ETSI Plugtests Test Plan

V1.0.0 (2017-02)

**1st ETSI NFV Plugtests
Madrid, Spain
23rd January – 3rd February**

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

Important notice

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

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

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

Contents

Foreword

This Test Plan has been produced by ETSI Centre for Testing and Interoperability during the preparation of the 1st ETSI NFV Plugtests.

Introduction

The present document describes the Interoperability Test Plan that was followed during the 1st ETSI NFV Plugtests held from 23rd January to 3rd February 2017 in Madrid, Spain.

The Test Plan was developed following the interoperability testing methodology defined by ETSI NFV in [TST002]

During the 1st NFV Plugtests, this Test Plan was run on 160 Test Sessions, that is, it allowed to test for interoperability over 160 Systems Under Test made of different combinations of the 35 participating implementations.

1 Scope

The goal of this document it to support the interoperability test sessions run during the 1st NFV Plugtests. This event aimed at verifying early interoperability between different implementations of the main components of the NFV Architectural Framework, which included:

- Virtual Network Functions (VNF)

- Management and Orchestration (MANO) solutions, providing pre-integrated NFV Orchestrator (NFVO) and VNF Manager (VNFM) functionality

- NFV Platforms providing pre-integrated NFV infrastructure (NFVI) and Virtual Infrastructure Manager (VIM) functionality

During the Test Sessions, the Systems Under Test were composed of the following Functions Under Test; one NFV Platform (NFVI&VIM) one MANO solution, and one or several VNFs, each of them from different participants. The Test Sessions were supported by Test VNFs which were included in the Network Service together with the VNF(s) Under Test in order to be able to validate appropriate functional behaviour of the VNFs and NS.

This document includes several System Under Test configurations to illustrate how the Test Descriptions can be applied to different types of VNFs (end point, middle point, ..) and how scaling or updating operations would impact each of them.

The NS and VNF compositions shown in this Test Plan are not mandated but provided as examples. The Test Descriptions have been written in such a way that also apply to more complex VNFs and NS derived from the expamples provided in the SUT Configurations.

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.

[NFV002] ETSI GS NFV 002: "Network Functions Virtualisation (NFV); Architectural Framework".

[NFV003] ETSI GS NFV 003: "Network Functions Virtualisation (NFV); Terminology for main concepts in NFV".

[IFA005] ETSI GS NFV-IFA 005: "Network Functions Virtualisation (NFV); Management and Orchestration; Or-Vi reference point - Interface and Information Model Specification".

[IFA006] ETSI GS NFV-IFA 006: "Network Functions Virtualisation (NFV); Management and Orchestration; Vi-Vnfm reference point - Interface and Information Model Specification".

[IFA007] ETSI GS NFV-IFA 007: "Network Functions Virtualisation (NFV); Management and Orchestration; Or-Vnfm reference point - Interface and Information Model Specification".

[IFA008]   ETSI GS NFV-IFA 008: "Network Functions Virtualisation (NFV); Management and Orchestration; Ve-Vnfm reference point - Interface and Information Model Specification".

[IFA010]   ETSI GS NFV-IFA 010: "Network Functions Virtualisation (NFV); Management and Orchestration; Functional requirements specification".

[IFA013]   ETSI GS NFV-IFA 013: "Network Functions Virtualisation (NFV); Management and Orchestration; Os-Ma-Nfvo reference point - Interface and Information Model Specification".

[TST002]   ETSI GS NFV-TST 002: "Network Functions Virtualisation (NFV); Testing Methodology; Report on NFV Interoperability Testing Methodology"

3   Definitions, symbols and abbreviations

3.1   Definitions

For the purposes of the present document, the terms and definitions given in [NFV003] and [TST002] apply.

3.2   Symbols

None

3.3   Abbreviations

For the purposes of the present document, the abbreviations given in [NFV003] and [TST002] apply.

4   Test Suite Structure

4.1   Naming Convention

The Test Ids of this Test Plan have been created as per the following naming convention:

TEST ID = TD_<ROOT>_<GROUP>_<OPERATION>_<COMPLEMENT>_<NN>

Where <ROOT> is NFV, and <NN> corresponds to a sequential number starting at 01.

The possible values for <GROUP>, <OPERATION> and <COMPLEMENT> are summarized in the table below:

<GROUP>	<OPERATION>	<COMPLEMENT>	Description
SETUP	ONBOARD	VNF_PKG	Onboard VNF Package
		NSD	Onboard Network Service Descriptor
NS_LCM	INSTANTIATE	-	Instantiate Network Service
	SCALE	OUT	Scale Out by adding VNF instance(s)
		IN	Scale In by adding VNF instance(s)
	SCALE (VNF)	OUT_VNF	Scale Out by adding VNFC instance(s)
		IN_VNF	Scale In by adding VNFC instance(s)
	UPDATE	STOP_VNF	Stop VNF
		START_VNF	Re-Start VNF
		ADD_VNF_VL	Add VNF and VL
		REM_VNF_VL	Remove VNF and VL
	TERMINATE	-	Terminate Network Service
TEARDOWN	DELETE	NSD	Delete Network Service Descriptor
		VNF_PKG	Delete VNF Package

4.2 Test Summary

4.2.1 SETUP

Test Id	Test Purpose
TD_NFV_SETUP_ONBOARD_VNF_PKG_001	To on-board a VNF Package
TD_NFV_SETUP_ONBOARD_NSD_001	To onboard a NSD

4.2.2 NS LCM

4.2.2.1 INSTANTIATE

Test Id	Test Purpose
TD_NFV_NS_LCM_INSTANTIATE_001	To verify that an NS can be successfully instantiated

