIM (VNFM_PM_VR_PMJOB_QUERY) • VIM supports VR PM job queries by the VNFM (VIM_PM_PMJOB_QUERY_BY_VNFM) • NFVO can query VNF PM jobs from the VNFM (NFVO_PM_VNF_PMJOB_QUERY) • VNFM supports VNF PM job queries by the NFVO (VNFM_PM_VNF_PMJOB_QUERY)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • Monitoring parameters (e.g. metrics, metric groups, collection and reporting periods) are defined on NFVO			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the NFVO to create a VNF PM job for a virtualised resource allocated to the relevant VNF instance inside the target NS instance	
	2	IOP Check	Verify that a VNF PM job has been created on the VNFM according to the monitoring parameters by querying the VNF PM jobs	
	3	IOP Check	Verify that a virtualised resource PM job has been created on the VIM according to the monitoring parameters by querying the VR PM jobs	
	4	Stimulus	Trigger the NFVO to subscribe to the VNF PM job created in step 1	
	5	IOP Check	Verify that a "performance information available" notification for the monitored virtualised resource was generated by the VIM to the VNFM by monitoring the VR PM notifications. See note.	
	6	IOP Check	Verify that a "performance information available" notification for the monitored virtualised resource was generated by the VNFM to the NFVO by monitoring the VNF PM notifications. See note.	
IOP Verdict				
NOTE: Metric collection and reporting periods should be taken into account.				

7.6.2.2 VNF PM Job Creation and Threshold Monitoring

Test Description: VNF PM job creation and threshold monitoring				
Identifier	TD_NFV_PM_VNF_CREATE_THRESHOLD_001			
Test Purpose	Verify that the performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be monitored using PM jobs and thresholds			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO can create VNF PM jobs on the VNFM (NFVO_PM_VNF_PMJOB_CREATE) • VNFM supports VNF PM jobs creation by the NFVO (VNFM_PM_VNF_PMJOB_CREATE) • VNFM can create VR PM jobs on the VIM (VNFM_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the VNFM (VIM_PM_PMJOB_CREATE_BY_VNFM) • NFVO can create VNF PM thresholds on the VNFM (NFVO_PM_VNF_PMTH_CREATE) • VNFM supports VNF PM thresholds creation by the NFVO (VNFM_PM_VNF_PMTH_CREATE) • VNFM can create VR PM thresholds on the VIM (VNFM_PM_VR_PMTH_CREATE) • VIM supports VR PM thresholds creation by the VNFM (VIM_PM_PMTH_CREATE_BY_VNFM) • NFVO can subscribe to VNF PM jobs on the VNFM (NFVO_PM_VNF_PMJOB_SUBSCRIBE) • VNFM supports VNF PM job subscriptions from the NFVO (VNFM_PM_VNF_PMJOB_SUBSCRIBE) • VNFM can subscribe to VR PM jobs on the VIM (VNFM_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM) • VIM can generate VR PM notifications to the VNFM (VIM_PM_PMJOB_NOTIFY_BY_VNFM) • VNFM supports VR PM job notifications from the VIM (VNFM_PM_VR_PMJOB_NOTIFY) • VNFM can generate VNF PM notifications to the NFVO (VNFM_PM_VNF_PMJOB_NOTIFY) • NFVO supports VNF PM job notifications from the VNFM (NFVO_PM_VNF_PMJOB_NOTIFY) • VNFM can query VR PM jobs from the VIM (VNFM_PM_VR_PMJOB_QUERY) • VIM supports VR PM job queries by the VNFM (VIM_PM_PMJOB_QUERY_BY_VNFM) • NFVO can query VNF PM jobs from the VNFM (NFVO_PM_VNF_PMJOB_QUERY) • VNFM supports VNF PM job queries by the NFVO (VNFM_PM_VNF_PMJOB_QUERY) • VNFM can query VR PM thresholds from the VIM (VNFM_PM_VR_PMTH_QUERY) • VIM supports VR PM threshold queries by the VNFM (VIM_PM_PMTH_QUERY_BY_VNFM) • NFVO can query VNF PM thresholds from the VNFM (NFVO_PM_VNF_PMTH_QUERY) • VNFM supports VNF PM threshold queries by the NFVO (VNFM_PM_VNF_PMTH_QUERY)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • Monitoring parameters (e.g. VNF instance, metrics, metric groups, threshold) are defined on NFVO			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the NFVO to create a PM job for a virtualised resource allocated to the relevant VNF instance inside the target NS instance	
	2	IOP Check	Verify that a VNF PM job has been created on the VNFM according to the monitoring parameters by querying the VNF PM jobs	
	3	IOP Check	Verify that a VR PM job has been created on the VIM according to the monitoring parameters by querying the VR PM jobs	
	4	Stimulus	Trigger the NFVO to create a VNF PM threshold for the virtualised resource monitored in step 1	
	5	Stimulus	Trigger the NFVO to subscribe to the threshold crossing notification for the VNF PM threshold created in step 4	
	6	Stimulus	Trigger the virtualised resource to cross the specified threshold (e.g. by increasing resource utilization levels in the virtualisation container)	
	7	IOP Check	Verify that the "threshold crossed" notification for the monitored virtualised resource was generated by the VIM to the VNFM by monitoring the VR PM notifications	
	8	IOP Check	Verify that the "threshold crossed" notification for the monitored virtualised resource was generated by the VNFM to the NFVO by monitoring the VNF PM notifications	
IOP Verdict				

7.6.2.3 VNF PM Job Deletion

<table>
<tr><th colspan="5">Test Description: VNF PM job deletion</th></tr>
<tr><td><b>Identifier</b></td><td colspan="4">TD_NFV_PM_VNF_DELETE_MONITOR_001</td></tr>
<tr><td><b>Test Purpose</b></td><td colspan="4">Verify that the monitoring of performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be stopped by deleting PM jobs</td></tr>
<tr><td><b>Configuration</b></td><td colspan="4">SUT Configuration 1</td></tr>
<tr><td><b>References</b></td><td colspan="4">ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]</td></tr>
<tr><td><b>Applicability</b></td><td colspan="4">• NFVO can delete VNF PM jobs from the VNFM (NFVO_PM_VNF_PMJOB_DELETE)
• VNFM supports VNF PM jobs deletion by the NFVO (VNFM_PM_VNF_PMJOB_DELETE)
• VNFM can delete VR PM jobs from the VIM (VNFM_PM_VR_PMJOB_DELETE)
• VIM supports VR PM jobs deletion by the VNFM (VIM_PM_PMJOB_DELETE_BY_VNFM)
• NFVO can delete VNF PM thresholds from the VNFM (NFVO_PM_VNF_PMTH_DELETE)
• VNFM supports VNF PM thresholds deletion by the NFVO (VNFM_PM_VNF_PMTH_DELETE)
• VNFM can delete VR PM thresholds from the VIM (VNFM_PM_VR_PMTH_DELETE)
• VIM supports VR PM thresholds deletion by the VNFM (VIM_PM_PMTH_DELETE_BY_VNFM)
• NFVO can subscribe to VNF PM jobs on the VNFM (NFVO_PM_VNF_PMJOB_SUBSCRIBE)
• VNFM supports VNF PM job subscriptions from the NFVO (VNFM_PM_VNF_PMJOB_SUBSCRIBE)
• VNFM can subscribe to VR PM jobs on the VIM (VNFM_PM_VR_PMJOB_SUBSCRIBE)
• VIM supports VR PM job subscriptions from the VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM)
• VIM can generate VR PM notifications to the VNFM (VIM_PM_PMJOB_NOTIFY_BY_VNFM)
• VNFM supports VR PM job notifications from the VIM (VNFM_PM_VR_PMJOB_NOTIFY)
• VNFM can generate PM notifications to the NFVO (VNFM_PM_VNF_PMJOB_NOTIFY)
• NFVO supports VNF PM job notifications from the VNFM (NFVO_PM_VNF_PMJOB_NOTIFY)
• VIM supports VR PM job queries by the VNFM (VIM_PM_PMJOB_QUERY_BY_VNFM)
• NFVO can query VNF PM jobs from the VNFM (NFVO_PM_VNF_PMJOB_QUERY)
• VNFM supports VNF PM job queries by the NFVO (VNFM_PM_VNF_PMJOB_QUERY)
• VNFM can query VR PM thresholds from the VIM (VNFM_PM_VR_PMTH_QUERY)
• VIM supports VR PM threshold queries by the VNFM (VIM_PM_PMTH_QUERY_BY_VNFM)
• NFVO can query VNF PM thresholds from the VNFM (NFVO_PM_VNF_PMTH_QUERY)
• VNFM supports VNF PM threshold queries by the NFVO (VNFM_PM_VNF_PMTH_QUERY)</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Pre-test conditions</b></td><td colspan="4">• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)
• A virtualised resource that is allocated to a VNF instance inside the target NS instance is monitored by the NFVO (TD_NFV_PM_VNF_CREATE_NOTIFY_001)</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Test Sequence</b></td><td><b>Step</b></td><td><b>Type</b></td><td><b>Description</b></td><td><b>Result</b></td></tr>
<tr><td></td><td>1</td><td>Stimulus</td><td>Trigger the NFVO to delete a PM job of a virtualised resource that is allocated to a VNF instance inside the target NS instance</td><td></td></tr>
<tr><td></td><td>2</td><td>IOP Check</td><td>Verify that the relevant VNF PM job has been deleted on the VNFM by querying the VNF PM jobs</td><td></td></tr>
<tr><td></td><td>3</td><td>IOP Check</td><td>Verify that the relevant VR PM job has been deleted on the VIM by querying the VR PM jobs</td><td></td></tr>
<tr><td></td><td>4</td><td>IOP Check</td><td>Verify that no "performance information available" notification for the monitored virtualised resource has been generated by the VIM to the VNFM by monitoring the VR PM notifications. See note.</td><td></td></tr>
<tr><td></td><td>5</td><td>IOP Check</td><td>Verify that no "performance information available" notification for the monitored virtualised resource has been generated by the VNFM to the NFVO by monitoring the VNF PM notifications. See note.</td><td></td></tr>
<tr><td><b>IOP Verdict</b></td><td colspan="3"></td><td></td></tr>
<tr><td colspan="5">NOTE: Virtualised resource metric collection and reporting periods should be taken into account.</td></tr>
</table>

7.6.2.4 VNF PM Threshold Deletion

<table>
<tr><th colspan="5">Test Description: VNF PM threshold deletion</th></tr>
<tr><td><b>Identifier</b></td><td colspan="4">TD_NFV_PM_VNF_DELETE_THRESHOLD_001</td></tr>
<tr><td><b>Test Purpose</b></td><td colspan="4">Verify that a threshold created for a virtualised resource that is allocated to a VNF instance inside a NS instance can be deleted</td></tr>
<tr><td><b>Configuration</b></td><td colspan="4">SUT Configuration 1</td></tr>
<tr><td><b>References</b></td><td colspan="4">ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]</td></tr>
<tr><td><b>Applicability</b></td><td colspan="4">• NFVO can delete VNF PM thresholds from the VNFM (NFVO_PM_VNF_PMTH_DELETE)
• VNFM supports VNF PM thresholds deletion by the NFVO (VNFM_PM_VNF_PMTH_DELETE)
• VNFM can delete VR PM thresholds from the VIM (VNFM_PM_VR_PMTH_DELETE)
• VIM supports VR PM thresholds deletion by the VNFM (VIM_PM_PMTH_DELETE_BY_VNFM)
• NFVO can subscribe to VNF PM jobs on the VNFM (NFVO_PM_VNF_PMJOB_SUBSCRIBE)
• VNFM supports VNF PM job subscriptions from the NFVO (VNFM_PM_VNF_PMJOB_SUBSCRIBE)
• VNFM can subscribe to VR PM jobs on the VIM (VNFM_PM_VR_PMJOB_SUBSCRIBE)
• VIM supports VR PM job subscriptions from the VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM)
• VIM can generate VR PM notifications to the VNFM (VIM_PM_PMJOB_NOTIFY_BY_VNFM)
• VNFM supports VR PM job notifications from the VIM (VNFM_PM_VR_PMJOB_NOTIFY)
• VNFM can query VR PM jobs from the VIM (VNFM_PM_VR_PMJOB_QUERY)
• VIM supports VR PM job queries by the VNFM (VIM_PM_PMJOB_QUERY_BY_VNFM)
• NFVO can query VNF PM jobs from the VNFM (NFVO_PM_VNF_PMJOB_QUERY)
• VNFM supports VNF PM job queries by the NFVO (VNFM_PM_VNF_PMJOB_QUERY)</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Pre-test conditions</b></td><td colspan="4">• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)
• A threshold for a virtualised resource that is allocated to a VNF instance inside a NS instance is created by the NFVO (TD_NFV_PM_VNF_CREATE_THRESHOLD_001)</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td rowspan="7"><b>Test Sequence</b></td><td><b>Step</b></td><td><b>Type</b></td><td><b>Description</b></td><td><b>Result</b></td></tr>
<tr><td>1</td><td>Stimulus</td><td>Trigger the NFVO to delete a VNF PM threshold of a virtualised resource that is allocated to a VNF instance inside the target NS instance</td><td></td></tr>
<tr><td>2</td><td>IOP Check</td><td>Verify that the relevant VNF PM threshold has been deleted on the VNFM by querying the VNF PM thresholds</td><td></td></tr>
<tr><td>3</td><td>IOP Check</td><td>Verify that the relevant VR PM threshold has been deleted on the VIM by querying the VR PM thresholds</td><td></td></tr>
<tr><td>4</td><td>Stimulus</td><td>Trigger the virtualised resource to cross the specified threshold (e.g. by increasing resource utilization levels in the virtualisation container)</td><td></td></tr>
<tr><td>5</td><td>IOP Check</td><td>Verify that no "threshold crossed" notification for the monitored virtualised resource has been generated by the VNFM to the NFVO by monitoring the VNF PM notifications. See note.</td><td></td></tr>
<tr><td>6</td><td>IOP Check</td><td>Verify that no "threshold crossed" notification for the monitored virtualised resource has been generated by the VIM to the VNFM by monitoring the VR PM notifications. See note.</td><td></td></tr>
<tr><td><b>IOP Verdict</b></td><td colspan="3"></td><td></td></tr>
<tr><td colspan="5">NOTE: Virtualised resource metric collection and reporting periods should be taken into account.</td></tr>
</table>

7.6.3 NS Performance Management

7.6.3.1 NS PM Monitoring Job Creation

Test Description: NS PM Monitoring Job Creation				
Identifier	TD_NFV_PM_NS_MONITORING_JOB_CREATE_001			
Test Purpose	Verify that performance monitoring job for monitoring NS related metrics can be created to start monitoring NS performance metrics			
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (notifications) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (notifications)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • Monitoring parameters (e.g. performance metrics, metric groups) are defined in the NS descriptor (e.g. CPU usage, memory usage, etc.)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Create a performance monitoring job on the NFVO to start monitoring, with given collection and reporting periods, one or more performance metrics defined in the NS descriptor	
	2	IOP Check	If applicable, verify that the NFVO creates one or more performance monitoring jobs on the VNFM to collect the VNF related performance metrics	
	3	IOP Check	If applicable, verify that the VNFM in turn issues the creation of performance monitoring jobs to collect the VNF performance metrics as virtualised resource metrics	
	4	IOP Check	Verify that performance metrics are properly collected by the VNFM with the requested collection period	
	5	IOP Check	Verify that NS performance metrics are properly collected by the NFVO from the VNFM with the requested collection period, e.g. by querying performance metrics database (if any) or checking directly the Graphical User Interface (if applicable)	
IOP Verdict				

7.6.3.2 NS PM Performance Metrics Query

Test Description: NS PM Performance Metrics Query	
Identifier	TD_NFV_PM_NS_PERFORMANCE_METRICS_QUERY_001
Test Purpose	Verify that NS related performance monitoring metrics can be retrieved in the form of reports by querying the NFVO
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (query) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (query response)

Test Description: NS PM Performance Metrics Query				
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS performance monitoring job is created and NFVO is collecting performance metrics (TD_NFV_PM_NS_MONITORING_JOB_CREATE_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Issue the query of one or more performance monitoring reports related to NS performance metrics to the NFVO	
	2	IOP Check	If applicable, verify that the NFVO queries VNF related performance monitoring reports from the VNFM to retrieve the given NS related metrics	
	3	IOP Check	If applicable, verify that the VNFM queries virtualised resource related performance monitoring reports to retrieve the VNF related metrics	
	4	IOP Check	Verify that the performance metrics are properly retrieved from the VNFM and returned by the NFVO, e.g. checking directly the NFVO Graphical User Interface (if applicable)	
IOP Verdict				