4.2.2.2 SCALE

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_OUT_001	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered by a MANO operator
TD_NFV_NS_LCM_SCALE_OUT_001	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered by a MANO operator
TD_NFV_NS_LCM_SCALE_IN_001	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator
TD_NFV_NS_LCM_SCALE_OUT_002	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by a VNF Indicator
TD_NFV_NS_LCM_SCALE_IN_002	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by a VNF Indicator
TD_NFV_NS_LCM_SCALE_OUT_003	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by a VIM KPI
TD_NFV_NS_LCM_SCALE_IN_003	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by a VIM KPI
TD_NFV_NS_LCM_SCALE_OUT_004	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered in MANO by a VNF/EM request
TD_NFV_NS_LCM_SCALE_IN_004	To verify that a NS can successfully scale in (by removing VNF instances) if triggered in MANO by a VNF/EM request

4.2.2.3 SCALE VNF

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_OUT_VNF_001	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered by a MANO operator
TD_NFV_NS_LCM_SCALE_IN_VNF_001	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered by a MANO operator

TD_NFV_NS_LCM_SCALE_OUT_VNF_002	To verify that a VNF in a NS can be successfully scaled out (by adding VNFV instances (VMs)) when triggered automatically in MANO by a VNF Indicator
TD_NFV_NS_LCM_SCALE_IN_VNF_002	To verify that a VNF in a NS can be successfully scaled in (by adding VNFC instances (VMs)) when triggered automatically in MANO by a VNF Indicator
TD_NFV_NS_LCM_SCALE_OUT_VNF_003	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered automatically in MANO by a VIM KPI
TD_NFV_NS_LCM_SCALE_IN_VNF_003	To verify that a VNF in a NS can be successfully scaled in (by adding VNFC instances (VMs)) when triggered automatically in MANO by a VIM KPI
TD_NFV_NS_LCM_SCALE_OUT_VNF_004	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered in MANO by a VNF/EM request
TD_NFV_NS_LCM_SCALE_IN_VNF_004	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered in MANO by a VNF/EM request

4.2.2.4   UPDATE

Test Id	Test Purpose
TD_NFV_NS_LCM_UPDATE_STOP_VNF_001	To verify that a VNF running in a NS can be successfully stopped by MANO
TD_NFV_NS_LCM_UPDATE_START_VNF_001	To verify that a stopped VNF in a NS can be successfully re-started by MANO
TD_NFV_NS_LCM_UPDATE_ADD_VNF_VL_001	To verify that VNF(s) and VL(s) can be added to a running NS
TD_NFV_NS_LCM_UPDATE_REM_VNF_VL_001	To verify that VNF(s) and VL(s) can be removed from a running NS

4.2.2.5   TERMINATE

Test Id	Test Purpose
TD_NFV_NS_LCM_TERMINATE_001	To verify that a NS can be successfully terminated

4.2.3   TEARDOWN

Test Id	Test Purpose
TD_NFV_TEARDOWN_DELETE_NSD_001	To delete a NSD
TD_NFV_TEARDOWN_DELETE_VNF_PKG_001	To delete a VNF Package