7.6.3.3 NS PM Threshold Creation

Test Description: NS PM Threshold Creation				
Identifier	TD_NFV_PM_NS_THRESHOLD_CREATE_001			
Test Purpose	Verify that performance monitoring thresholds can be created for one or more NS related performance metrics on the NFVO			
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (notifications) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (notifications)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS performance monitoring job is created and NFVO is collecting performance metrics (TD_NFV_PM_NS_MONITORING_JOB_CREATE_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Create on the NFVO a performance monitoring threshold for a NS related monitored performance metric	
	2	IOP Check	If applicable, verify that the NFVO in creates one or more thresholds on the VNFM for VNF monitored performance metrics related to the given NS instance	
	3	IOP Check	If applicable, verify that the VNFM in turn creates one or more thresholds on virtualised resource monitored performance metrics related to the given VNF	
	4	IOP Check	Verify that the performance monitoring thresholds are properly created and maintained by the NFVO and the VNFM (e.g. checking the Graphical User Interface, if applicable)	
IOP Verdict				

7.6.3.4 NS PM Subscription Creation for Threshold Information

Test Description: NS PM Subscription Creation for Threshold Information				
Identifier	TD_NFV_PM_NS_SUBSCRIPTION_CREATE_001			
Test Purpose	Verify that it is possible to subscribe to NS performance metrics notifications related to threshold crossed notifications			
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (notifications) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (notifications)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS performance monitoring job is created and NFVO is collecting performance metrics (TD_NFV_PM_NS_THRESHOLD_CREATE_001, TD_NFV_PM_NS_MONITORING_JOB_CREATE_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Subscribe to the threshold crossed notifications related to a NS performance monitoring threshold available in the NFVO (i.e. create new subscription resource)	
	2	IOP Check	Verify that a new subscription resource is created in the NFVO	
	3	IOP Check	Verify that the NFVO is subscribed on the VNFM to the threshold crossed notifications related to VNF performance metrics for the given NS instance	
	4	IOP Check	If applicable, verify that the VNFM in turn subscribes to the threshold crossed notifications related to virtualised resource performance metrics for the given VNFs	
IOP Verdict				

7.6.3.5 NS PM Subscription Creation for Performance Information

Test Description: NS PM Subscription Creation for Performance Information	
Identifier	TD_NFV_PM_NS_SUBSCRIPTION_CREATE_002
Test Purpose	Verify that it is possible to subscribe to NS performance metrics notifications related to availability of performance information
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (notifications) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (notifications)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS performance monitoring job is created and NFVO is collecting performance metrics (TD_NFV_PM_NS_MONITORING_JOB_CREATE_001)

Test Description: NS PM Subscription Creation for Performance Information				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Subscribe to the performance information availability notifications related to a NS performance job in the NFVO (i.e. create new subscription resource)	
	2	IOP Check	Verify that a new subscription resource is created in the NFVO	
	3	IOP Check	Verify that the NFVO is subscribed on the VNFM to the threshold crossed notifications related to VNF performance metrics for the given NS instance	
	4	IOP Check	If applicable, verify that the VNFM in turn subscribes to the performance information availability related to virtualised resource performance metrics for the given VNFs	
IOP Verdict				

7.6.3.6 NS PM Threshold Notification

Test Description: NS PM Threshold Notification	
Identifier	TD_NFV_PM_NS_THRESHOLD_NOTIFICATION_001
Test Purpose	Verify that a threshold crossed notification is exposed by the NFVO when a NS performance metric crosses a previously created threshold
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (notifications) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (notifications)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS performance monitoring subscription is created to collect performance monitoring notifications related to thresholds crossed (TD_NFV_PM_NS_SUBSCRIPTION_CREATE_001)

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the monitored NS performance metric to cross the specified threshold	
	2	IOP Check	If applicable, verify that the VNFM receives and collects virtualised resource threshold crossed notifications for the impacted VNF	
	3	IOP Check	Verify that a "threshold crossed" notification for the monitored VNF performance metric is generated and dispatched by the VNFM	
	4	IOP Check	Verify that a "threshold crossed" notification for the monitored NS performance metric is collected from the VNFM and dispatched by the NFVO (e.g. checking the Graphical User Interface, if applicable)	
IOP Verdict				

7.6.3.7 NS PM Monitoring Information Notification

Test Description: NS PM Threshold Notification				
Identifier	TD_NFV_PM_NS_MONITORING_INFO_NOTIFICATION_001			
Test Purpose	Verify that a monitoring information availability notification is exposed by the NFVO when new and updated NS performance monitoring metrics are available			
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (notifications) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (notifications)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS performance monitoring subscription is created to collect performance monitoring notifications related to the availability of new monitoring information (TD_NFV_PM_NS_SUBSCRIPTION_CREATE_002)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	If applicable, verify that the VNFM receives and collects virtualised resource monitoring information availability notification for the impacted VNF	
	2	IOP Check	Verify that a monitoring information availability notification for the monitored VNF performance metric is generated and dispatched by the VNFM	
	3	IOP Check	Verify that a monitoring information availability notification for the monitored NS performance metric is collected from the VNFM and dispatched by the NFVO (e.g. checking the Graphical User Interface, if applicable)	
IOP Verdict				

7.6.3.8 NS PM Subscription Deletion

Test Description: NS PM Subscription Deletion	
Identifier	TD_NFV_PM_NS_SUBSCRIPTION_DELETE_001
Test Purpose	Verify that the subscription to NS performance monitoring notifications can be deleted
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (notifications) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (notifications)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVO is subscribed to the NS faults alarm notifications (TD_NFV_PM_NS_SUBSCRIPTION_CREATE_001 or TD_NFV_PM_NS_SUBSCRIPTION_CREATE_002)

Test Description: NS PM Subscription Deletion				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Delete the subscription related to NS performance monitoring notifications in the NFVO	
	2	IOP Check	Verify that the subscription related to NS performance monitoring notifications is deleted in the NFVO	
	3	IOP Check	Verify that the NFVO deletes on the VNFM the subscription to VNF related resources performance monitoring notifications	
	4	IOP Check	If applicable, verify that the VNFM in turn deletes the subscription to virtualised resource related resources performance monitoring notifications	
IOP Verdict				

7.6.3.9 NS PM Monitoring Job Deletion

Test Description: NS PM Monitoring Job Deletion	
Identifier	TD_NFV_PM_NS_MONITORING_JOB_DELETE_001
Test Purpose	Verify that performance monitoring job for monitoring NS related metrics can be deleted to stop monitoring NS performance metrics
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (notifications) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (notifications)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS performance monitoring job is created and NFVO is collecting performance metrics (TD_NFV_PM_NS_MONITORING_JOB_CREATE_001)

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Delete a performance monitoring job on the NFVO to stop monitoring one or more performance metrics	
	2	IOP Check	If applicable, verify that the NFVO deletes the related VNF performance monitoring jobs on the VNFM to stop monitoring the impacted VNF performance metrics	
	3	IOP Check	Verify that the NS and VNF performance metrics are no more collected by the NFVO and the VNFM, e.g. by querying the performance metrics database (if any) or checking directly the Graphical User Interface (if applicable)	
IOP Verdict				

7.6.3.10 NS PM Performance Metrics Threshold Deletion

Test Description: NS PM Performance Metrics Threshold Deletion				
Identifier	TD_NFV_PM_NS_THRESHOLD_DELETE_001			
Test Purpose	Verify that performance monitoring thresholds can be deleted for one or more NS related performance metrics on the NFVO			
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO supports receiving VNF performance metrics from VNFM (notifications) • VNFM supports receiving virtualised resource performance metrics from VIM • VNFM exposes VNF performance metrics towards NFVO (notifications)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS performance monitoring threshold is created in the NFVO (TD_NFV_PM_NS_THRESHOLD_CREATE_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Delete on the NFVO the performance monitoring threshold for the NS related monitored performance metric	
	2	IOP Check	If applicable, verify that the NFVO deletes the related thresholds on VNF monitored performance metrics in the VNFM	
	3	IOP Check	Verify that the NS and VNFM performance monitoring thresholds are properly deleted in the NFVO and VNFM (e.g. checking the Graphical User Interface, if applicable)	
IOP Verdict				

7.7 NS Lifecycle Management

7.7.1 NS Instantiation

7.7.1.1 Standalone NS Instantiation

Test Description: standalone NS instantiation	
Identifier	TD_NFV_NSLCM_INSTANTIATE_001
Test Purpose	To verify that a standalone NS can be successfully instantiated
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can query software image information from the VIM (NFVO_SWIM_QUERY_IM or VNFM_SWIM_QUERY_IM) • VIM supports software image information queries by the NFVO/VNFM (VIM_SWIM_QUERY_IM_BY_NFVO or VIM_SWIM_QUERY_IM_BY_VNFM)

Test Description: standalone NS instantiation				
	• NFVO/VNFM can query compute resource information from the VIM (NFVO_CRM_QUERY or VNFM_CRM_QUERY) • NFVO/VNFM can query network resource information from the VIM (NFVO_NRM_QUERY or VNFM_NRM_QUERY) • NFVO/VNFM can query storage resource information from the VIM (NFVO_SRM_QUERY or VNFM_SRM_QUERY) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY)			
Pre-test conditions	• NSD, its associated descriptors (VLD(s), VNFFGD(s)) and VNF Package(s) have been on-boarded to the NFVO • The software image repository is reachable by the VIM • The required resources are available on the NFVI			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger NS instantiation on the NFVO	
	2	IOP Check	Verify that the VNFM receives instantiation requests for the VNFs composing the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the VNFs composing the given NS in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the software images have been successfully added to the image repository managed by the VIM	
	5	IOP Check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the VNF instance(s) have been deployed according to the NSD (i.e. query the VIM and VNFM for VMs, VLs and CPs)	
	7	IOP Check	Verify that the VNF instance(s) are reachable via the management network	
	8	IOP Check	Verify that the VNF instance(s) have been configured according to the VNFD(s) by querying the VNFM	
	9	IOP Check	Verify that the VNF instance(s), VL(s) and VNFFG(s) have been connected according to the descriptors	
	10	IOP Check	Verify that the NFVO indicates NS instantiation operation result as successful	
	11	IOP Check	Verify that the NS is successfully instantiated by running the end-to-end functional test	
IOP Verdict				

7.7.1.2 Nested NS Instantiation

Test Description: nested NS instantiation				
Identifier	TD_NFV_NSLCM_INSTANTIATE_NEST_NS_001			
Test Purpose	To verify that a NS referencing an existing nested NS can be successfully instantiated			
Configuration	SUT Configuration 3			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can query software image information from the VIM (NFVO_SWIM_QUERY_IM or VNFM_SWIM_QUERY_IM) • VIM supports software image information queries by the NFVO/VNFM (VIM_SWIM_QUERY_IM_BY_NFVO or VIM_SWIM_QUERY_IM_BY_VNFM) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY)			
Pre-test conditions	• NSD2, its associated descriptors (VLD(s), VNFFGD(s)) and VNF Package(s) have been on-boarded to the NFVO • NSD2 references nested NSD1 • NS2 is not instantiated • NS1 is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • The software image repository is reachable by the VIM • The required resources are available on the NFVI			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger NS2 instantiation on the NFVO	
	2	IOP Check	Verify that the software images of the VNF(s) referenced in NSD2 have been successfully added to the image repository managed by the VIM	
	3	IOP Check	Verify that resources associated to NS2 have been allocated by the VIM according to the descriptors	
	4	IOP Check	Verify that the VNF instance(s) have been deployed according to the NSD (i.e. query the VIM and VNFM for VMs, VLs and CPs)	
	5	IOP Check	Verify that existing VNF instance(s) in NS1 are running and reachable via the management network	
	6	IOP Check	Verify that the VNF instance(s) in NS2 are running and reachable through the management network	
	7	IOP Check	Verify that the VNF instances(s) in NS2 have been configured according to the VNFD(s) by querying the VNFM	
	8	IOP Check	Verify that the VNF instance(s), VL(s) and VNFFG(s) in NS2 have been connected according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates NS2 instantiation operation result as successful	
	10	IOP Check	Verify that NS2 is successfully instantiated by running an end-to-end functional test re-using the functionality of VNF instance(s) inside NS1	
IOP Verdict				

7.7.1.3 Multi-Site NS Instantiation

Test Description: Multi-Site NS instantiation	
Identifier	TD_NFV_NSLCM_INSTANTIATE_MUTLISITE_001
Test Purpose	To verify that the NS can be successfully instantiated across different sites
Configuration	SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIMs (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) deployed in different sites • NFVO/VNFM can generate "allocate network resource" operation requests to the VIMs (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) deployed in different sites • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIMs (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) deployed in different sites • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can query software image information from the VIM (NFVO_SWIM_QUERY_IM or VNFM_SWIM_QUERY_IM) deployed in different sites • VIM supports software image information queries by the NFVO/VNFM (VIM_SWIM_QUERY_IM_BY_NFVO or VIM_SWIM_QUERY_IM_BY_VNFM) • NFVO/VNFM can query compute resource information from the VIMs (NFVO_CRM_QUERY or VNFM_CRM_QUERY) deployed in different sites • NFVO/VNFM can query network resource information from the VIMs (NFVO_NRM_QUERY or VNFM_NRM_QUERY) deployed in different sites • NFVO/VNFM can query storage resource information from the VIMs (NFVO_SRM_QUERY or VNFM_SRM_QUERY) deployed in different sites • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY) • NFVO/VNFM supports multi-site deployments
Pre-test conditions	• NSD, its associated descriptors (VLD(s), VNFFGD(s)) and VNF Package(s) have been on-boarded to the NFVO • The software image repository is reachable by the VIMs • The required resources are available on the NFVI

Test Description: Multi-Site NS instantiation				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger multi-site NS instantiation on the NFVO	
	2	IOP Check	Verify that the VNFM receives instantiation requests for the VNFs composing the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the VNFs composing the given NS in the VIMs If VNFM is in indirect mode: Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the software images have been successfully added to the image repository managed by the VIMs	
	5	IOP Check	Verify that the requested resources have been allocated by the VIMs according to the descriptors	
	6	IOP Check	Verify that the VNF instance(s) have been deployed according to the NSD (i.e. query the VIMs and VNFM for VMs, VLs and CPs)	
	7	IOP Check	Verify that the VNF instance(s), VL(s) and VNFFG(s) have been created according to the descriptors	
	8	IOP Check	Verify that the VNF(s) have been deployed according to the multi-site location constraints.	
	9	IOP Check	Verify that the VNF instance(s) are reachable via the management network.	
	10	IOP Check	Verify that the VNF instance(s) have been configured according to the VNFD(s) by querying the VNFM	
	11	IOP Check	Verify that the VNF instance(s), VL(s) and VNFFG(s) have been connected according to the descriptors	
	12	IOP Check	Verify that the VNF instance(s) have multi-site connectivity through the multi-site VL(s).	
	13	IOP Check	Verify that the NS is successfully instantiated by running the end-to-end functional test	
IOP Verdict				

7.7.1.4 SFC NS Instantiation

Test Description: SFC NS instantiation	
Identifier	TD_NFV_NSLCM_INSTANTIATE_SFC _001
Test Purpose	To verify that a NS with NSH based SFC can be successfully instantiated
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can query software image information from the VIM (NFVO_SWIM_QUERY_IM or VNFM_SWIM_QUERY_IM) • VIM supports software image information queries by the NFVO/VNFM (VIM_SWIM_QUERY_IM_BY_NFVO or VIM_SWIM_QUERY_IM_BY_VNFM) • NFVO/VNFM can query compute resource information from the VIM (NFVO_CRM_QUERY or VNFM_CRM_QUERY) • NFVO/VNFM can query network resource information from the VIM (NFVO_NRM_QUERY or VNFM_NRM_QUERY) • NFVO/VNFM can query storage resource information from the VIM (NFVO_SRM_QUERY or VNFM_SRM_QUERY) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY)
Pre-test conditions	• NSD, its associated descriptors (VLD(s), VNFFGD(s)) and VNF Package(s) have been on-boarded to the NFVO • The software image repository is reachable by the VIM • The required resources are available on the NFVI

Test Description: SFC NS instantiation				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger NS instantiation on the NFVO	
	2	IOP Check	Verify that the VNFM receives instantiation requests for the VNFs composing the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the VNFs composing the given NS in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the software images have been successfully added to the image repository managed by the VIM	
	5	IOP Check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the VNF instance(s) have been deployed according to the NSD (i.e. query the VIM and VNFM for VMs, VLs and CPs)	
	7	IOP Check	Verify that the VL and VNFFG instance(s) have been created according to the descriptors	
	8	IOP Check	Verify that the VNF(s) are running and reachable through the management network	
	9	IOP Check	Verify that the VNF(s) have been configured according to VNFD(s) (i.e. by obtaining a result from the management interface)	
	10	IOP Check	Verify that the VNF(s), VL(s) and VNFFG(s) have been connected according to the Descriptors	
	11	IOP Check	Verify that the NS is successfully instantiated by running the end-to-end functional test (NSH Traffic)	
IOP Verdict				

7.7.2 NS Scaling

7.7.2.1 NS Scale Out

7.7.2.1.1 NS Scale out with an Operator Action

Test Description: NS scale out with an operator action	
Identifier	TD_NFV_NSLCM_SCALE_OUT_001
Test Purpose	Verify that the NS can be successfully scaled out by adding VNF instances triggered by an operator action
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO can generate "scale out by adding VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_OUT) • VNFM supports "scale out by adding VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_OUT)

Test Description: NS scale out with an operator action				
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger NS scale out by adding VNF instances to the NS in NFVO with an operator action	
	2	IOP Check	Verify that the VNFM receives instantiation request for the additional VNF(s) to be deployed for the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the additional VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the additional VNF instance(s) have been deployed by querying the VNFM	
	5	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the additional VNF instance(s) are running and reachable via their management network	
	7	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.1.2 NS Scale out with a VNF Indicator notification

Test Description: NS scale out with a VNF indicator notification	
Identifier	TD_NFV_NSLCM_SCALE_OUT_002
Test Purpose	Verify that the NS can be successfully scaled out by adding VNF instances triggered automatically by a VNF indicator
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE) • EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE) • EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY) • VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY) • NFVO can generate "scale out by adding VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_OUT) • VNFM supports "scale out by adding VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_OUT)

Test Description: NS scale out with a VNF indicator notification				
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS • NFVO/VNFM is configured to trigger "scale out by adding VNF instances" when a given VNF indicator value crosses a certain threshold • VNF/EM is configured to trigger notification when VNF Indicator value crosses a certain threshold			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the EM/VNF to send the targeted VNF indicator to the VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale out by adding VNF instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the additional VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the additional VNF instance(s) have been deployed by querying the VNFM	
	5	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the additional VNF instance(s) are running and reachable via their management network	
	7	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.1.3 NS Scale out with a VIM KPI

Test Description: NS scale out with a VIM KPI	
Identifier	TD_NFV_NSLCM_SCALE_OUT_003
Test Purpose	Verify that the NS can be successfully scaled out by adding VNF instances triggered automatically by a VIM KPI
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can create VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_CREATE or VNFM_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the NFVO/VNFM (VIM_PM_PMJOB_CREATE_BY_NFVO or VIM_PM_PMJOB_CREATE_BY_VNFM) • NFVO/VNFM can create VR PM thresholds on the VIM (NFVO_PM_VR_PMTH_CREATE or VNFM_PM_VR_PMTH_CREATE)

Test Description: NS scale out with a VIM KPI				
	• VIM supports VR PM thresholds creation by the NFVO/VNFM (VIM_PM_PMTH_CREATE_BY_NFVO or VIM_PM_PMTH_CREATE_BY_VNFM) • NFVO/VNFM can subscribe to VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_SUBSCRIBE or VNFM_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the NFVO/VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO or VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM) • VIM can generate VR PM notifications to the NFVO/VNFM (VIM_PM_PMJOB_NOTIFY_BY_NFVO or VIM_PM_PMJOB_NOTIFY_BY_VNFM) • NFVO/VNFM supports VR PM notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY or VNFM_PM_VR_PMJOB_NOTIFY) • NFVO can generate "scale out by adding VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_OUT) • VNFM supports "scale out by adding VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_OUT)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS • NFVO/VNFM is configured to trigger "scale out by adding VNF instances" when a given VIM KPI value crosses a certain threshold • Virtualised resource PM job with threshold monitoring has been created (TD_NFV_PM_VR_CREATE_THRESHOLD_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the VIM to send the targeted KPI to the NFVO/VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale out by adding VNF instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the additional VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the additional VNF instance(s) have been deployed by querying the VNFM	
	5	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the additional VNF instance(s) are running and reachable via their management network	
	7	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.1.4 NS Scale out with a query to VNF Indicator by VNFM

Test Description: NS scale out with a query to VNF indicator by VNFM	
Identifier	TD_NFV_NSLCM_SCALE_OUT_004
Test Purpose	Verify that the NS can be successfully scaled out by adding VNF instances triggered automatically in VNFM by a querying VNF Indicator
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE) • EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE) • EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY) • VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY) • NFVO can generate "scale out by adding VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_OUT) • VNFM supports "scale out by adding VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_OUT)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS • NFVO/VNFM is configured to trigger "scale out by adding VNF instances" when a given VNF indicator value crosses a certain threshold

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	In the VNF, trigger the target VNF indicator to cross the configured auto-scaling threshold value for scale out operation	
	2	IOP Check	Verify that the "scale out by adding VNF instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the additional VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the additional VNF instance(s) have been deployed by querying the VNFM	
	5	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the additional VNF instance(s) are running and reachable via their management network	
	7	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.1.5 NS Scale out for Multi-Site with an Operator Action

Test Description: NS scale out for Multi-Site with an Operator Action				
Identifier	TD_NFV_NSLCM_SCALE_OUT_MULTISITE_001			
Test Purpose	Verify that a multi-site NS can be successfully scaled out by adding VNF instances triggered by an operator action			
Configuration	SUT Configuration 4			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIMs (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) deployed in different sites • NFVO/VNFM can generate "allocate network resource" operation requests to the VIMs (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) deployed in different sites • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIMs (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) deployed in different sites • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO can generate "scale out by adding VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_OUT) • VNFM supports "scale out by adding VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_OUT) • NFVO/VNFM supports multi-site deployments			
Pre-test conditions	• Multi-site NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_MULTISITE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger multi-site NS scale out by adding VNF instances to the NS in NFVO with an operator action	
	2	IOP Check	Verify that the VNFM receives instantiation request for the additional VNF(s) to be deployed for the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the additional VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	5	IOP Check	Verify that the additional VNF instance(s) have been deployed by querying the VNFM (VMs, VLs, CPs, etc.)	
	6	IOP Check	Verify that the additional resources have been allocated by the VIM according to the multi-site location constraints	
	7	IOP Check	Verify that the additional VNF instance(s) are running and reachable via their management network	
	8	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors by querying the VNFM	
	9	IOP Check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the descriptors	
	10	IOP Check	Verify that the additional VNF Instance(s) have multi-site connectivity through the multi-site VL(s)	
	11	IOP Check	Verify that multi-site NS has been scaled out by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.2 NS Scale In

7.7.2.2.1 NS Scale In with an Operator Action

<table>
<tr><th colspan="5">Test Description: NS scale in with an operator action</th></tr>
<tr><td><b>Identifier</b></td><td colspan="4">TD_NFV_NSLCM_SCALE_IN_001</td></tr>
<tr><td><b>Test Purpose</b></td><td colspan="4">Verify that the NS can be successfully scaled in by removing VNF instances triggered by an operator action</td></tr>
<tr><td><b>Configuration</b></td><td colspan="4">SUT Configuration 1</td></tr>
<tr><td><b>References</b></td><td colspan="4">ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]</td></tr>
<tr><td><b>Applicability</b></td><td colspan="4">• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE)
• NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE)
• NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE)
• VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM)
• VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM)
• VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM)
• NFVO can generate "scale in by removing VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_IN)
• VNFM supports "scale in by removing VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_IN)</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Pre-test conditions</b></td><td colspan="4">• NFVO supports triggering "scale in by removing VNF instances" with an operator's action
• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)
• The current NS deployment size supports scaling in</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td rowspan="10"><b>Test Sequence</b></td><td><b>Step</b></td><td><b>Type</b></td><td><b>Description</b></td><td><b>Result</b></td></tr>
<tr><td>1</td><td>Stimulus</td><td>Trigger NS scale in by removing VNF instances from the NS in NFVO with an operator action</td><td></td></tr>
<tr><td>2</td><td>IOP Check</td><td>Verify that the VNFM receives terminate request from the NFVO for the VNF(s) to be removed in the given NS</td><td></td></tr>
<tr><td>3</td><td>IOP Check</td><td>If VNFM is in direct mode:
• Verify that the VNFM is granted by the NFVO to delete the virtualised resources in use by the impacted VNFs in the VIM
If VNFM is in indirect mode:
• Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS</td><td></td></tr>
<tr><td>4</td><td>IOP Check</td><td>Verify that the impacted VNF instance(s) have been terminated by querying the VNFM</td><td></td></tr>
<tr><td>5</td><td>IOP Check</td><td>Verify that the impacted VNF related resources have been released by the VIM</td><td></td></tr>
<tr><td>6</td><td>IOP Check</td><td>Verify that the remaining VNF instances(s) are still running and reachable via their management network</td><td></td></tr>
<tr><td>7</td><td>IOP Check</td><td>Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors</td><td></td></tr>
<tr><td>8</td><td>IOP Check</td><td>Verify that the NFVO indicates the scaling operation result as successful</td><td></td></tr>
<tr><td>9</td><td>IOP Check</td><td>Verify that NS has been scaled in by running the end-to-end functional test factoring the VNF scale and capacity</td><td></td></tr>
<tr><td><b>IOP Verdict</b></td><td colspan="3"></td><td></td></tr>
</table>

7.7.2.2.2 NS Scale in with a VNF Indicator notification

Test Description: NS scale in with a VNF indicator notification	
Identifier	TD_NFV_NSLCM_SCALE_IN_002
Test Purpose	Verify that the NS can be successfully scaled in by removing VNF instances triggered automatically by a VNF indicator
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE) • EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE) • EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY) • VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY) • NFVO can subscribe to VNF indicators on the VNFM (NFVO_NSVNFINDI_SUBSCRIBE) • VNFM supports VNF indicator subscriptions from the NFVO (VNFM_NSVNFINDI_SUBSCRIBE) • VNFM can generate VNF indicator notifications to the NFVO (VNFM_NSVNFINDI_NOTIFY) • NFVO supports VNF indicator notifications from the VNFM (NFVO_NSVNFINDI_NOTIFY) • NFVO can generate "scale in by removing VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_IN) • VNFM supports "scale in by removing VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_IN)
Pre-test conditions	• NFVO is configured to trigger "scale in by removing VNF instances" when a given VNF indicator value crosses a certain threshold • NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • The current NS deployment size should support scaling in • VNF/EM is configured to trigger notification when VNF Indicator value crosses a certain threshold

Test Description: NS scale in with a VNF indicator notification				
Test	**Step**	**Type**	**Description**	**Result**
Sequence	1	Stimulus	Trigger the EM/VNF to send the targeted VNF indicator to the VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale in by removing VNF instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to delete the virtualised resources in use by the impacted VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the impacted VNF instance(s) have been terminated by querying the VNFM	
	5	IOP Check	Verify that the impacted VNF related resources have been released by the VIM	
	6	IOP Check	Verify that the remaining VNF instance(s) are still running and reachable via their management network	
	7	IOP Check	Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors	
	8	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	9	IOP Check	Verify that NS has been scaled in by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.2.3 NS Scale in with a VIM KPI

Test Description: NS scale in with a VIM KPI	
Identifier	TD_NFV_NSLCM_SCALE_IN_003
Test Purpose	Verify that the NS can be successfully scaled in by removing VNF instances triggered automatically by a VIM KPI
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • NFVO/VNFM can create VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_CREATE or VNFM_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the NFVO/VNFM (VIM_PM_PMJOB_CREATE_BY_NFVO or VIM_PM_PMJOB_CREATE_BY_VNFM) • NFVO/VNFM can create VR PM thresholds on the VIM (NFVO_PM_VR_PMTH_CREATE or VNFM_PM_VR_PMTH_CREATE) • VIM supports VR PM thresholds creation by the NFVO/VNFM (VIM_PM_PMTH_CREATE_BY_NFVO or VIM_PM_PMTH_CREATE_BY_VNFM) • NFVO/VNFM can subscribe to VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_SUBSCRIBE or VNFM_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the NFVO/VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO or VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM) • VIM can generate VR PM notifications to the NFVO/VNFM (VIM_PM_PMJOB_NOTIFY_BY_NFVO or VIM_PM_PMJOB_NOTIFY_BY_VNFM)

Test Description: NS scale in with a VIM KPI				
	• NFVO/VNFM supports VR PM notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY or VNFM_PM_VR_PMJOB_NOTIFY) • NFVO can generate "scale in by removing VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_IN) • VNFM supports "scale in by removing VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_IN)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • The current NS deployment size should support scaling in • NFVO/VNFM is configured to trigger "scale in by removing VNF instances" when a given VIM KPI value crosses a certain threshold • Virtualised resource PM job with threshold monitoring has been created (TD_NFV_PM_VR_CREATE_THRESHOLD_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the VIM to send the targeted KPI to the NFVO/VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale in by removing VNF instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to delete the virtualised resources in use by the impacted VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the impacted VNF instance(s) have been terminated by querying the VNFM	
	5	IOP Check	Verify that the impacted VNF related resources have been released by the VIM	
	6	IOP Check	Verify that the remaining VNF instance(s) are still running and reachable via their management network	
	7	IOP Check	Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors	
	8	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	9	IOP Check	Verify that NS has been scaled in by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.2.4 NS Scale in with a query to VNF Indicator by VNFM

Test Description: NS scale in with a query to VNF indicator by VNFM	
Identifier	TD_NFV_NSLCM_SCALE_OUT_004
Test Purpose	Verify that the NS can be successfully scaled in by adding VNF instances triggered automatically in VNFM by a querying VNF Indicator
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE)

Test Description: NS scale in with a query to VNF indicator by VNFM				
	• EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE) • EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY) • VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY) • NFVO can subscribe to VNF indicators on the VNFM (NFVO_NSVNFINDI_SUBSCRIBE) • VNFM supports VNF indicator subscriptions from the NFVO (VNFM_NSVNFINDI_SUBSCRIBE) • VNFM can generate VNF indicator notifications to the NFVO (VNFM_NSVNFINDI_NOTIFY) • NFVO supports VNF indicator notifications from the VNFM (NFVO_NSVNFINDI_NOTIFY) • NFVO can generate "scale in by removing VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_IN) • VNFM supports "scale in by removing VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_IN)			
Pre-test conditions	• NFVO is configured to trigger "scale in by removing VNF instances" when a given VNF indicator value crosses a certain threshold • NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • The current NS deployment size should support scaling in			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	In the VNF, trigger the target VNF indicator to cross the configured autoscaling threshold value for scale in operation	
	2	IOP Check	Verify that the "scale in by removing VNF instance(s)" procedure has been started in NFVO	
	3		If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to delete the virtualised resources in use by the impacted VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the impacted VNF instance(s) have been terminated by querying the VNFM	
	5	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the additional VNF instance(s) are running and reachable via their management network	
	7	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled in by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.2.5 NS Scale in for Multi-Site with an Operator Action

<table>
<tr><th colspan="5">Test Description: NS Scale in for Multi-Site with an Operator Action</th></tr>
<tr><td><b>Identifier</b></td><td colspan="4">TD_NFV_NSLCM_SCALE_IN_MULTISITE_001</td></tr>
<tr><td><b>Test Purpose</b></td><td colspan="4">Verify that the multi-site NS can be successfully scaled in by removing VNF instances triggered by an operator action</td></tr>
<tr><td><b>Configuration</b></td><td colspan="4">SUT Configuration 4</td></tr>
<tr><td><b>References</b></td><td colspan="4">ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]</td></tr>
<tr><td><b>Applicability</b></td><td colspan="4">• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIMs (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) deployed in different sites
• NFVO/VNFM can generate "terminate network resource" operation requests to the VIMs (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) deployed in different sites
• NFVO/VNFM can generate "terminate storage resource" operation requests to the VIMs (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) deployed in different sites
• VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM)
• VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM)
• VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM)
• NFVO can generate "scale in by removing VNF instances" requests to the VNFM (NFVO_VNFLCM_NS_SCALE_IN)
• VNFM supports "scale in by removing VNF instances" requests from the NFVO (VNFM_VNFLCM_NS_SCALE_IN)</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Pre-test conditions</b></td><td colspan="4">• NFVO supports triggering "scale in by removing VNF instances" with an operator's action
• Multi-site NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_MULTISITE_001)
• Multi-site NS has been scaled out by adding VNF instances (TD_NFV_NSLCM_SCALE_OUT_MULTISITE_001)
• The current NS deployment size supports scaling in</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Test Sequence</b></td><td><b>Step</b></td><td><b>Type</b></td><td><b>Description</b></td><td><b>Result</b></td></tr>
<tr><td></td><td>1</td><td>Stimulus</td><td>Trigger multi-site NS scale in by removing VNF instances from the NS in NFVO with an operator action</td><td></td></tr>
<tr><td></td><td>2</td><td>IOP Check</td><td>Verify that the VNFM receives from the NFVO scale in operation for the impacted VNF in the given NS</td><td></td></tr>
<tr><td></td><td>3</td><td>IOP Check</td><td>If VNFM is in direct mode:
• Verify that the VNFM is granted by the NFVO to delete the virtualised resources related to the impacted VNF
If VNFM is in indirect mode:
• Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS</td><td></td></tr>
<tr><td></td><td>4</td><td>IOP Check</td><td>Verify that the impacted VNF instance(s) have been terminated and not running in the correspondent VIM site / instance</td><td></td></tr>
<tr><td></td><td>5</td><td>IOP Check</td><td>Verify that the impacted VNF related resources have been released by the proper VIM site / VIM instance</td><td></td></tr>
<tr><td></td><td>6</td><td>IOP Check</td><td>Verify that the remaining VNF instances(s) are still running and reachable via their management network</td><td></td></tr>
<tr><td></td><td>7</td><td>IOP Check</td><td>Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors</td><td></td></tr>
<tr><td></td><td>8</td><td>IOP Check</td><td>Verify that the remaining VNF instance(s) have still multi-site connectivity through the multi-site VL(s)</td><td></td></tr>
<tr><td></td><td>9</td><td>IOP Check</td><td>Verify that multi-site NS has been scaled in by running the end-to-end functional test</td><td></td></tr>
<tr><td><b>IOP Verdict</b></td><td colspan="3"></td><td></td></tr>
</table>