5 System Under Test Configurations

5.1 SUT_1_NS_1_ENDPOINT
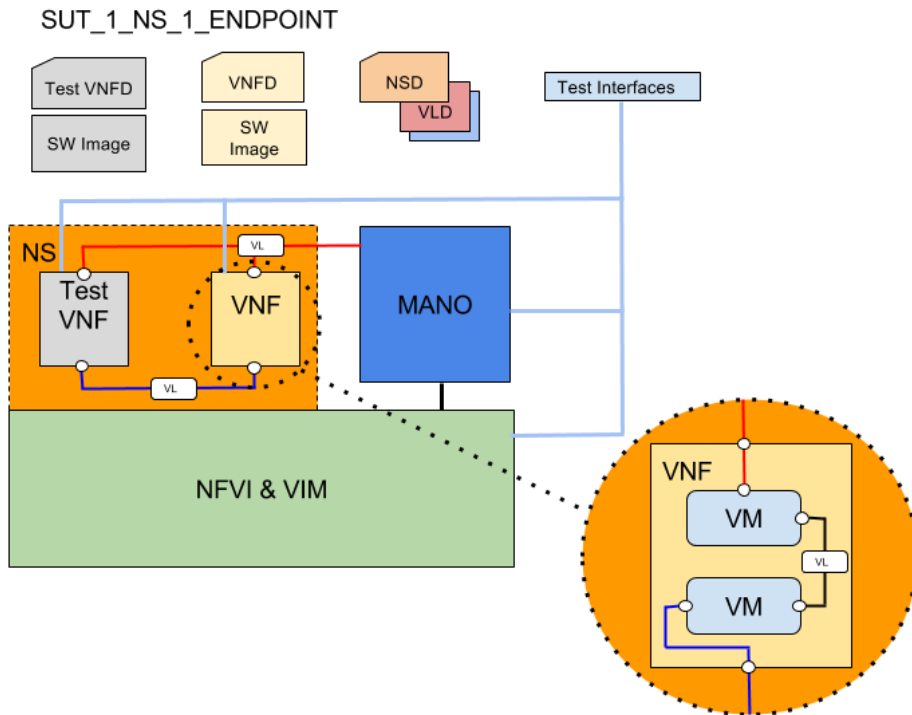
Figure 1. SUT_1_NS_1_ENDPOINT

5.2   SUT_1_NS_1_MIDDLEPOINT
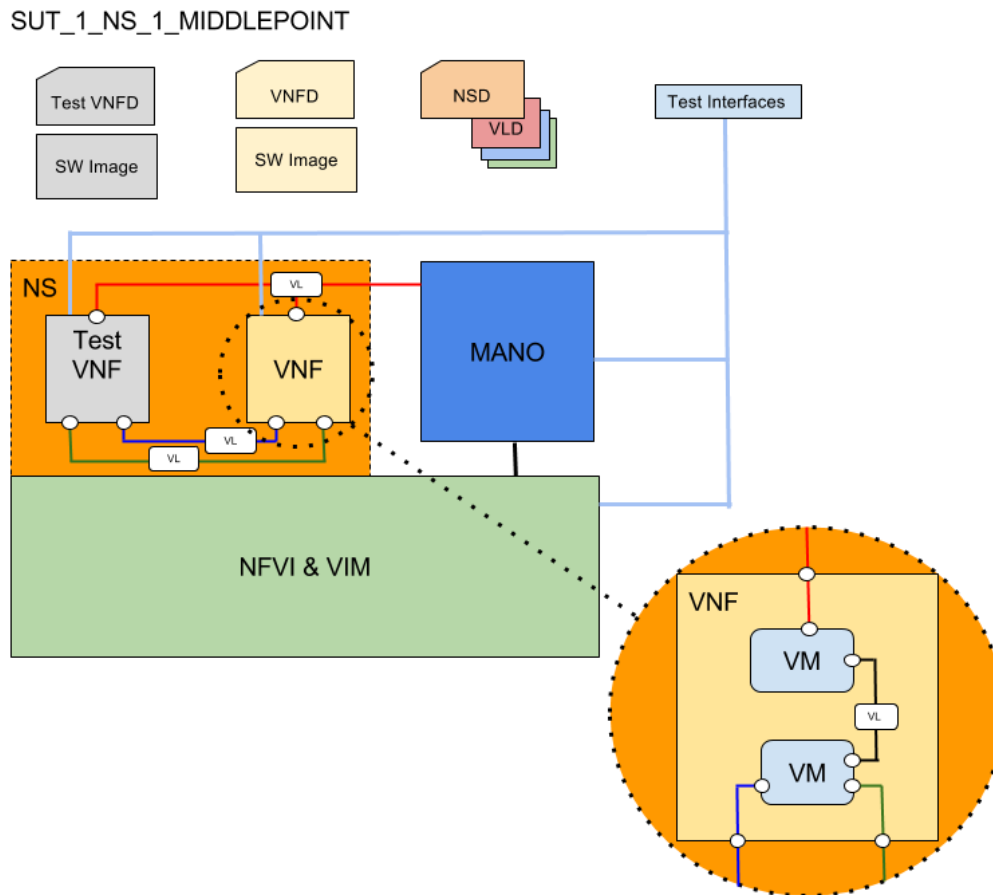
Figure 2. SUT_1_NS_1_MIDDLEPOINT

5.3   SUT_1_NS_1_ENDPOINT_SCALE

SUT_1_NS_1_ENDPOINT_SCALE

Figure 3. SUT_1_NS_1_ENDPOINT_SCALE

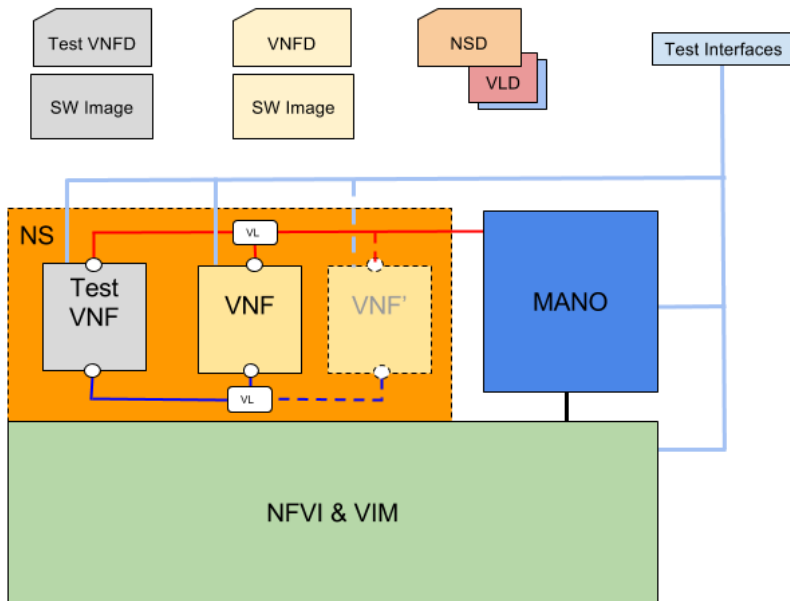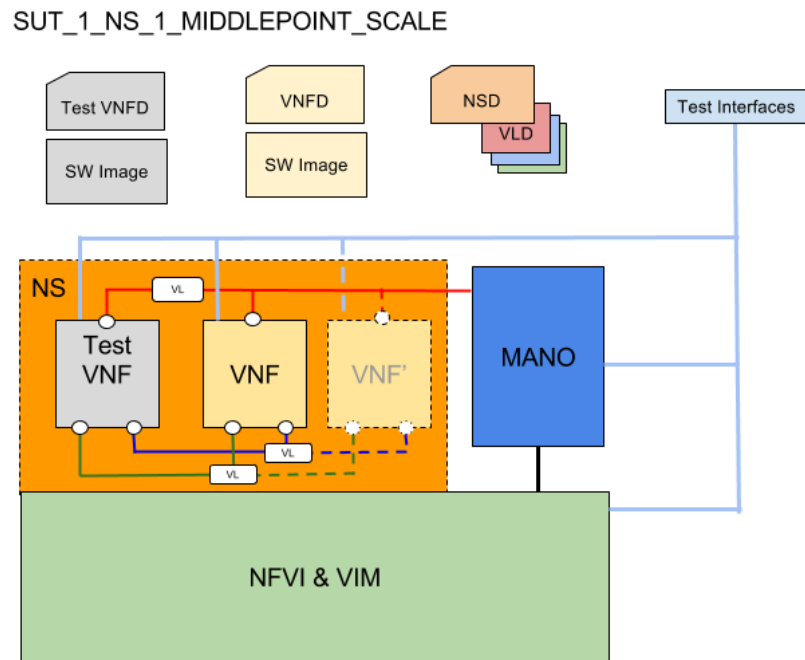## 5.4   SUT_1_NS_1_MIDDLEPOINT_SCALE