7.7.2.3 NS VNF Scale Out

7.7.2.3.1 NS VNF Scale Out with an Operator Action

<table>
<tr><th colspan="5">Test Description: NS VNF scale out with an operator action</th></tr>
<tr><td><b>Identifier</b></td><td colspan="4">TD_NFV_NSLCM_SCALE_OUT_VNF_001</td></tr>
<tr><td><b>Test Purpose</b></td><td colspan="4">To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered by a NFVO operator</td></tr>
<tr><td><b>Configuration</b></td><td colspan="4">SUT Configuration 1</td></tr>
<tr><td><b>References</b></td><td colspan="4">ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]</td></tr>
<tr><td><b>Applicability</b></td><td colspan="4">• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE)
• NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE)
• NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE)
• VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM)
• VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM)
• VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM)
• NFVO can generate "scale out by adding VNFC instances" to the VNFM (NFVO_VNFLCM_VNF_SCALE_OUT)
• VNFM supports "scale out by adding VNFC instances" requests from the NFVO (VNFM_VNFLCM_VNF_SCALE_OUT)</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Pre-test conditions</b></td><td colspan="4">• NFVO supports triggering "scale out by adding VNFC instances" with an operator's action
• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)
• NFVI has the required amount of consumable virtual resources to run the scaled-out NS</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Test Sequence</b></td><td><b>Step</b></td><td><b>Type</b></td><td><b>Description</b></td><td><b>Result</b></td></tr>
<tr><td></td><td>1</td><td>Stimulus</td><td>Trigger NS scale out by adding VNFC instance(s) to a VNF in the NS in NFVO with an operator action</td><td></td></tr>
<tr><td></td><td>2</td><td>IOP Check</td><td>If VNFM is in direct mode:
• Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the VNFs composing the given NS in the VIM
If VNFM is in indirect mode:
• Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS</td><td></td></tr>
<tr><td></td><td>3</td><td>IOP Check</td><td>Verify that the additional VNFC instance(s) have been deployed for the VNF by querying the VNFM</td><td></td></tr>
<tr><td></td><td>4</td><td>IOP Check</td><td>Verify that the additional resources have been allocated by the VIM according to the descriptors</td><td></td></tr>
<tr><td></td><td>5</td><td>IOP Check</td><td>Verify that the additional VNFC instance(s) are running and reachable via the management network</td><td></td></tr>
<tr><td></td><td>6</td><td>IOP Check</td><td>Verify that the VNF configuration has been updated to include the additional VNFC instances according to the descriptors by querying the VNFM</td><td></td></tr>
<tr><td></td><td>7</td><td>IOP Check</td><td>Verify that the additional VNFC instances(s) are connected to the VL(s) according to the descriptors</td><td></td></tr>
<tr><td></td><td>8</td><td>IOP Check</td><td>Verify that the NFVO indicates the scaling operation result as successful</td><td></td></tr>
<tr><td></td><td>9</td><td>IOP Check</td><td>Verify that NS has been scaled out by running the end-to-end functional test in relevance to the VNF scale and capacity</td><td></td></tr>
<tr><td><b>IOP Verdict</b></td><td colspan="3"></td><td></td></tr>
</table>

7.7.2.3.2 NS VNF Scale Out with a VNF Indicator notification

Test Description: NS VNF scale out with a VNF indicator notification				
Identifier	TD_NFV_NSLCM_SCALE_OUT_VNF_002			
Test Purpose	To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered automatically by a VNF indicator			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE) • EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE) • EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY) • VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS • VNFM is configured to trigger "scale out by adding VNFC instances" when a given VNF indicator value crosses a certain threshold • VNF/EM is configured to trigger notification when VNF Indicator value crosses a certain threshold			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the EM/VNF to send the targeted VNF indicator to the VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale out by adding VNFC instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the VNFs composing the given NS in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the additional VNFC instance(s) have been deployed by querying the VNFM	
	5	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the additional VNFC instance(s) are running and are reachable through their management network	
	7	IOP Check	Verify that the VNF configuration has been updated to include the additional VNFC instances according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the additional VNFC instances(s) are connected to the VL(s) according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.3.3 NS VNF Scale Out with a VIM KPI

Test Description: NS VNF scale out with a VIM KPI	
Identifier	TD_NFV_NSLCM_SCALE_OUT_VNF_003
Test Purpose	To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered automatically by a VIM KPI
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can create VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_CREATE or VNFM_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the NFVO/VNFM (VIM_PM_PMJOB_CREATE_BY_NFVO or VIM_PM_PMJOB_CREATE_BY_VNFM) • NFVO/VNFM can create VR PM thresholds on the VIM (NFVO_PM_VR_PMTH_CREATE or VNFM_PM_VR_PMTH_CREATE) • VIM supports VR PM thresholds creation by the NFVO/VNFM (VIM_PM_PMTH_CREATE_BY_NFVO or VIM_PM_PMTH_CREATE_BY_VNFM) • NFVO/VNFM can subscribe to VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_SUBSCRIBE or VNFM_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the NFVO/VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO or VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM) • VIM can generate VR PM notifications to the NFVO/VNFM (VIM_PM_PMJOB_NOTIFY_BY_NFVO or VIM_PM_PMJOB_NOTIFY_BY_VNFM) • NFVO/VNFM supports VR PM notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY or VNFM_PM_VR_PMJOB_NOTIFY) • NFVO can generate "scale out by adding VNFC instances" to the VNFM (NFVO_VNFLCM_VNF_SCALE_OUT) • VNFM supports "scale out by adding VNFC instances" requests from the NFVO (VNFM_VNFLCM_VNF_SCALE_OUT)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS • NFVO/VNFM is configured to trigger "scale out by adding VNFC instances" when a given VIM KPI value crosses a certain threshold • Virtualised resource PM job with threshold monitoring has been created (TD_NFV_PM_VR_CREATE_THRESHOLD_001)

Test Description: NS VNF scale out with a VIM KPI				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the VIM to send the targeted KPI to the NFVO/VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale out by adding VNFC instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the VNFs composing the given NS in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the additional VNFC instance(s) have been deployed by querying the VNFM	
	5	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the additional VNFC instance(s) are running and are reachable through their management network	
	7	IOP Check	Verify that the VNF configuration has been updated to include the additional VNFC instances according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the additional VNFC instances(s) are connected to the VL(s) according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.3.4 NS VNF Scale Out with a query to VNF Indicator by VNFM

Test Description: NS VNF scale out with a query VNF indicator by VNFM	
Identifier	TD_NFV_NSLCM_SCALE_OUT_VNF_004
Test Purpose	To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered automatically in VNFM by querying a VNF indicator
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE) • EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE) • EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY) • VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS • VNFM is configured to trigger "scale out by adding VNFC instances" when a given VNF indicator value crosses a certain threshold

Test Description: NS VNF scale out with a query VNF indicator by VNFM				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	In the VNF, trigger the target VNF indicator to cross the configured autoscaling threshold value for scale out operation	
	2	IOP Check	Verify that the "scale out by adding VNFC instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for the VNFs composing the given NS in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the additional VNFC instance(s) have been deployed by querying the VNFM	
	5	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	6	IOP Check	Verify that the additional VNFC instance(s) are running and are reachable through their management network	
	7	IOP Check	Verify that the VNF configuration has been updated to include the additional VNFC instances according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the additional VNFC instances(s) are connected to the VL(s) according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.3.5 NS VNF Scale Out for Multi-Site with an Operator Action

Test Description: NS VNF scale out for Multi-Site with an Operator Action	
Identifier	TD_NFV_NSLCM_SCALE_OUT_VNF_MULTISITE_001
Test Purpose	To verify that a VNF in a multi-site NS can be successfully scaled out by adding VNFC instances when triggered by a NFVO operator
Configuration	SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIMs (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIMs (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIMs (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO can generate "scale out by adding VNFC instances" to the VNFM (NFVO_VNFLCM_VNF_SCALE_OUT) • VNFM supports "scale out by adding VNFC instances" requests from the NFVO (VNFM_VNFLCM_VNF_SCALE_OUT)
Pre-test conditions	• NFVO supports triggering "scale out by adding VNFC instances" with an operator's action • Multi-Site NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_MULTISITE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS • NFVO is configured to trigger SCALE OUT (by adding VNFC instances) when triggered by a NFVO operator

Test Description: NS VNF scale out for Multi-Site with an Operator Action				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger multi-site NS scale out by adding VNFC instance(s) to a VNF in the NS in NFVO with an operator action	
	2	IOP Check	Verify that the VNFM receives from the NFVO scale out request for the impacted VNF in the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for scaling the impacted VNF If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	5	IOP Check	Verify that the additional VNFC instance(s) have been deployed for the VNF by querying the VNFM	
	6	IOP Check	Verify that the additional VNFC instance(s) have been deployed according to the multi-site location constraints	
	7	IOP Check	Verify that the additional VNFC instance(s) are running and reachable via the management network	
	8	IOP Check	Verify that the additional VNFC instances(s) have been configured according to the descriptors (i.e. by getting a result through the management interface)	
	9	IOP Check	Verify that the additional VNFC instances(s), VL(s) and VNFFG(s) are connected according to the Descriptors	
	10	IOP Check	Verify that the additional VNFC instance(s) have multi-site connectivity through the multi-site VL(s)	
	11	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.4 NS VNF Scale In

7.7.2.4.1 NS VNF Scale In with an Operator Action

Test Description: NS VNF scale in with an operator action	
Identifier	TD_NFV_NSLCM_SCALE_IN_VNF_001
Test Purpose	Verify that a VNF in a NS can be successfully scaled in by removing VNFC instances from an existing VNF triggered by an operator action
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • NFVO can generate "scale in by removing VNFC instances" to the VNFM (NFVO_VNFLCM_VNF_SCALE_IN) • VNFM supports "scale in by removing VNFC instances" requests from the NFVO (VNFM_VNFLCM_VNF_SCALE_IN)

Test Description: NS VNF scale in with an operator action				
Pre-test conditions	• NFVO supports triggering scale in with an operator's action • NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • The current NS deployment size should support scaling in			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger NS scale in by removing VNFC instance(s) from a VNF in the NS in NFVO with an operator action	
	2	IOP Check	Verify that the VNFM receives from the NFVO scale in operation for the impacted VNF in the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for scaling the impacted VNF If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	2	IOP Check	Verify that the impacted VNFC instance(s) inside the VNF have been terminated by querying the VNFM	
	3	IOP Check	Verify that the impacted VNFC instance(s) resources have been released by the VIM	
	4	IOP Check	Verify that the remaining VNFC instance(s) are still running and reachable via their management network	
	5	IOP Check	Verify that the VNF configuration has been updated to exclude the removed VNFC instances according to the descriptors by querying the VNFM	
	6	IOP Check	Verify that the remaining VNFC instances(s) and VL(s) are still connected according to the descriptors	
	7	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	8	IOP Check	Verify that NS has been scaled in by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.4.2 NS VNF Scale In with a VNF Indicator notification

Test Description: NS VNF scale in with a VNF indicator notification	
Identifier	TD_NFV_NSLCM_SCALE_IN_VNF_002
Test Purpose	Verify that a VNF in a NS can be successfully scaled in by removing VNFC instances triggered automatically by a VNF indicator
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE) • EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE) • EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY) • VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY)

Test Description: NS VNF scale in with a VNF indicator notification				
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • The current NS deployment size should support scaling in • VNFM is configured to trigger "scale in by removing VNFC instances" when a given VNF indicator value crosses a certain threshold			
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Trigger the EM/VNF to send the targeted VNF indicator to the VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale in by removing VNFC instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for scaling the impacted VNF If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the impacted VNFC instance(s) inside the VNF have been terminated by querying the VNFM	
	5	IOP Check	Verify that the impacted VNFC instance(s) resources have been released by the VIM	
	6	IOP Check	Verify that the remaining VNFC instance(s) are still running and reachable via their management network	
	7	IOP Check	Verify that the VNF configuration has been updated to exclude the removed VNFC instances according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the remaining VNFC instances(s) and VL(s) are still connected according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled in by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.4.3 NS VNF Scale In with a VIM KPI

Test Description: NS scale in with a VIM KPI	
Identifier	TD_NFV_NSLCM_SCALE_IN_VNF_003
Test Purpose	Verify that a VNF in a NS can be successfully scaled in by removing VNFC instances triggered automatically by a VIM KPI
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • NFVO/VNFM can create VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_CREATE or VNFM_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the NFVO/VNFM (VIM_PM_PMJOB_CREATE_BY_NFVO or VIM_PM_PMJOB_CREATE_BY_VNFM) • NFVO/VNFM can subscribe to VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_SUBSCRIBE or VNFM_PM_VR_PMJOB_SUBSCRIBE)

Test Description: NS scale in with a VIM KPI				
	• VIM supports VR PM job subscriptions from the NFVO/VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO or VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM) • NFVO/VNFM can create VR PM thresholds on the VIM (NFVO_PM_VR_PMTH_CREATE or VNFM_PM_VR_PMTH_CREATE) • VIM supports VR PM thresholds creation by the NFVO/VNFM (VIM_PM_PMTH_CREATE_BY_NFVO or VIM_PM_PMTH_CREATE_BY_VNFM) • VIM can generate VR PM notifications to the NFVO/VNFM (VIM_PM_PMJOB_NOTIFY_BY_NFVO or VIM_PM_PMJOB_NOTIFY_BY_VNFM) • NFVO/VNFM supports VR PM notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY or VNFM_PM_VR_PMJOB_NOTIFY) • NFVO can generate "scale in by removing VNFC instances" to the VNFM (NFVO_VNFLCM_VNF_SCALE_IN) • VNFM supports "scale in by removing VNFC instances" requests from the NFVO (VNFM_VNFLCM_VNF_SCALE_IN)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • The current NS deployment size should support scaling in • NFVO/VNFM is configured to trigger "scale in by removing VNFC instances" when a given VIM KPI value crosses a certain threshold • Virtualised resource PM job with threshold monitoring has been created (TD_NFV_PM_VR_CREATE_THRESHOLD_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the VIM to send notifications of the targeted KPI to the NFVO/VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale in by removing VNFC instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for scaling the impacted VNF If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the impacted VNFC instance(s) inside the VNF have been terminated by querying the VNFM	
	5	IOP Check	Verify that the impacted VNFC instance(s) resources have been released by the VIM	
	6	IOP Check	Verify that the remaining VNFC instance(s) are still running and reachable via their management network	
	7	IOP Check	Verify that the VNF configuration has been updated to exclude the removed VNFC instances according to the descriptors by querying the VNFM	
	8	IOP Check	Verify that the remaining VNFC instances(s) and VL(s) are still connected according to the descriptors	
	9	IOP Check	Verify that the NFVO indicates the scaling operation result as successful	
	10	IOP Check	Verify that NS has been scaled in by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.4.4 NS VNF Scale In with a query to VNF indicator by VNFM

<table>
<tr><th colspan="5">Test Description: NS VNF scale in with a query to VNF indicator by VNFM</th></tr>
<tr><td><b>Identifier</b></td><td colspan="4">TD_NFV_NSLCM_SCALE_IN_VNF_004</td></tr>
<tr><td><b>Test Purpose</b></td><td colspan="4">Verify that a VNF in a NS can be successfully scaled in by removing VNFC instances triggered automatically in VNFM by querying a VNF indicator</td></tr>
<tr><td><b>Configuration</b></td><td colspan="4">SUT Configuration 1</td></tr>
<tr><td><b>References</b></td><td colspan="4">ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]</td></tr>
<tr><td><b>Applicability</b></td><td colspan="4">• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE)
• NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE)
• NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE)
• VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM)
• VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM)
• VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM)
• VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE)
• EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE)
• EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY)
• VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY)</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Pre-test conditions</b></td><td colspan="4">• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)
• The current NS deployment size should support scaling in
• VNFM is configured to trigger "scale in by removing VNFC instances" when a given VNF indicator value crosses a certain threshold</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td rowspan="11"><b>Test Sequence</b></td><td><b>Step</b></td><td><b>Type</b></td><td><b>Description</b></td><td><b>Result</b></td></tr>
<tr><td>1</td><td>Stimulus</td><td>In the VNF, trigger the target VNF indicator to cross the configured autoscaling threshold value for scale out operation</td><td></td></tr>
<tr><td>2</td><td>IOP Check</td><td>Verify that the "scale in by removing VNFC instance(s)" procedure has been started in NFVO</td><td></td></tr>
<tr><td>3</td><td>IOP Check</td><td>If VNFM is in direct mode:
• Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for scaling the impacted VNF
If VNFM is in indirect mode:
• Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS</td><td></td></tr>
<tr><td>4</td><td>IOP Check</td><td>Verify that the impacted VNFC instance(s) inside the VNF have been terminated by querying the VNFM</td><td></td></tr>
<tr><td>5</td><td>IOP Check</td><td>Verify that the impacted VNFC instance(s) resources have been released by the VIM</td><td></td></tr>
<tr><td>6</td><td>IOP Check</td><td>Verify that the remaining VNFC instance(s) are still running and reachable via their management network</td><td></td></tr>
<tr><td>7</td><td>IOP Check</td><td>Verify that the VNF configuration has been updated to exclude the removed VNFC instances according to the descriptors by querying the VNFM</td><td></td></tr>
<tr><td>8</td><td>IOP Check</td><td>Verify that the remaining VNFC instances(s) and VL(s) are still connected according to the descriptors</td><td></td></tr>
<tr><td>9</td><td>IOP Check</td><td>Verify that the NFVO indicates the scaling operation result as successful</td><td></td></tr>
<tr><td>10</td><td>IOP Check</td><td>Verify that NS has been scaled in by running the end-to-end functional test in relevance to the VNF scale and capacity</td><td></td></tr>
<tr><td><b>IOP Verdict</b></td><td colspan="3"></td><td></td></tr>
</table>

7.7.2.4.5 NS VNF Scale In for Multi-Site with an Operator Action

Test Description: NS VNF scale in for Multi-Site with an Operator Action	
Identifier	TD_NFV_NSLCM_SCALE_IN_VNF_MULTISITE_001
Test Purpose	To verify that a VNF in a multi-site NS can be successfully scaled in by removing VNFC instances from an existing VNF triggered by an operator action
Configuration	SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIMs (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) deployed in different sites • NFVO/VNFM can generate "terminate network resource" operation requests to the VIMs (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) deployed in different sites • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIMs (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) deployed in different sites • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • NFVO can generate "scale in by removing VNFC instances" to the VNFM (NFVO_VNFLCM_VNF_SCALE_IN) • VNFM supports "scale in by removing VNFC instances" requests from the NFVO (VNFM_VNFLCM_VNF_SCALE_IN) • NFVO/VNFM supports multi-site deployments
Pre-test conditions	• NFVO supports triggering scale in with an operator's action • Multi-site NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_MULTISITE_001) • The current NS deployment size should support scaling in NS • Multi-site NS has been scaled out by adding VNFC instances (TD_NFV_NSLCM_SCALE_OUT_VNF_MULTISITE_001)

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger NS scale in by removing VNFC instance(s) from a VNF in the NS in NFVO with an operator action	
	2	IOP Check	Verify that the VNFM receives terminate request from the NFVO for the VNF(s) to be removed in the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate the virtualised resources required for scaling the impacted VNF If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the impacted VNFC instance(s) have been terminated and not running in the correspondent VIM site / instance	
	5	IOP Check	Verify that the impacted VNFC instance(s) resources have been released by the VIM site / VIM instance	
	6	IOP Check	Verify that the remaining VNFC instance(s) are still running and reachable via their management network	
	7	IOP Check	Verify that the remaining VNFC instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors	
	8	IOP Check	Verify that the remaining VNFC instance(s) have still multi-site connectivity through the multi-site VL(s)	
	9	IOP Check	Verify that multi-site NS has been scaled in by running the end-to-end functional test	
IOP Verdict				

7.7.2.5 NS Scale to Level

7.7.2.5.1 NS Scale to Level with an Operator Action

<table>
<tr><th colspan="5">Test Description: NS Scale to Level with an Operator Action</th></tr>
<tr><td><b>Identifier</b></td><td colspan="4">TD_NFV_NSLCM_SCALE_TO_LEVEL_001</td></tr>
<tr><td><b>Test Purpose</b></td><td colspan="4">Verify that the NS can be successfully scaled to another existing instantiation level by an operator action</td></tr>
<tr><td><b>Configuration</b></td><td colspan="4">SUT Configuration 1
SUT Configuration 3
SUT Configuration 4</td></tr>
<tr><td><b>References</b></td><td colspan="4">ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]</td></tr>
<tr><td><b>Applicability</b></td><td colspan="4">• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE)
• NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE)
• NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE)
• VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM)
• VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM)
• VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM)
• NFVO supports scale to level
• NS supports scale to level</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Pre-test conditions</b></td><td colspan="4">• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)
• The NS initial deployment size should support scaling to or from a specified level
• The Status of NS supports scale to level</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td rowspan="9"><b>Test Sequence</b></td><td><b>Step</b></td><td><b>Type</b></td><td><b>Description</b></td><td><b>Result</b></td></tr>
<tr><td>1</td><td>Stimulus</td><td>Trigger NS scale to another existing instantiation level in NFVO with an operator action</td><td></td></tr>
<tr><td>2</td><td>IOP Check</td><td>Verify that the VNFM receives instantiation or termination request (according to the target scale level) for the impacted VNF(s)</td><td></td></tr>
<tr><td>3</td><td>IOP Check</td><td>If VNFM is in direct mode:
• Verify that the VNFM is granted by the NFVO to allocate or delete the virtualised resources for the impacted VNFs in the VIM
If VNFM is in indirect mode:
• Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS</td><td></td></tr>
<tr><td>4</td><td>IOP Check</td><td>Verify that the all resources have been allocated or deleted in the VIM according to the descriptors</td><td></td></tr>
<tr><td>5</td><td>IOP Check</td><td>Verify that the all VNF instance(s) are running and reachable via their management network</td><td></td></tr>
<tr><td>6</td><td>IOP Check</td><td>Verify that the all VNF instances(s), VL(s) and VNFFG(s) are connected according to the descriptors</td><td></td></tr>
<tr><td>7</td><td>IOP Check</td><td>Verify in the NFVO that the NS has been scaled as requested (i.e. query or display the NS instance resource)</td><td></td></tr>
<tr><td>8</td><td>IOP Check</td><td>Verify that NS has been scaled to level by running the end-to-end functional test factoring the VNF scale and capacity</td><td></td></tr>
<tr><td><b>IOP Verdict</b></td><td colspan="3"></td><td></td></tr>
</table>

7.7.2.5.2 NS Scale to Level from VNF Indicator notification

Test Description: NS Scale to Level from VNF indicator notification	
Identifier	TD_NFV_NSLCM_SCALE_TO_LEVEL_002
Test Purpose	Verify that the NS can be successfully scaled to another existing instantiation level automatically by a VNF indicator notification
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE) • EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE) • EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY) • VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY) • NFVO supports scale to level • NS supports scale to level
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVO/VNFM is configured to trigger "scale NS to level" when a given VNF indicator value crosses a certain threshold • The NS initial deployment size should support scaling to or from a specified level • VNF/EM is configured to trigger notification when VNF Indicator value crosses a certain threshold • The Status of NS supports scale to level

Test Description: NS Scale to Level from VNF indicator notification				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the EM/VNF to send the targeted VNF indicator to the VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale NS to level by adding/removing VNF instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate or delete the virtualised resources for the impacted VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the virtualised resources have been allocated or deleted in the VIM according to the descriptors	
	5	IOP Check	Verify that the all VNF instance(s) are running and reachable via their management network	
	6	IOP Check	Verify that the all VNF instances(s), VL(s) and VNFFG(s) are connected according to the descriptors	
	7	IOP Check	Verify in the NFVO that the NS has been scaled as requested (i.e. query or display the NS instance resource)	
	8	IOP Check	Verify that NS has been scaled to level by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.5.3 NS Scale to Level from a VIM KPI

Test Description: NS scale to Level from a VIM KPI	
Identifier	TD_NFV_NSLCM_SCALE_TO_LEVEL_003
Test Purpose	Verify that the NS can be successfully scaled to another existing instantiation level automatically by a VIM KPI
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can create VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_CREATE or VNFM_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the NFVO/VNFM (VIM_PM_PMJOB_CREATE_BY_NFVO or VIM_PM_PMJOB_CREATE_BY_VNFM) • NFVO/VNFM can create VR PM thresholds on the VIM (NFVO_PM_VR_PMTH_CREATE or VNFM_PM_VR_PMTH_CREATE)

Test Description: NS scale to Level from a VIM KPI				
	• VIM supports VR PM thresholds creation by the NFVO/VNFM (VIM_PM_PMTH_CREATE_BY_NFVO or VIM_PM_PMTH_CREATE_BY_VNFM) • NFVO/VNFM can subscribe to VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_SUBSCRIBE or VNFM_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the NFVO/VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO or VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM) • VIM can generate VR PM notifications to the NFVO/VNFM (VIM_PM_PMJOB_NOTIFY_BY_NFVO or VIM_PM_PMJOB_NOTIFY_BY_VNFM) • NFVO/VNFM supports VR PM notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY or VNFM_PM_VR_PMJOB_NOTIFY) • NFVO supports scale to level • NS supports scale to level			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled-out NS • NFVO/VNFM is configured to trigger "scale NS to level" when a given VIM KPI value crosses a certain threshold • Virtualised resource PM job with threshold monitoring has been created (TD_NFV_PM_VR_CREATE_THRESHOLD_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the VIM to send the targeted KPI to the NFVO/VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale NS to level by adding/removing VNF instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate or delete the virtualised resources for the impacted VNFs in the VIM If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the virtualised resources have been allocated or deleted in the VIM according to the descriptors	
	5	IOP Check	Verify that the all VNF instance(s) are running and reachable via their management network	
	6	IOP Check	Verify that the all VNF instances(s), VL(s) and VNFFG(s) are connected according to the descriptors	
	7	IOP Check	Verify in the NFVO that the NS has been scaled as requested (i.e. query or display the NS instance resource	
	8	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test factoring the VNF scale and capacity	
IOP Verdict				

7.7.2.5.4 NS Scale to Level with an Operator Action for Multi-Site

<table>
<tr><th colspan="5">Test Description: NS scale to Level with an Operator Action for Multi-Site</th></tr>
<tr><td><b>Identifier</b></td><td colspan="4">TD_NFV_NSLCM_SCALE_TO_LEVEL_MULTISITE_001</td></tr>
<tr><td><b>Test Purpose</b></td><td colspan="4">Verify that a multi-site NS can be successfully scaled to another existing instantiation level (Scale NS to Level) by an operator</td></tr>
<tr><td><b>Configuration</b></td><td colspan="4">SUT Configuration 4</td></tr>
<tr><td><b>References</b></td><td colspan="4">ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]</td></tr>
<tr><td><b>Applicability</b></td><td colspan="4">• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIMs (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) deployed in different sites
• NFVO/VNFM can generate "allocate network resource" operation requests to the VIMs (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) deployed in different sites
• NFVO/VNFM can generate "allocate storage resource" operation requests to the VIMs (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) deployed in different sites
• VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM)
• VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM)
• VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM)
• NFVO can "scale to level by adding/removing VNF instances" requests to the VNFM
• VNFM supports "scale to level by adding/removing VNF instances" requests from the NFVO
• NFVO/VNFM supports multi-site deployments
• NFVO supports NS scale to level
• NS supports scale to level</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Pre-test conditions</b></td><td colspan="4">• Multi-site NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_MULTISITE_001)
• NFVI has the required amount of consumable virtual resources to run the scaled to level for NS
• The NS initial deployment size should support scaling to or from a specified level
• Current status of NS supports scale to level</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td><b>Test Sequence</b></td><th>Step</th><th>Type</th><th>Description</th><th>Result</th></tr>
<tr><td></td><td>1</td><td>Stimulus</td><td>Trigger multi-site NS scale to another existing instantiation level in NFVO with an operator request</td><td></td></tr>
<tr><td></td><td>2</td><td>IOP Check</td><td>Verify that the VNFM receives instantiation or termination request (according to the target scale level) for the impacted VNF(s)</td><td></td></tr>
<tr><td></td><td>3</td><td>IOP Check</td><td>If VNFM is in direct mode:
• Verify that the VNFM is granted by the NFVO to allocate or delete the virtualised resources for the impacted VNFs in the VIM
If VNFM is in indirect mode
• Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS</td><td></td></tr>
<tr><td></td><td>4</td><td>IOP Check</td><td>Verify that the virtualised resources have been allocated or deleted in the VIM according to the descriptors</td><td></td></tr>
<tr><td></td><td>5</td><td>IOP Check</td><td>Verify that all VNF instance(s) are running and reachable via the management network</td><td></td></tr>
<tr><td></td><td>6</td><td>IOP Check</td><td>Verify in the NFVO that the multi-site NS has been scaled as requested (i.e. query or display the NS instance resource)</td><td></td></tr>
<tr><td></td><td>7</td><td>IOP Check</td><td>Verify that NS is functional by running the end-to-end functional test</td><td></td></tr>
<tr><td><b>IOP Verdict</b></td><td colspan="3"></td><td></td></tr>
</table>

7.7.2.6 NS VNF Scale to Level

7.7.2.6.1 NS VNF Scale to Level with an Operator Action

Test Description: NS VNF Scale to Level with an operator action	
Identifier	TD_NFV_NSLCM_SCALE_TO_LEVEL_VNF_001
Test Purpose	To verify that a VNF in a NS can be successfully scaled to another existing instantiation level by a NFVO operator
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO supports scale to level • NS supports scale to level • NFVO supports NS scaling by adding/removing VNF instances • VNFM supports VNF scaling out/in by adding/removing VNFC instances
Pre-test conditions	• NFVO supports triggering "scale to level by adding/removing VNFC instances" with an operator's action • NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled NS to level

Test Description: NS VNF Scale to Level with an operator action				
Test	Step	Type	Description	Result
Sequence	1	Stimulus	Trigger NS scale by scaling to existing instantiation level to a VNF in the NS in NFVO with an operator action	
	2	IOP Check	Verify that the VNFM receives scale to level request for the impacted VNF(s) in the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate or delete (according to the target scale level) the virtualised resources for scaling the impacted VNF(s) If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	3	IOP Check	Verify that the all resources have been allocated or deleted in the VIM according to the descriptors by VNFM	
	4	IOP Check	Verify that the all VNFC instance(s) are running and reachable via the management network	
	6	IOP Check	Verify from the NFVO that the VNF in a NS has been successfully scaled (i.e. query or display the NS instance resource)	
	8	IOP Check	Verify that VNF has been scaled to level by running the NS end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.6.2 NS VNF Scale to Level from VNF Indicator notification

Test Description: NS VNF Scale to Level from VNF indicator notification	
Identifier	TD_NFV_NSLCM_SCALE_TO_LEVEL_VNF_002
Test Purpose	To verify that a VNF in a NS can be successfully scaled to another existing instantiation level by a VNF indicator notification
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • VNFM can subscribe to VNF indicators on the EM/VNF (VNFM_VNFINDI_SUBSCRIBE) • EM/VNF supports VNF indicator subscriptions from the VNFM (EM_VNFINDI_SUBSCRIBE) • EM/VNF can generate VNF indicator notifications to the VNFM (EM_VNFINDI_NOTIFY) • VNFM supports VNF indicator notifications from the EM/VNF (VNFM_VNFINDI_NOTIFY) • NFVO supports scale to level • NS supports scale to level

Test Description: NS VNF Scale to Level from VNF indicator notification				
	• NFVO supports NS scaling by adding/removing VNF instances • VNFM supports VNF scaling out/in by adding/removing VNFC instances			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the scaled to level NS • VNFM is configured to trigger "scale NS to level by adding/removing VNFC instances" when a given VNF indicator value crosses a certain threshold • VNF/EM is configured to trigger notification when VNF Indicator value crosses a certain threshold			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the EM/VNF to send the targeted VNF indicator to the VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale VNF(s) in NS to level by adding/removing VNF instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate or delete (according to the target scale level) the virtualised resources for scaling the impacted VNF(s) If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the all resources have been allocated or deleted in the VIM according to the descriptors by VNFM	
	5	IOP Check	Verify that the all VNFC instance(s) are running and are reachable through their management network	
	6	IOP Check	Verify from the NFVO that the VNF in a NS has been successfully scaled (i.e. query or display the NS instance resource)	
	7	IOP Check	Verify that VNF has been scaled to level by running the end-to-end NS functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.6.3 NS VNF Scale to Level with a VIM KPI