Figure 4. SUT_1_NS_1_MIDDLEPOINT_SCALE

5.5 SUT_1_NS_1_ENDPOINT_SCALE_VNF
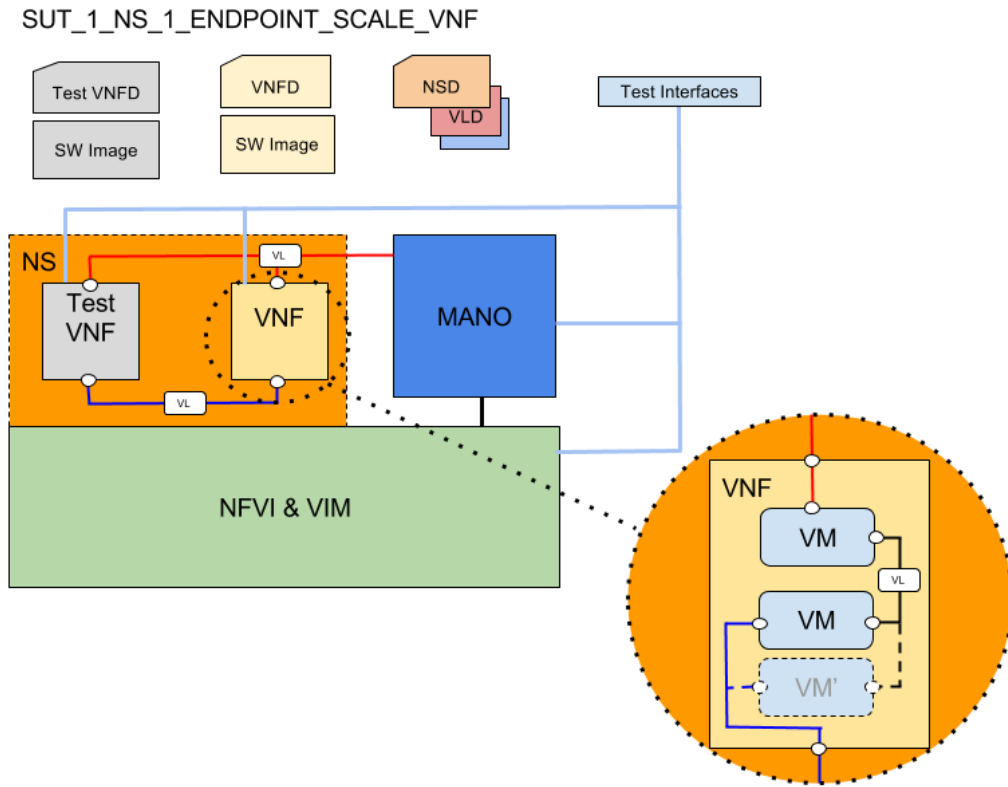
Figure 5. SUT_1_NS_1_ENDPOINT_SCALE_VNF

5.6 SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF

SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF
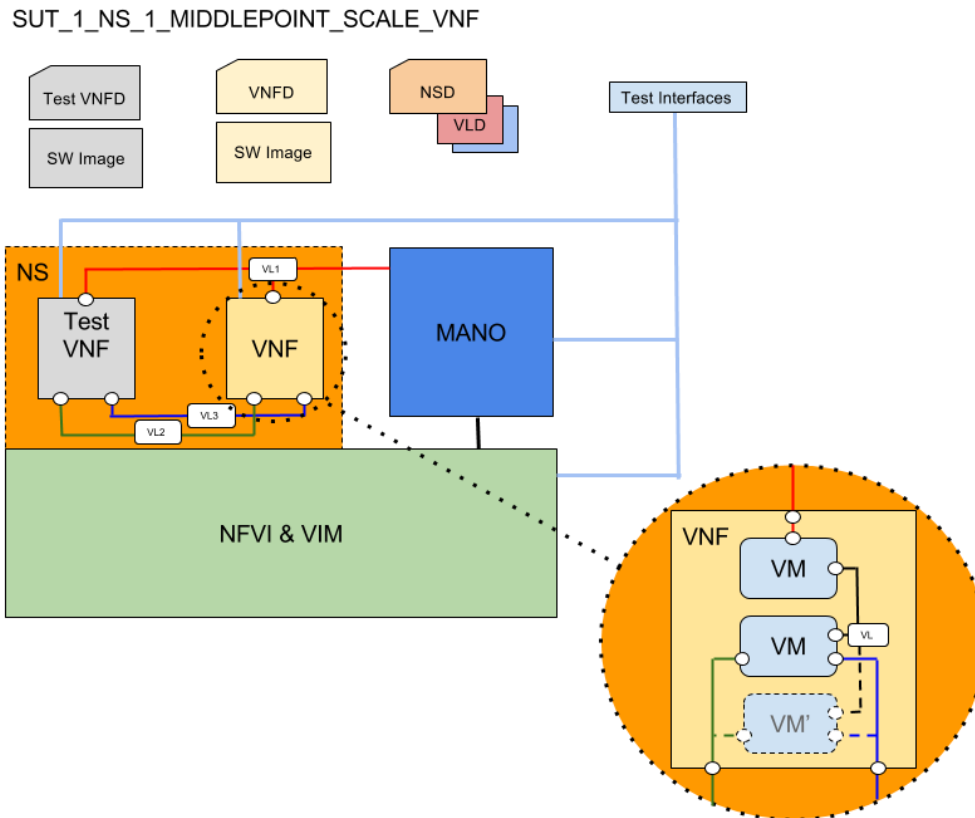
Figure 6. SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF

5.7 SUT_1_NS_1_MIDDLEPOINT_UPDATE
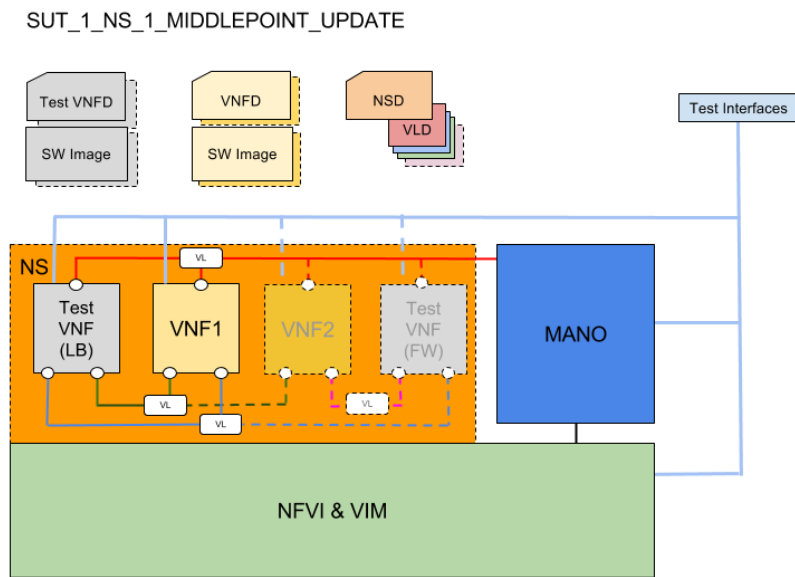
Figure 7. SUT_1_NS_1_MIDDLEPOINT_UPDATE

6 Interoperability Test Descriptions

6.1 SETUP

6.1.1 ONBOARD

6.1.1.1 TD_NFV_SETUP_ONBOARD_VNF_PKG_001

Interoperability Test Description				
Identifier	TD_NFV_SETUP_ONBOARD_VNF_PKG_001			
Test Purpose	To on-board a VNF Package			
Configuration	SUT_1_NS_1_ENDPOINT SUT_1_NS_1_MIDDLEPOINT			
References	ETSI GS NFV-IFA013 V2.1.1 (clauses 7.7) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.5.1, 7.4.1)			
Applicability				
Pre-test conditions	• VNF Package resides on a repository reachable by MANO • VNF Package is complete and consumable by MANO			

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	stimulus	Trigger the on-boarding of VNF package on MANO	
	2	IOP check	Verify the VNF Package information is correct and complete on MANO (i.e. display or query and check release date, vendor info, manifest, VNFD, SW image meta-data, files contained in the VNF Package, ...)	
IOP Verdict				

6.1.1.2 TD_NFV_SETUP_ONBOARD_NSD_001

Interoperability Test Description				
Identifier	TD_NFV_SETUP_ONBOARD_NSD_001			
Test Purpose	To onboard a NSD			
Configuration	SUT_1_NS_1_ENDPOINT SUT_1_NS_1_MIDDLEPOINT			
References	ETSI GS NFV-IFA013 V2.1.1 (clauses 7.2.2, 7.3.2) ETSI GS NFV-IFA014 V2.1.1 (clause 6.2)			
Applicability				
Pre-test conditions	• NSD and referenced VLD and VNFFGDs exist and can be reached and conumed by MANO • Required VNF Packages have been on-boarded (TD_NFV_SETUP_VNF_PKG_ONB_001)			

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	stimulus	Trigger the on-boarding of the NSD on MANO	
	2	IOP check	Verify that NSD is successfully on-boarded in MANO (i.e query, display, ...)	
	3	IOP check	Verify that all VLDs and VNFFGDs referenced in the NSD have been successfully on-boarded in MANO	
IOP Verdict				

6.2 NS LCM

6.2.1 INSTANTIATE

6.2.1.1 TD_NFV_NS_LCM_INSTANTIATE_001

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_INSTANTIATE_001			
Test Purpose	To verify that an NS can be successfully instantiated			
Configuration	SUT_1_NS_1_ENDPOINT SUT_1_NS_1_MIDDLEPOINT			
References	ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.3) ETSI GS NFV-IFA005 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA006 V2.1.1 (clause 7.2.3) ETSI GS NFV-IFA008 V2.1.1 (clause 6.2.2) ETSI GS NFV-IFA010 V2.1.1 (clause 6.3.2)			
Applicability	• MANO can request VIM_NFVI to add a SW image • VIM_NFVI supports adding a SW image • MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • (If required by NSD) MANO can request VIM_NFVI to create NFP(s) • (If required by NSD) VIM_NFVI supports creating NFP(s)			
Pre-test conditions	• NSD, VLD(s), VNFFGD(s) and VNF Package(s) have been on-boarded in MANO • 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 in MANO	
	2	IOP check	Verify that the software images have been onboarded in the VIM	
	3	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	4	IOP check	Verify that the VNF (s) have been deployed according to the descriptors (VMs, VLs, CPs...)	
	5	IOP check	Verify that the VL and VNFFG instance(s) have been created according to the descriptors	
	6	IOP check	Verify that the VNF(s) are running and reachable through the management network	
	7	IOP check	Verify that the VNF(s) have been configured according to VNFD(s) (i.e by obtaining a result from the management interface)	
	8	IOP check	Verify that the VNF(s), VL(s) and VNFFG(s) have been connected according to the Descriptors	
	9	IOP check	Verify that the NS is successfully instantiated by running the end-to-end functional test	
IOP Verdict				

6.2.3   SCALE

6.2.3.1   TD_NFV_NS_LCM_SCALE_OUT_001

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_SCALE_OUT_001			
Test Purpose	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered by a MANO operator			
Configuration	SUT_1_NS_1_ENDPOINT_SCALE SUT_1_NS_1_MIDDLEPOINT_SCALE			
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)			
Applicability	• MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • MANO supports triggering scale out with an operator's action • MANO supports scale out by adding VNF instances • NS/VNF supports scale out by adding VNF instances			
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)			

Test Sequence	Step	Type	Description	Result
	1	stimulus	Trigger NS scale out (by adding VNF instances) in MANO with an operator action	
	2	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	3	IOP check	Verify that the additional VNF instance(s) have been deployed	
	4	IOP check	Verify that the additional VNF instances(s) are running and reachable from the management network	
	5	IOP check	Verify that the additional VNF instances(s) have been configured according to the descriptors (i.e. by geting a result through the management interface)	
	6	IOP check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the Descriptors	
	7	IOP check	Verify that NS has been scaled out by running the end-to-end functional test	
IOP Verdict				