Test Description: NS VNF scale out with a VIM KPI	
Identifier	TD_NFV_NSLCM_SCALE_TO_LEVEL_VNF_003
Test Purpose	To verify that a VNF in a NS can be successfully scaled out by adding VNFC instances when triggered automatically by a VIM KPI
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can create VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_CREATE or VNFM_PM_VR_PMJOB_CREATE) • VIM supports VR PM jobs creation by the NFVO/VNFM (VIM_PM_PMJOB_CREATE_BY_NFVO or VIM_PM_PMJOB_CREATE_BY_VNFM) • NFVO/VNFM can create VR PM thresholds on the VIM (NFVO_PM_VR_PMTH_CREATE or VNFM_PM_VR_PMTH_CREATE) • VIM supports VR PM thresholds creation by the NFVO/VNFM (VIM_PM_PMTH_CREATE_BY_NFVO or VIM_PM_PMTH_CREATE_BY_VNFM) • NFVO/VNFM can subscribe to VR PM jobs on the VIM (NFVO_PM_VR_PMJOB_SUBSCRIBE or VNFM_PM_VR_PMJOB_SUBSCRIBE) • VIM supports VR PM job subscriptions from the NFVO/VNFM (VIM_PM_PMJOB_SUBSCRIBE_BY_NFVO or VIM_PM_PMJOB_SUBSCRIBE_BY_VNFM) • VIM can generate VR PM notifications to the NFVO/VNFM (VIM_PM_PMJOB_NOTIFY_BY_NFVO or VIM_PM_PMJOB_NOTIFY_BY_VNFM) • NFVO/VNFM supports VR PM notifications from the VIM (NFVO_PM_VR_PMJOB_NOTIFY or VNFM_PM_VR_PMJOB_NOTIFY) • NFVO can generate "scaled VNF to level by adding/removing the VNFC instances" to the VNFM • NFVO supports scale to level • NS supports scale to level • NFVO supports NS scaling by adding/removing VNF instances • VNFM supports VNF scaling out/in by adding/removing VNFC instances
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NFVI has the required amount of consumable virtual resources to run the run the scaled VNF to level • NFVO/VNFM is configured to trigger "scale VNF to level by adding/removing the VNFC instances" when a given VIM KPI value crosses a certain threshold • Virtualised resource PM job with threshold monitoring has been created (TD_NFV_PM_VR_CREATE_THRESHOLD_001)

Test Description: NS VNF scale out with a VIM KPI				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the VIM to send the targeted KPI to the NFVO/VNFM until the configured threshold is crossed	
	2	IOP Check	Verify that the "scale VNF(s) in NS to level by adding/removing VNF instance(s)" procedure has been started in NFVO	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate or delete (according to the target scale level) the virtualised resources for scaling the impacted VNF(s) If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the all resources have been allocated or deleted in the VIM according to the descriptors by VNFM	
	5	IOP Check	Verify that the all VNFC instance(s) are running and are reachable through their management network	
	6	IOP Check	Verify from the NFVO that the VNF in a NS has been successfully scaled (i.e. query or display the NS instance resource)	
	7	IOP Check	Verify that VNF has been scaled to level by running the end-to-end NS functional test in relevance to the VNF scale and capacity	
IOP Verdict				

7.7.2.6.4 NS VNF Scale to Level with an Operator Action for Multi-Site

Test Description: NS VNF scale to Level with an Operator Action for Multi-Site	
Identifier	TD_NFV_NSLCM_SCALE_TO_LEVEL_VNF_MULTISITE_001
Test Purpose	Verify that a VNF in a multi-site NS can be successfully scaled to another existing instantiation level (Scale VNF to Level) by an operator
Configuration	SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIMs (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIMs (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIMs (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO can generate "scale out by adding VNFC instances" to the VNFM (NFVO_VNFLCM_VNF_SCALE_OUT) • VNFM supports "scale out by adding VNFC instances" requests from the NFVO (VNFM_VNFLCM_VNF_SCALE_OUT) • NFVO can "scale to level by adding/removing VNF instances" requests to the VNFM • VNFM supports "scale to level by adding/removing VNF instances" requests from the NFVO • NFVO/VNFM supports multi-site deployments • NFVO supports NS scale to level • VNF(s) in the NS can scale to level

Test Description: NS VNF scale to Level with an Operator Action for Multi-Site				
Pre-test conditions	• Multi-Site NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_MULTISITE_001) • The NS initial deployment size should support scaling to a specified level • NS/VNF supports scale to level by adding/removing VNFC instances			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger multi-site NS scale by scaling to another existing instantiation level a VNF in the NS in NFVO with an operator request	
	2	IOP Check	Verify that the VNFM receives scale to level request for the impacted VNF(s) in the given NS	
	3	IOP Check	If VNFM is in direct mode: • Verify that the VNFM is granted by the NFVO to allocate or delete (according to the target scale level) the virtualised resources for scaling the impacted VNF(s) If VNFM is in indirect mode: • Verify that the VNFM sends resource allocation request using resource provider ID through the NFVO to manage the instantiation of the VNFs composing the given NS	
	4	IOP Check	Verify that the virtualised resources have been allocated or deleted in the VIM according to the descriptors by the VNFM	
	5	IOP Check	Verify that all VNFC instance(s) are running and reachable via the management network	
	6	IOP Check	Verify from the NFVO that the VNF in a multi-site NS has been successfully scaled (i.e. query or display the NS instance resource)	
	7	IOP Check	Verify that the NS is functional by running the end-to-end functional test	
IOP Verdict				

7.7.3 NS Update

7.7.3.1 Start VNF Instance

Test Description: start VNF instance	
Identifier	TD_NFV_NSLCM_UPDATE_START_001
Test Purpose	Verify the capability to start a VNF instance inside a NS instance
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO can generate "operate VNF" operation requests to the VNFM (NFVO_VNFLCM_OPERATE) • VNFM supports "operate VNF" operation requests from the NFVO (VNFM_VNFLCM_OPERATE) • NFVO/VNFM can generate "operate compute resource" operation requests to the VIM (NFVO_CRM_OPERATE or VNFM_CRM_OPERATE) • VIM supports "operate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_OPERATE_BY_NFVO or VIM_CRM_OPERATE_BY_VNFM) • NFVO/VNFM can query compute resource information from the VIM (NFVO_CRM_QUERY or VNFM_CRM_QUERY) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM)

Test Description: start VNF instance				
	• NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • Target VNF is in a stopped operational state. See note			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the NFVO to start the target VNF instance inside the NS instance	
	2	IOP Check	Verify that the compute resources allocated to the VNFC instances inside the target VNF instance have been started by querying the VIM	
	3	IOP Check	Verify that other existing compute resources have not been affected by the performed operation by querying the VIM	
	4	IOP Check	Verify that the VNF instance operational state on the VNFM is indicated as "started"	
	5	IOP Check	Verify that the NFVO shows no "operate VNF" operation errors	
	6	IOP Check	Verify that the NS functionality that utilizes the started VNF instance operates successfully by running the end-to-end functional test	
IOP Verdict				
NOTE: In the state STOPPED, the virtualised container(s), where the VNFC instance(s) of the VNF run, are shut down but not terminated.				

7.7.3.2 Stop VNF Instance

Test Description: stop VNF instance	
Identifier	TD_NFV_NSLCM_UPDATE_STOP_001
Test Purpose	Verify the capability to stop a VNF instance inside a NS instance
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO can generate "operate VNF" operation requests to the VNFM (NFVO_VNFLCM_OPERATE) • VNFM supports "operate VNF" operation requests from the NFVO (VNFM_VNFLCM_OPERATE) • NFVO/VNFM can generate "operate compute resource" operation requests to the VIM (NFVO_CRM_OPERATE or VNFM_CRM_OPERATE) • VIM supports "operate compute resource" operation requests from the NFVO/VNFM NFVO/VNFM (VIM_CRM_OPERATE_BY_NFVO or VIM_CRM_OPERATE_BY_VNFM) • NFVO/VNFM can query compute resource information from the VIM (NFVO_CRM_QUERY or VNFM_CRM_QUERY) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)

Test Description: stop VNF instance				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the NFVO to stop the target VNF instance inside the NS instance	
	2	IOP Check	Verify that the compute resources allocated to the VNFC instances inside the target VNF instance have been stopped by querying the VIM. See note	
	3	IOP Check	Verify that other existing compute resources have not been affected by the performed operation by querying the VIM	
	4	IOP Check	Verify that the VNF instance operational state on the VNFM is indicated as "stopped"	
	5	IOP Check	Verify that the NFVO shows no "operate VNF" operation errors	
IOP Verdict				
NOTE: In the state STOPPED, the virtualised container(s), where the VNFC instance(s) of the VNF run, are shut down but not terminated.				

7.7.3.3 Instantiate VNF and Add Instance to NS Instance

Test Description: NS update instantiating VNFs and adding instances				
Identifier	TD_NFV_NSLCM_UPDATE_INST_ADD_VNF_001			
Test Purpose	To verify that one or more VNFs can be instantiated and the instances added to a running NS instance			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO can generate "create VNF identifier" operation requests to the VNFM (NFVO_VNFLCM_CREATE_VNFID) • VNFM supports "create VNF identifier" operation requests from the NFVO (VNFM_VNFLCM_CREATE_VNFID) • NFVO can generate "instantiate VNF" operation requests to the VNFM (NFVO_VNFLCM_INSTANTIATE) • VNFM supports "instantiate VNF" operation requests from the NFVO (VNFM_VNFLCM_INSTANTIATE)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • VNF Packages whose VNFDs are referred to in the NSD are on-boarded to the NFVO			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a NS update instantiating VNF(s) and adding the resulting instance(s) to a running NS on the NFVO	
	2	IOP Check	Verify that the additional VNF instance(s) have been deployed by querying the VNFM	
	3	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	4	IOP Check	Verify that the additional VNF instance(s) are running and reachable via their management network	
	5	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors by querying the VNFM	
	6	IOP Check	Verify that the NFVO indicates the VNF(s) instantiation and instance(s) addition operation result as successful	
	7	IOP Check	Verify that NS has been updated by running the end-to-end functional test that includes the additional VNF instance(s)	
IOP Verdict				

7.7.3.4 Remove VNF Instances from a NS Instance

Test Description: NS update removing VNF instances				
Identifier	TD_NFV_NSLCM_UPDATE_REM_VNF_001			
Test Purpose	To verify that one or more VNF instances can be removed from a running NS instance			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • NFVO can generate "terminate VNF" operation requests to the VNFM (NFVO_VNFLCM_TERMINATE) • VNFM supports "terminate VNF" operation requests from the NFVO (VNFM_VNFLCM_TERMINATE) • NFVO can generate "delete VNF identifier" operation requests to the VNFM (NFVO_VNFLCM_DELETE_VNFID) • VNFM supports "delete VNF identifier" operation requests from the NFVO (VNFM_VNFLCM_DELETE_VNFID)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS can function without the impacted VNF instance(s) • Functional verification of the VNF instance(s) removal is possible (i.e. reduced capacity)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a NS update removing one or more VNF instances from a running NS instance on the NFVO	
	2	IOP Check	Verify that the impacted resources have been terminated by the VIM according to the descriptors	
	3	IOP Check	Verify that the remaining VNF instance(s) are running and reachable via their management network	
	4	IOP Check	NFVO indicates the update operation was successful	
	5	IOP Check	Verify that NS instance has been updated by running the end-to-end functional test factoring the removal of the VNF instance(s)	
IOP Verdict				

7.7.3.5 Add Shared VNF Instances to NS Instance

Test Description: NS update adding shared VNF instances	
Identifier	TD_NFV_NSLCM_UPDATE_ADD_SHVNF_001
Test Purpose	To verify that one or more shared VNF instances can be added to a running NS instance
Configuration	SUT Configuration 2
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM)

Test Description: NS update adding shared VNF instances				
	• VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM)			
Pre-test conditions	• NS1 is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS1 can utilize shared VNF instances • Functional verification of the additional shared VNF instances is possible (i.e. traffic load sharing)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a NS update adding one or more shared VNF instances to NS1 on the NFVO	
	2	IOP Check	Verify that any additional resources associated to NS1 have been allocated and deployed by the VIM according to the updated descriptors	
	3	IOP Check	Verify that the existing VNF instance(s) in NS1 are running and reachable through the management network	
	4	IOP Check	Verify that the additional shared VNF instances(s) have been configured according to the descriptors by querying the VNFM	
	5	IOP Check	Verify that the NFVO indicates the shared VNF instance(s) addition operation result as successful	
	6	IOP Check	Verify that NS1 has been updated by running the end-to-end functional test that includes the additional shared VNF instance(s)	
	7	IOP Check	Verify that NS2 instance was unaffected by the NS update operation by running the relevant end-to-end functional tests that include the shared VNF instance(s)	
IOP Verdict				

7.7.3.6 Remove Shared VNF Instances from NS Instance

Test Description: NS update removing shared VNF instances	
Identifier	TD_NFV_NSLCM_UPDATE_REM_SHVNF_001
Test Purpose	To verify that one or more shared VNF instances can be removed from a running NS instance
Configuration	SUT Configuration 2
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM)
Pre-test conditions	• NS1 is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS1 can function without the impacted VNF instances • Functional verification of the VNF instance(s) removal is possible (i.e. reduced capacity)

Test Description: NS update removing shared VNF instances				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a NS update removing one or more existing VNF instances from NS1 on the NFVO	
	2	IOP Check	Verify that any additional resources associated to NS1 have been removed by the VIM according to the updated descriptors	
	3	IOP Check	Verify that the existing VNF instance(s) in NS1 are still running and reachable through the management network	
	4	IOP Check	Verify that the previously shared VNF instances(s) have been configured according to the updated descriptors by querying the VNFM	
	5	IOP Check	Verify that the NFVO indicates the shared VNF instance(s) removal operation result as successful	
	6	IOP Check	Verify that NS1 has been updated by running the end-to-end functional test factoring the shared VNF instance(s) removal	
	7	IOP Check	Verify that NS2 instance was unaffected by the NS update operation by running the relevant end-to-end functional tests that include the previously shared VNF instance(s)	
IOP Verdict				

7.7.3.7 Change VNF Deployment Flavour

Test Description: NS update changing VNF instances DF	
Identifier	TD_NFV_NSLCM_UPDATE_VNF_DF_001
Test Purpose	To verify that the deployment flavour of one or more VNF instances in a NS instance can be changed
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS contains at least one VNF that accepts multiple deployment flavours • The new deployment flavour is expected to impact a functional aspect of the VNF (i.e. scale or performance)

Test Description: NS update changing VNF instances DF				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger a NS update changing the deployment flavour (DF) of one or more VNF instances in a NS instance on NFVO	
	2	IOP Check	Verify that the virtualised resources have been updated by the VIM according to the new deployment flavour	
	3	IOP Check	Verify that the impacted VNF instance(s) are running and reachable through the management network	
	4	IOP Check	Verify that the NFVO indicates the VNF DF update operation result as successful	
	5	IOP Check	Verify that NS has been updated by running the end-to-end functional test factoring the new VNF DF	
IOP Verdict				

7.7.3.8 VNF Configuration update on runtime

Test Description: VNF Configuration update on runtime	
Identifier	TD_NFV_NSLCM_UPDATE_VNF_CONFIG_001
Test Purpose	To verify that the configuration of a VNF running in a NS can be successfully updated
Configuration	SUT Configuration 1 SUT Configuration 3 SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO can generate "operate VNF" operation requests to the VNFM (NFVO_VNFLCM_OPERATE) • VNFM supports "operate VNF" operation requests from the NFVO (VNFM_VNFLCM_OPERATE) • NFVO/VNFM can generate "operate compute resource" operation requests to the VIM (NFVO_CRM_OPERATE or VNFM_CRM_OPERATE) • VIM supports "operate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_OPERATE_BY_NFVO or VIM_CRM_OPERATE_BY_VNFM) • NFVO/VNFM can query compute resource information from the VIM (NFVO_CRM_QUERY or VNFM_CRM_QUERY) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries by the NFVO (VNFM_VNFLCM_QUERY) • NFVO supports NS update operations for VNF configuration modification • VNFM supports VNF modification operations for configuration modification • VNF supports VNF configuration modification
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the NS update in NFVO with an operator request, specifying the new modified configuration to be applied to the given VNF(s)	
	2	IOP Check	Verify that the VNFM receives a VNF modification request for each new configuration to be applied to the given VNF(s)	
	3	IOP Check	Verify that the VNFM issue the modified configuration request to the relevant VNF(s) instance(s)	
	4	IOP Check	Verify that the modified configuration has been successfully applied in the given VNF(s)	

Test Description: VNF Configuration update on runtime		
IOP Verdict		

7.7.4 NS Healing

7.7.4.1 Partial NS Healing with an Operator Action

Test Description: partial NS healing with an operator action	
Identifier	TD_NFV_NSLCM_HEAL_001
Test Purpose	Verify that VNF instances inside the NS can be successfully healed when partial NS healing (VNF healing) is triggered by an operator action
Configuration	SUT Configuration 1
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • NFVO can generate VNF healing requests to the VNFM (NFVO_VNFLCM_HEAL) • VNFM supports VNF healing requests from the NFVO (VNFM_VNFLCM_HEAL)
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS is in a failed state (for example a virtualised resource needed by one or more VNF instances inside the NS has been terminated directly on the VIM)