6.2.3.2 TD_NFV_NS_LCM_SCALE_IN_001

Interoperability Test Description					
Identifier	TD_NFV_NS_LCM_SCALE_IN_001				
Test Purpose	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator				
Configuration	SUT_1_NS_1_ENDPOINT_SCALE SUT_1_NS_1_MIDDLEPOINT_SCALE				
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)				
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports terminating virtualised resources • MANO supports triggering scale in with an operator's action • MANO supports scale in by removing VNF instances • NS/VNF supports scale in by removing VNF instances				
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • NS has been scaled out by adding VNF instances				
Test Sequence	**Step**	**Type**	**Description**		**Result**
	1	stimulus	Trigger NS scale in (by removing VNFs) in MANO with an operator action		
	2	IOP check	Verify that the impacted VNF instance(s) have been terminated		
	3	IOP check	Verify that the impacted VNF related resources have been released by the VIM		
	4	IOP check	Verify that the remaining VNF instances(s) are still running and reachable through the management network		
	5	IOP check	Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors		
	6	IOP check	Verify that NS has been scaled in by running the end-to-end functional test		
IOP Verdict					

6.2.3.3   TD_NFV_NS_LCM_SCALE_OUT_002

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_SCALE_OUT_002			
Test Purpose	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by a VNF Indicator			
Configuration	SUT_1_NS_1_ENDPOINT_SCALE SUT_1_NS_1_MIDDLEPOINT_SCALE			
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)			
Applicability	• MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • MANO supports receiving VNF indicators from VNF/EM • VNF/EM can send VNF indicator values to MANO • MANO supports triggering scale out when a given VNF Indicator value crosses a certain threshold • MANO supports scale out by adding VNF instances • NS/VNF supports scale out by adding VNF instances			
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • MANO 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	Trigger the VNF/EM to send the targeted VNF indicator to MANO until the configured threshold is crossed	
	2	IOP check	Verify that the scale out (by adding VNF(s)) procedure has been started in MANO	
	3	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	4	IOP check	Verify that the additional VNF instance(s) have been deployed	
	5	IOP check	Verify that the additional VNF(s) are running and reachable through the management network	
	6	IOP check	Verify that the additional VNF instances(s) have been configured according to VNFD (i.e by obtaining a result from the management interface)	
	7	IOP check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the Descriptors	
	8	IOP check	Verify that NS has been scaled out by running the end-to-end functional test	
IOP Verdict				

6.2.3.4 TD_NFV_NS_LCM_SCALE_IN_002

Interoperability Test Description					
Identifier	TD_NFV_NS_LCM_SCALE_IN_002				
Test Purpose	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by a VNF Indicator				
Configuration	SUT_1_NS_1_ENDPOINT_SCALE SUT_1_NS_1_MIDDLEPOINT_SCALE				
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)				
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports termintaing virtualised resources • MANO supports receiving VNF indicators from VNF/EM • VNF/EM can send VNF indicator values to MANO • MANO supports triggering scale in when a given VNF Indicator value crosses a certain threshold • MANO supports scale in by removing VNF instances • NS/VNF supports scale in by removing VNF instances				
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • NS has been scaled out by adding VNF instances • MANO is configured to trigger SCALE IN (by removing VNF instances) when a given VNF Indicator value crosses a certain threshold				
Test Sequence	**Step**	**Type**	**Description**		**Result**
	1	stimulus	Trigger the VNF/EM to send the targeted VNF indicator to MANO until the configured threshold is crossed		
	2	IOP check	Verify that the scale out (by removing VNF(s)) procedure has been started in MANO		
	3	IOP check	Verify that the impacted VNF instance(s) have been terminated		
	4	IOP check	Verify that the impacted VNF related resources have been released by the VIM		
	5	IOP check	Verify that the remaining VNF instances(s) are still running and reachable through the management network		
	6	IOP check	Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors		
	7	IOP check	Verify that NS has been scaled in by running the end-to-end functional test		
IOP Verdict					

6.2.3.5   TD_NFV_NS_LCM_SCALE_OUT_003

Interoperability Test Description					
Identifier	TD_NFV_NS_LCM_SCALE_OUT_003				
Test Purpose	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by a VIM KPI				
Configuration	SUT_1_NS_1_ENDPOINT_SCALE SUT_1_NS_1_MIDDLEPOINT_SCALE				
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)				
Applicability	• MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • MANO supports receiving VR related KPIs from VIM • VIM_NFVI can send VR related KPIs to MANO • MANO supports triggering scale out when a given KPI crosses a certain treshold • MANO supports scale out by adding VNF instances • NS/VNF supports scale out by adding VNF instances				
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • MANO is configured to trigger SCALE OUT (by adding VNF instances) when a given VIM KPI value crosses a certain threshold				
Test Sequence	**Step**	**Type**	**Description**		**Result**
	1	stimulus	Trigger the VIM to send the targeted KPI to MANO until the configured threshold is crossed		
	2	IOP check	Verify that the scale out (by adding VNF(s)) procedure has been started in MANO		
	3	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors		
	4	IOP check	Verify that the additional VNF instance(s) have been deployed		
	5	IOP check	Verify that the additional VNF(s) are running and reachable through the management network		
	6	IOP check	Verify that the additional VNF instances(s) have been configured according to VNFD (i.e by obtaining a result from the management interface)		
	7	IOP check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the Descriptors		
	8	IOP check	Verify that NS has been scaled out by running the end-to-end functional test		
IOP Verdict					

6.2.3.6 TD_NFV_NS_LCM_SCALE_IN_003

Interoperability Test Description					
Identifier	TD_NFV_NS_LCM_SCALE_IN_003				
Test Purpose	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by a VIM KPI				
Configuration	SUT_1_NS_1_ENDPOINT_SCALE SUT_1_NS_1_MIDDLEPOINT_SCALE				
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)				
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports termintaing virtualised resources • MANO supports receiving VR related KPIs from VIM • VIM_NFVI can send VR related KPIs to MANO • MANO supports triggering scale in when a given KPI crosses a certain threshold • MANO supports scale in by removing VNF instances • NS/VNF supports scale in by removing VNF instances				
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • NS has been scaled out by adding VNF instances • MANO is configured to trigger SCALE IN (by removing VNF instances) when a given VIM KPI value crosses a certain threshold				
Test Sequence	**Step**	**Type**	**Description**		**Result**
	1	stimulus	Trigger the VIM to send the targeted KPI to MANO until the configured threshold is crossed		
	2	IOP check	Verify that the scale out (by removing VNF(s)) procedure has been started in MANO		
	3	IOP check	Verify that the impacted VNF instance(s) have been terminated		
	4	IOP check	Verify that the impacted VNF related resources have been released by the VIM		
	5	IOP check	Verify that the remaining VNF instances(s) are still running and reachable through the management network		
	6	IOP check	Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors		
	7	IOP check	Verify that NS has been scaled in by running the end-to-end functional test		
IOP Verdict					

6.2.3.7   TD_NFV_NS_LCM_SCALE_OUT_004

Interoperability Test Description					
Identifier	TD_NFV_NS_LCM_SCALE_OUT_004				
Test Purpose	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered in MANO by a VNF/EM request				
Configuration	SUT_1_NS_1_ENDPOINT_SCALE SUT_1_NS_1_MIDDLEPOINT_SCALE				
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)				
Applicability	• MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • VIM_NFVI can send scale out request to MANO • MANO supports triggering scale out when the scale out request is received from VNF/EM • MANO supports scale out by adding VNF instances • NS/VNF supports scale out by adding VNF instances				
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)				

Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	stimulus	Trigger the VNF/EM to send a scale out (by adding VNFs) request to MANO	
	2	IOP check	Verify that the scale out (by adding VNF(s)) procedure has been started in MANO	
	3	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	4	IOP check	Verify that the additional VNF instance(s) have been deployed	
	5	IOP check	Verify that the additional VNF(s) are running and reachable through the management network	
	6	IOP check	Verify that the additional VNF instances(s) have been configured according to VNFD (i.e by obtaining a result from the management interface)	
	7	IOP check	Verify that the additional VNF instances(s), VL(s) and VNFFG(s) are connected according to the Descriptors	
	8	IOP check	Verify that NS has been scaled out by running the end-to-end functional test	
IOP Verdict				

6.2.3.8 TD_NFV_NS_LCM_SCALE_IN_004

Interoperability Test Description					
Identifier	TD_NFV_NS_LCM_SCALE_IN_004				
Test Purpose	To verify that a NS can successfully scale in (by removing VNF instances) if triggered in MANO by a VNF/EM request				
Configuration	SUT_1_NS_1_ENDPOINT_SCALE SUT_1_NS_1_MIDDLEPOINT_SCALE				
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)				
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports terminating virtualised resources • VIM_NFVI can send scale in request to MANO • MANO supports triggering scale in when the scale in request is received from VNF/EM • MANO supports scale in by removing VNF instances • NS/VNF supports scale in by removing VNF instances				
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • NS has been scaled out by adding VNF instances				
Test Sequence	**Step**	**Type**	**Description**		**Result**
	1	stimulus	Trigger the VNF/EM to send a scale out (by removing VNFs) request to MANO		
	2	IOP check	Verify that the scale out (by removingVNF(s)) procedure has been started in MANO		
	3	IOP check	Verify that the impacted VNF instance(s) have been terminated		
	4	IOP check	Verify that the impacted VNF related resources have been released by the VIM		
	5	IOP check	Verify that the remaining VNF instances(s) are still running and reachable through the management network		
	6	IOP check	Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors		
	7	IOP check	Verify that NS has been scaled in by running the end-to-end functional test		
IOP Verdict					

6.2.4   SCALE VNF

6.2.4.1   TD_NFV_NS_LCM_SCALE_OUT_VNF_001

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_SCALE_OUT_VNF_001			
Test Purpose	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered by a MANO operator			
Configuration	SUT_1_NS_1_ENDPOINT_SCALE_VNF SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF			
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)			
Applicability	• MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • MANO supports triggering scale out with an operator's action • MANO supports scale out by adding VNFC instances (VMs) • NS/VNF supports scale out by adding VNF instances (VMs)			
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)			

Test Sequence	Step	Type	Description	Result
	1	stimulus	Trigger NS scale out (by adding VNFC instances (VMs) to a VNF in the NS) in MANO with an operator action	
	2	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	3	IOP check	Verify that the additional VM(s) have been deployed (i.e by querying the VIM)	
	4	IOP check	Verify that the additional VM(s) are running and are reachable through the management network	
	5	IOP check	Verify that the additional VM(s) are connected to the VL(s) according to the descriptors	
	6	IOP check	Verify that NS has been scaled out by running the end-to-end functional test	
IOP Verdict				

6.2.4.2   TD_NFV_NS_LCM_SCALE_IN_VNF_001

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_SCALE_IN_VNF_001			
Test Purpose	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered by a MANO operator			
Configuration	SUT_1_NS_1_ENDPOINT_SCALE_VNF SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF			
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)			
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports terminating virtualised resources • MANO supports triggering scale in with an operator's action • MANO supports scale in by removing VNFC instances (VMs) • NS/VNF supports scale in by removing VNFC instances (VMs)			
Pre-test conditions	• NS has been scaled out by adding VM			

Test Sequence	Step	Type	Description	Result

Interoperability Test Description				
	1	stimulus	Trigger NS scale in (by removing VNFC instances (VMs)) in MANO with an operator action	
	2	IOP check	Verify that the impacted VM(s) have been terminated	
	3	IOP check	Verify that the impacted VM related resources have been released by the VIM	
	4	IOP check	Verify that the remaining VM(s) are still running and reachable through the management network	
	5	IOP check	Verify that the remaining VM(s) and VL(s) are still connected according to the descriptors	
	6	IOP check	Verify that NS has been scaled in by running the end-to-end functional test	
IOP Verdict				