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger partial NS healing (VNF healing) operation on the NFVO with an operator action	
	2	IOP Check	Verify that any additional resources required for the healing process have been allocated by the VIM according to the descriptors	
	3	IOP Check	Verify that other VNF instances(s) inside the NS are still running and reachable via their management network	
	4	IOP Check	Verify that healed VNF instance(s) are running and reachable via their management network	
	5	IOP Check	Verify that the healed VNF instances(s) have been configured according to the descriptors by querying the VNFM	
	6	IOP Check	Verify that any failed resources have been terminated and released by the VIM	
	7	IOP Check	Verify that the NFVO indicates the partial NS healing (VNF healing) operation result as successful	
	8	IOP Check	Verify that NS has been successfully healed by running an end-to-end functional test factoring the healed VNF instance(s)	

Test Description: partial NS healing with an operator action		
IOP Verdict		

7.7.4.2 Complete NS Healing with an Operator Action

Test Description: complete NS healing with an operator action				
Identifier	TD_NFV_NSLCM_HEAL_002			
Test Purpose	Verify that a NS can be successfully healed when complete NS healing is triggered by an operator action			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO/VNFM can generate "allocate compute resource" operation requests to the VIM (NFVO_CRM_ALLOCATE or VNFM_CRM_ALLOCATE) • NFVO/VNFM can generate "allocate network resource" operation requests to the VIM (NFVO_NRM_ALLOCATE or VNFM_NRM_ALLOCATE) • NFVO/VNFM can generate "allocate storage resource" operation requests to the VIM (NFVO_SRM_ALLOCATE or VNFM_SRM_ALLOCATE) • VIM supports "allocate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_ALLOCATE_BY_NFVO or VIM_CRM_ALLOCATE_BY_VNFM) • VIM supports "allocate network resource" operation requests from the NFVO/VNFM (VIM_NRM_ALLOCATE_BY_NFVO or VIM_NRM_ALLOCATE_BY_VNFM) • VIM supports "allocate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_ALLOCATE_BY_NFVO or VIM_SRM_ALLOCATE_BY_VNFM) • NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001) • NS is in a failed state (for example a virtualised resource needed by one or more VNF instances inside the NS has been terminated directly on the VIM)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger complete NS healing operation on the NFVO with an operator action	
	2	IOP Check	Verify that resources allocated to the faulty NS instance have been terminated by the VIM	
	3	IOP Check	Verify that resources required by the new NS instance have been allocated by the VIM	
	4	IOP Check	Verify that VNF instance(s) inside the new NS instance are running and reachable via their management network	
	5	IOP Check	Verify that the VNF instances(s) inside the new NS instance have been configured according to the descriptors by querying the VNFM	
	6	IOP Check	Verify that the NFVO indicates the complete NS healing operation result as successful	
	7	IOP Check	Verify that NS has been successfully healed by running an end-to-end functional test	
IOP Verdict				

7.7.5 NS Termination

7.7.5.1 Standalone NS Termination

Test Description: standalone NS termination				
Identifier	TD_NFV_NSLCM_TERMINATE_001			
Test Purpose	To verify that a standalone NS instance can be successfully terminated			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries from the NFVO (VNFM_VNFLCM_QUERY)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the termination of the NS instance on the NFVO	
	2	IOP Check	Verify that all the VNF instance(s) have been terminated by querying the VNFM	
	3	IOP Check	Verify that the resources allocated to the NS and VNF instance(s) have been released by the VIM	
	4	IOP Check	Verify that the NFVO indicates NS instance termination operation result as successful	
IOP Verdict				

7.7.5.2 Nested NS Termination

Test Description: nested NS termination	
Identifier	TD_NFV_NSLCM_TERMINATE_NESTED_NS_001
Test Purpose	To verify that a NS instance referencing an existing nested NS can be successfully terminated
Configuration	SUT Configuration 3
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM)

Test Description: nested NS termination	
	• VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM)VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries from the NFVO (VNFM_VNFLCM_QUERY)
Pre-test conditions	• NSD2 references nested NSD1 • NS2 is instantiated (TD_NFV_NSLCM_INSTANTIATE_NEST_NS_001) • NS1 instance can function without the impacted NS2 instance

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger NS2 instance termination on the NFVO	
	2	IOP Check	Verify that the resources that were allocated to the VNF instance(s) inside NS2 have been released by the VIM	
	3	IOP Check	Verify that VNF instance(s) inside NS1 are still running and reachable through the management network	
	4	IOP Check	Verify that all VNF instance(s) in NS2 have been terminated by querying the VNFM	
	5	IOP Check	Verify that the NFVO indicates NS2 instance termination operation result as successful	
	6	IOP Check	Verify that NS1 instance was unaffected by NS2 instance termination by running an end-to-end functional test factoring in the functionality of VNF instance(s) in NS1	
IOP Verdict				

7.7.5.3 Multi-Site NS Termination

Test Description: Multi-Site NS terminate	
Identifier	TD_NFV_NSLCM_TERMINATE_MULTISITE_001
Test Purpose	To verify that a Multi-Site NS instance can be successfully terminated
Configuration	SUT Configuration 4
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIMs (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) deployment in different site • NFVO/VNFM can generate "terminate network resource" operation requests to the VIMs (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) deployment in different site • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIMs (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) deployment in different site • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries from the NFVO (VNFM_VNFLCM_QUERY)
Pre-test conditions	• Multi-Site NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_MULTISITE_001)

Test Description: Multi-Site NS terminate				
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the termination of the Multi-Site NS instance on the NFVO	
	2	IOP Check	Verify that all the VNF instance(s) have been terminated in the given sites	
	3	IOP Check	Verify that the resources allocated to the Multi-Site NS and VNF instance(s) have been released by the involved VIMs	
	4	IOP Check	If applicable, verify that the NFPs have been deleted	
	5	IOP Check	Verify that the Multi-Site NS instance does no longer exist	
IOP Verdict				

7.7.5.4 SFC NS Termination

Test Description: SFC NS termination				
Identifier	TD_NFV_NSLCM_TERMINATE_SFC_001			
Test Purpose	To verify that a NS instance can be successfully terminated			
Configuration	SUT Configuration 1			
References	ETSI GS NFV-IFA 005 [i.3], ETSI GS NFV-IFA 006 [i.4], ETSI GS NFV-IFA 007 [i.5], ETSI GS NFV-IFA 008 [i.6], ETSI GS NFV-IFA 010 [i.7] and ETSI GS NFV-IFA 013 [i.9]			
Applicability	• NFVO/VNFM can generate "terminate compute resource" operation requests to the VIM (NFVO_CRM_TERMINATE or VNFM_CRM_TERMINATE) • NFVO/VNFM can generate "terminate network resource" operation requests to the VIM (NFVO_NRM_TERMINATE or VNFM_NRM_TERMINATE) • NFVO/VNFM can generate "terminate storage resource" operation requests to the VIM (NFVO_SRM_TERMINATE or VNFM_SRM_TERMINATE) • VIM supports "terminate compute resource" operation requests from the NFVO/VNFM (VIM_CRM_TERMINATE_BY_NFVO or VIM_CRM_TERMINATE_BY_VNFM) • VIM supports "terminate network resource" operation requests from the NFVO/VNFM (VIM_NRM_TERMINATE_BY_NFVO or VIM_NRM_TERMINATE_BY_VNFM) • VIM supports "terminate storage resource" operation requests from the NFVO/VNFM (VIM_SRM_TERMINATE_BY_NFVO or VIM_SRM_TERMINATE_BY_VNFM) • VIM can send compute resource information to the NFVO (VIM_CRM_INFO_TO_NFVO or VIM_CRM_INFO_TO_VNFM) • VIM can send network resource information to the NFVO (VIM_NRM_INFO_TO_NFVO or VIM_NRM_INFO_TO_VNFM) • VIM can send storage resource information to the NFVO (VIM_SRM_INFO_TO_NFVO or VIM_SRM_INFO_TO_VNFM) • NFVO can query VNF information from the VNFM (NFVO_VNFLCM_QUERY) • VNFM supports VNF information queries from the NFVO (VNFM_VNFLCM_QUERY)			
Pre-test conditions	• NS is instantiated (TD_NFV_NSLCM_INSTANTIATE_SFC_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	Stimulus	Trigger the termination of the NS instance on the NFVO	
	2	IOP Check	Verify that all the VNF instance(s) have been terminated by querying the VNFM.	
	3	IOP Check	Verify that the resources allocated to the NS and VNF instance(s) have been released by the VIM	
	4	IOP Check	If applicable, verify that the NFPs have been deleted	
	5	IOP Check	Verify that the NFVO indicates NS instance termination operation result as successful	
IOP Verdict				

Annex A:
Technical Report Card Sample

Test ID	xxxx
Tested By	xxxx
Date and Time	dd-mm-yyyy hh:mm

Functional Block	Product Name	Version	Product Description	Configuration
NFVO				
VNFM				
VIM				
NFVI				
EM				
VNF				

Functional Area	Test Description ID	SUT Configuration	IOP Verdict
Software Image Management	TD_NFV_SWIM_ADD_001	SUT Configuration 1	OK
Software Image Management	TD_NFV_SWIM_DELETE_001	SUT Configuration 1	OK
VNF Package Management	TD_NFV_VNFPM_ONBOARD _001	SUT Configuration 1	OK
VNF Package Management	TD_NFV_VNFPM_DELETE_001	SUT Configuration 1	OK
VNF Lifecycle Management	TD_NFV_VNFLCM_INSTANTIATE_VNF_001	SUT Configuration 1	OK
VNF Lifecycle Management	TD_NFV_VNFLCM_QUERY_VNF_001	SUT Configuration 1	OK
VNF Lifecycle Management	TD_NFV_VNFLCM_MODIFY_VNF_INFO_001	SUT Configuration 1	NOK
VNF Lifecycle Management	TD_NFV_VNFLCM_START_VNF_001	SUT Configuration 1	OK
VNF Lifecycle Management	TD_NFV_VNFLCM_STOP_VNF_001	SUT Configuration 1	NOK
VNF Lifecycle Management	TD_NFV_VNFLCM_HEAL_VNF_001	SUT Configuration 1	N/A
VNF Lifecycle Management	TD_NFV_VNFLCM_SCALE_OUT_VNF_001	SUT Configuration 1	NOK
VNF Lifecycle Management	TD_NFV_VNFLCM_SCALE_IN_VNF_001	SUT Configuration 1	NOK
VNF Lifecycle Management	TD_NFV_VNFLCM_TERMINATE_VNF_001	SUT Configuration 1	OK
NS Lifecycle Management	TD_NFV_NSLCM_INSTANTIATE_001	SUT Configuration 1	OK
NS Lifecycle Management	TD_NFV_NSLCM_SCALE_OUT_001	SUT Configuration 1	OK
NS Lifecycle Management	TD_NFV_NSLCM_SCALE_IN_001	SUT Configuration 1	NOK
NS Lifecycle Management	TD_NFV_NSLCM_TERMINATE_001	SUT Configuration 1	OK

Annex B: Document Usage Process Diagram

This annex explains how the present document can be best utilized in a process diagram.

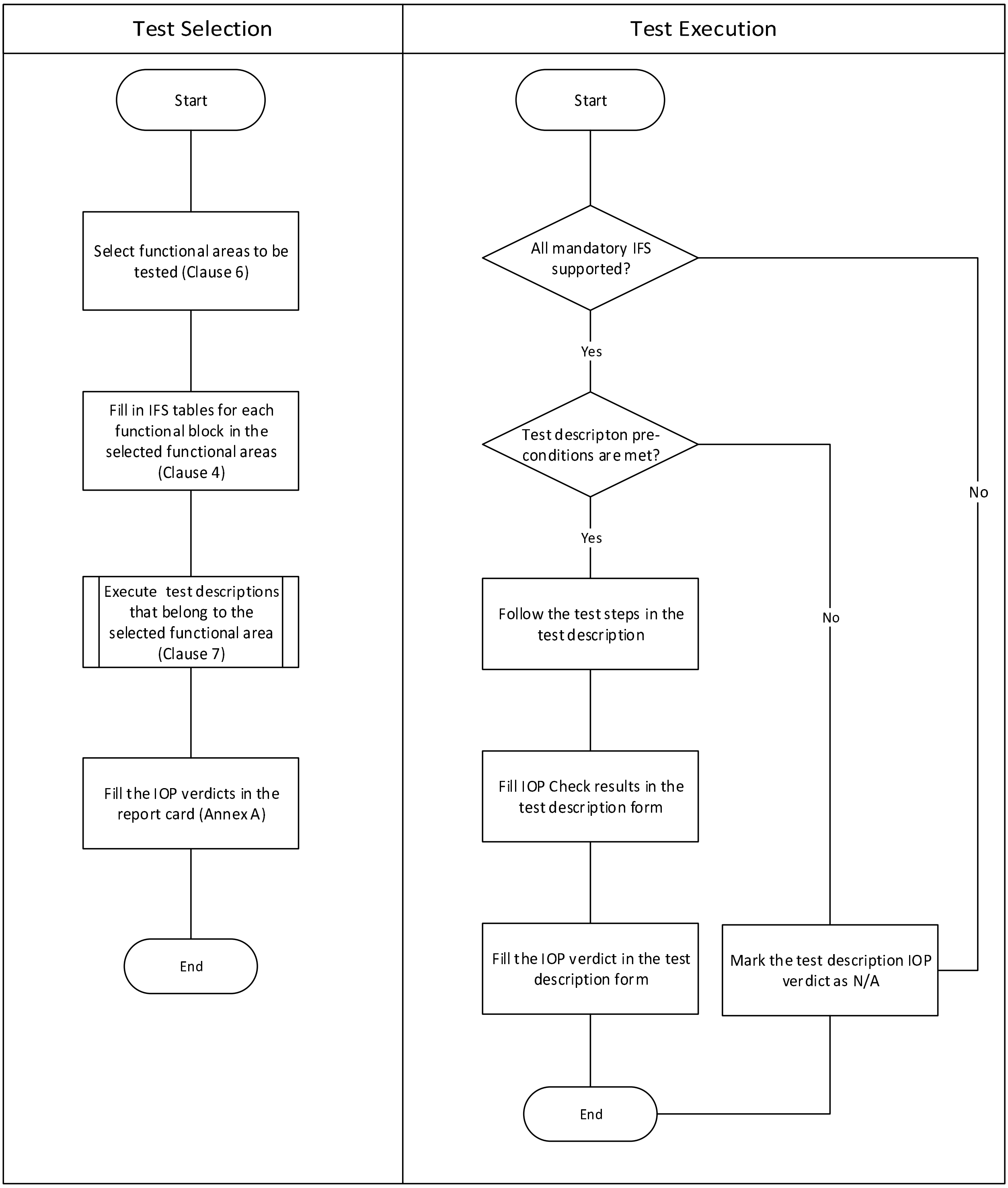
Test Selection
Test Execution
Start
Select functional areas to be tested (Clause 6)
Fill in IFS tables for each functional block in the selected functional areas (Clause 4)
Execute test descriptions that belong to the selected functional area (Clause 7)
Fill the IOP verdicts in the report card (Annex A)
End
Start
All mandatory IFS supported?
Yes
Test descripton pre-conditions are met?
Yes
Follow the test steps in the test description
Fill IOP Check results in the test description form
Fill the IOP verdict in the test description form
No
No
Mark the test description IOP verdict as N/A
End

Figure B-1: Document usage process diagram

Annex C: Interoperability Features Statement (IFS) Diagram

C.1 Introduction

This annex illustrates the IFS defined or VIM, VNFM, NFVO, and EM/VNF based on clause 4 of the present document in the figures C.2-1, C.3-1, C.4-1 and C.5-1.

C.2 IFS for VIM

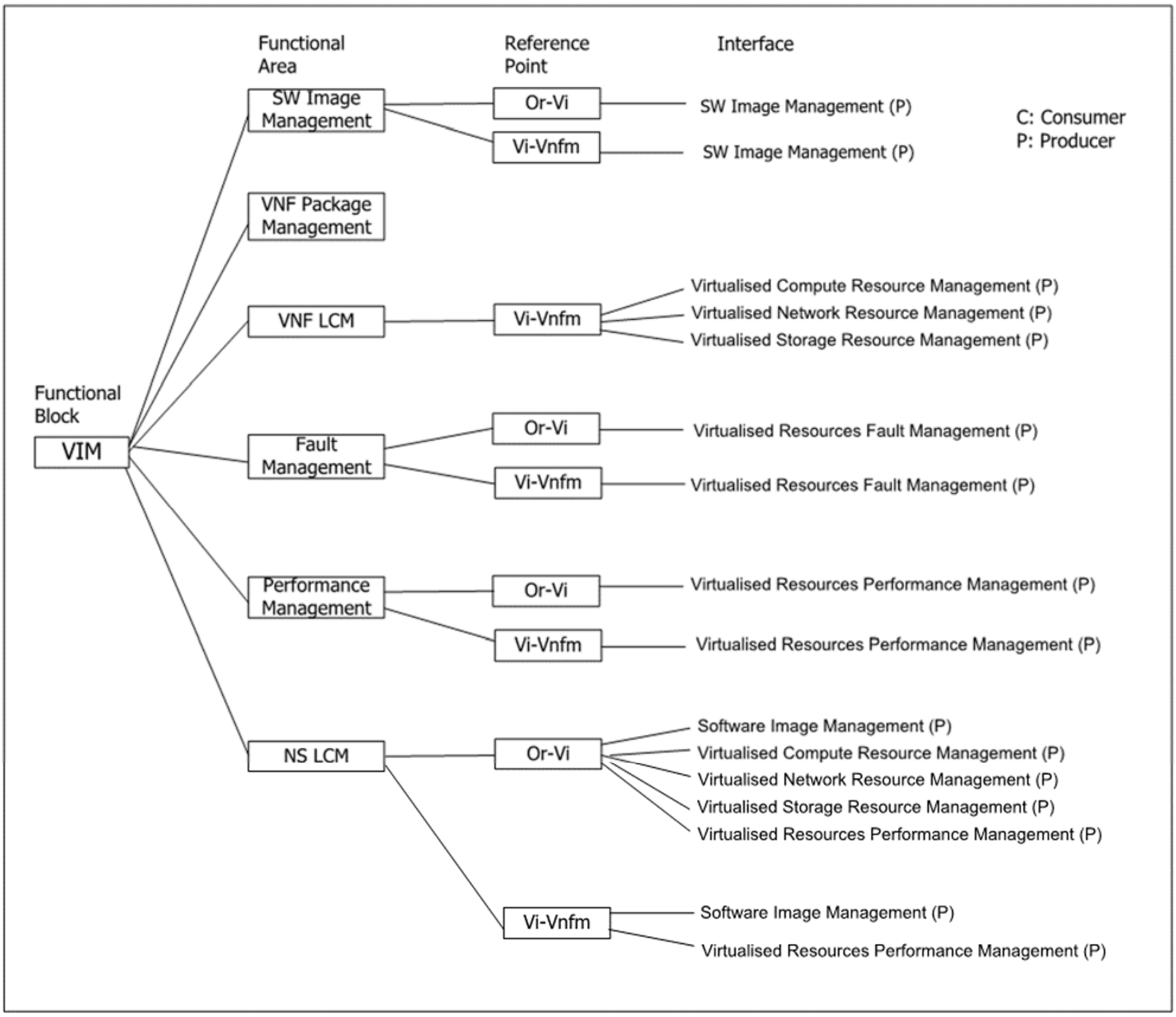
Functional Area
Reference Point
Interface
C: Consumer
P: Producer
Functional Block
VIM
SW Image Management
Or-Vi
SW Image Management (P)
Vi-Vnfm
SW Image Management (P)
VNF Package Management
VNF LCM
Vi-Vnfm
Virtualised Compute Resource Management (P)
Virtualised Network Resource Management (P)
Virtualised Storage Resource Management (P)
Fault Management
Or-Vi
Virtualised Resources Fault Management (P)
Vi-Vnfm
Virtualised Resources Fault Management (P)
Performance Management
Or-Vi
Virtualised Resources Performance Management (P)
Vi-Vnfm
Virtualised Resources Performance Management (P)
NS LCM
Or-Vi
Software Image Management (P)
Virtualised Compute Resource Management (P)
Virtualised Network Resource Management (P)
Virtualised Storage Resource Management (P)
Virtualised Resources Performance Management (P)
Vi-Vnfm
Software Image Management (P)
Virtualised Resources Performance Management (P)

Figure C.2-1: IFS for VIM

C.3 IFS for VNFM

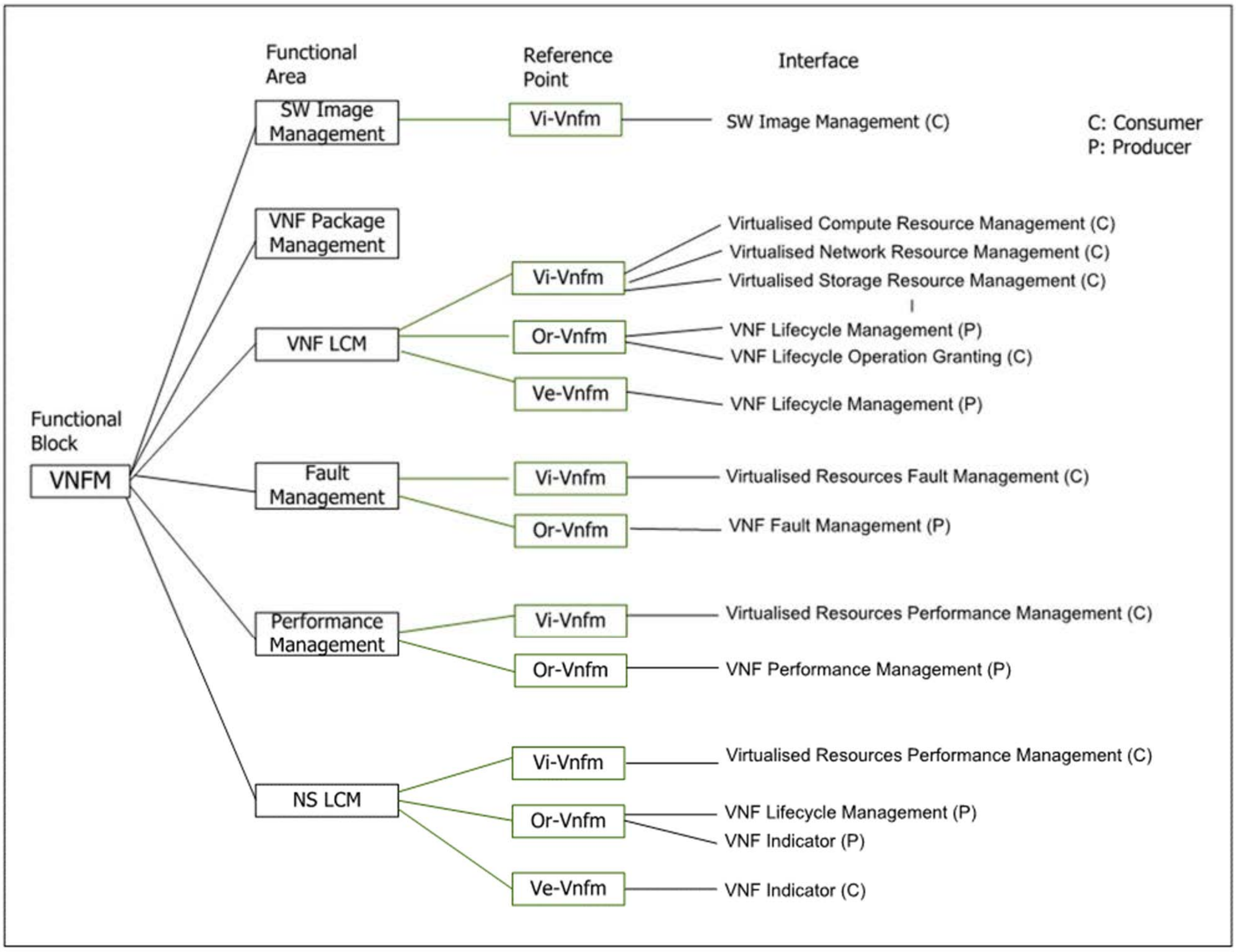
Functional Block
VNFM
Functional Area
Reference Point
Interface
C: Consumer
P: Producer
SW Image Management
Vi-Vnfm
SW Image Management (C)
VNF Package Management
Vi-Vnfm
Virtualised Compute Resource Management (C)
Virtualised Network Resource Management (C)
Virtualised Storage Resource Management (C)
VNF LCM
Or-Vnfm
VNF Lifecycle Management (P)
VNF Lifecycle Operation Granting (C)
Ve-Vnfm
VNF Lifecycle Management (P)
Fault Management
Vi-Vnfm
Virtualised Resources Fault Management (C)
Or-Vnfm
VNF Fault Management (P)
Performance Management
Vi-Vnfm
Virtualised Resources Performance Management (C)
Or-Vnfm
VNF Performance Management (P)
NS LCM
Vi-Vnfm
Virtualised Resources Performance Management (C)
Or-Vnfm
VNF Lifecycle Management (P)
VNF Indicator (P)
Ve-Vnfm
VNF Indicator (C)

Figure C.3-1: IFS for VNFM

C.4 IFS for NFVO

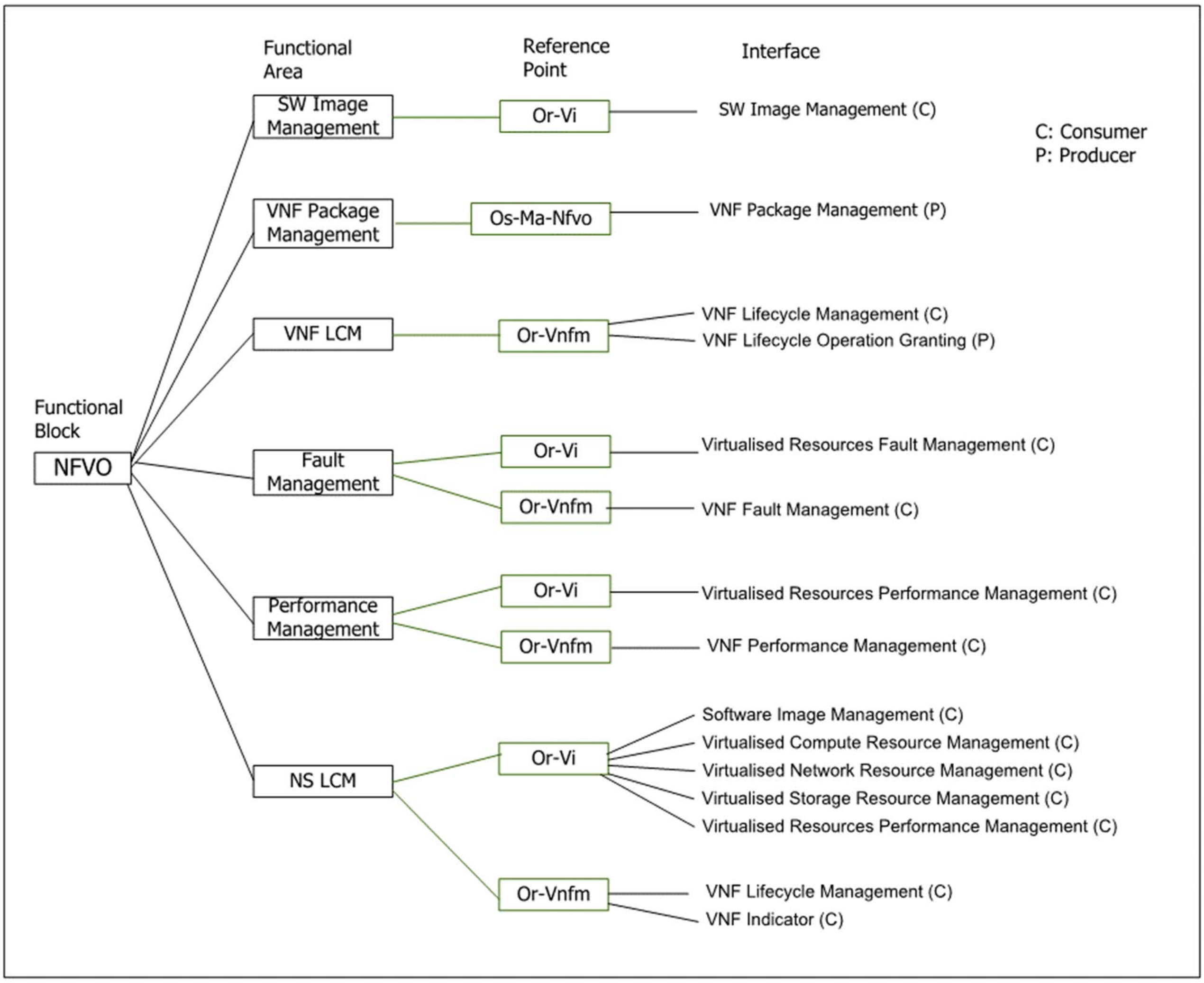
Functional Area
Reference Point
Interface
C: Consumer
P: Producer
Functional Block
NFVO
SW Image Management
Or-Vi
SW Image Management (C)
VNF Package Management
Os-Ma-Nfvo
VNF Package Management (P)
VNF LCM
Or-Vnfm
VNF Lifecycle Management (C)
VNF Lifecycle Operation Granting (P)
Fault Management
Or-Vi
Virtualised Resources Fault Management (C)
Or-Vnfm
VNF Fault Management (C)
Performance Management
Or-Vi
Virtualised Resources Performance Management (C)
Or-Vnfm
VNF Performance Management (C)
NS LCM
Or-Vi
Software Image Management (C)
Virtualised Compute Resource Management (C)
Virtualised Network Resource Management (C)
Virtualised Storage Resource Management (C)
Virtualised Resources Performance Management (C)
Or-Vnfm
VNF Lifecycle Management (C)
VNF Indicator (C)

Figure C.4-1: IFS for NFVO

C.5 IFS for EM/VNF

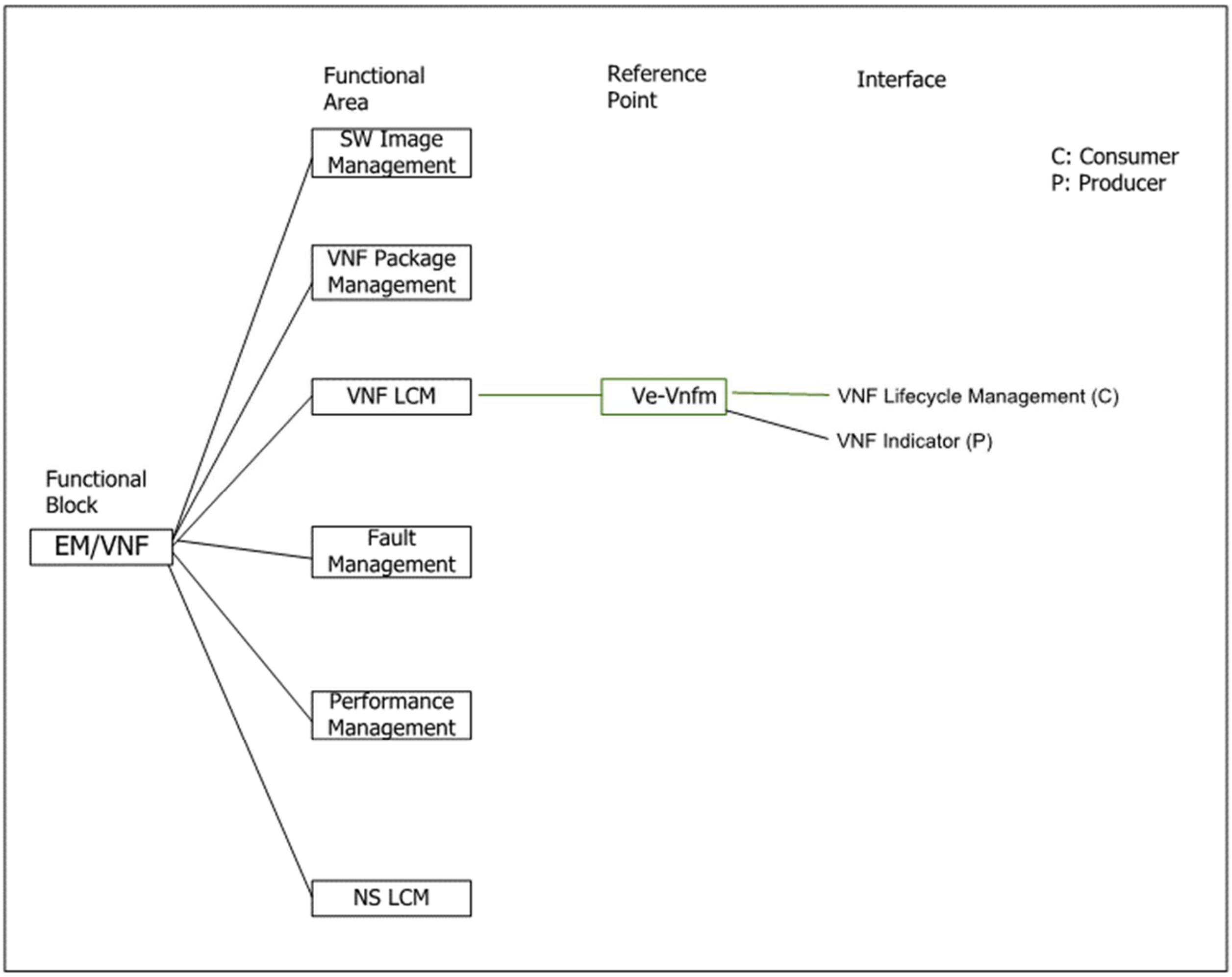

Figure C.5-1: IFS for EM/VNF

History

Document history		
V2.5.1	August 2018	Publication
V2.6.1	January 2020	Publication