6.2.4.3 TD_NFV_NS_LCM_SCALE_OUT_VNF_002

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_SCALE_OUT_VNF_002			
Test Purpose	To verify that a VNF in a NS can be successfully scaled out (by adding VNFV instances (VMs)) when triggered automatically in MANO by a VNF Indicator			
Configuration	SUT_1_NS_1_ENDPOINT_SCALE_VNF SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF			
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)			
Applicability	• MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • MANO supports receiving VNF indicators from VNF/EM • VNF/EM can send VNF indicator values to MANO • MANO supports triggering scale out when a given VNF Indicator value crosses a certain threshold • MANO supports scale out by adding VNFC instances (VMs) • NS/VNF supports scale out by adding VNFC instances (VMs)			
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • MANO is configured to trigger SCALE OUT (by adding VM(s)) when a given VNF Indicator value crosses a certain threshold			

Test Sequence	Step	Type	Description	Result
	1	stimulus	Trigger the VNF/EM to send the targeted VNF indicator to MANO until the configured threshold is crossed	
	2	IOP check	Verify that the scale out (by adding VNFC instances (VMs)) procedure has been started in MANO	
	3	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	4	IOP check	Verify that the additional VM(s) have been deployed (i.e by querying the VIM)	
	5	IOP check	Verify that the additional VM(s) are running and are reachable through the management network	
	6	IOP check	Verify that the additional VM(s) are connected to the VL(s) according to the descriptors	
	7	IOP check	Verify that NS has been scaled out by running the end-to-end functional test	
IOP Verdict				

6.2.4.4 TD_NFV_NS_LCM_SCALE_IN_VNF_002

Interoperability Test Description	
Identifier	TD_NFV_NS_LCM_SCALE_IN_VNF_002
Test Purpose	To verify that a VNF in a NS can be successfully scaled in (by adding VNFC instances (VMs)) when triggered automatically in MANO by a VNF Indicator
Configuration	SUT_1_NS_1_ENDPOINT_SCALE_VNF SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports termintaing virtualised resources • MANO supports receiving VNF indicators from VNF/EM • VNF/EM can send VNF indicator values to MANO • MANO supports triggering scale in when a given VNF Indicator value crosses a certain threshold • MANO supports scale in by removing VNFC instances (VMs) • NS/VNF supports scale in by removing VNFC instances (VMs)
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • NS has been scaled out by adding VM(s) • MANO is configured to trigger SCALE IN (by removing VM(s)) when a given VNF Indicator value crosses a certain threshold

Test Sequence	Step	Type	Description	Result
	1	stimulus	Trigger the VNF/EM to send the targeted VNF indicator to MANO until the configured threshold is crossed	
	2	IOP check	Verify that the scale out (by removing VNFC instances (VMs)) procedure has been started in MANO	
	3	IOP check	Verify that the impacted VM(s) have been terminated	
	4	IOP check	Verify that the impacted VM related resources have been released by the VIM	
	5	IOP check	Verify that the remaining VM(s) are still running and reachable through the management network	
	6	IOP check	Verify that the remaining VM(s) and VL(s) are still connected according to the descriptors	
	7	IOP check	Verify that NS has been scaled in by running the end-to-end functional test	
IOP Verdict				

6.2.4.5   TD_NFV_NS_LCM_SCALE_OUT_VNF_003

Interoperability Test Description					
Identifier	TD_NFV_NS_LCM_SCALE_OUT_VNF_003				
Test Purpose	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered automatically in MANO by a VIM KPI				
Configuration	SUT_1_NS_1_ENDPOINT_SCALE_VNF SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF				
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)				
Applicability	• MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • MANO supports receiving VR related KPIs from VIM • VIM_NFVI can send VR related KPIs to MANO • MANO supports triggering scale out when a given KPI crosses a certain treshold • MANO supports scale out by adding VNFC instances (VMs) • NS/VNF supports scale out by adding VNFC instances (VMs)				
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • MANO is configured to trigger SCALE OUT (by adding VM(s)) when a given VIM KPI value crosses a certain threshold				
Test Sequence	**Step**	**Type**	**Description**		**Result**
	1	stimulus	Trigger NS scale out (by adding VMs to a VNF inside the NS) in MANO with a VIM KPI		
	2	IOP check	Verify that the scale out (by adding VNFC instances (VMs)) procedure has been started in MANO		
	3	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors		
	4	IOP check	Verify that the additional VM(s) have been deployed (i.e by querying the VIM)		
	5	IOP check	Verify that the additional VM(s) are running and are reachable through the management network		
	6	IOP check	Verify that the additional VM(s) are connected to the VL(s) according to the descriptors		
	7	IOP check	Verify that NS has been scaled out by running the end-to-end functional test		
IOP Verdict					

6.2.4.6 TD_NFV_NS_LCM_SCALE_IN_VNF_003

Interoperability Test Description					
Identifier	TD_NFV_NS_LCM_SCALE_IN_VNF_003				
Test Purpose	To verify that a VNF in a NS can be successfully scaled in (by adding VNFC instances (VMs)) when triggered automatically in MANO by a VIM KPI				
Configuration	SUT_1_NS_1_ENDPOINT_SCALE_VNF SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF				
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)				
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports termintaing virtualised resources • MANO supports receiving VR related KPIs from VIM • VIM_NFVI can send VR related KPIs to MANO • MANO supports triggering scale in when a given KPI crosses a certain threshold • MANO supports scale in by removing VNFC instances (VMs) • NS/VNF supports scale in by removing VNFC instances (VMs)				
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • NS has been scaled out by adding VM(s) • MANO is configured to trigger SCALE IN (by removing VM(s)) when a given VIM KPI value crosses a certain threshold				

Test Sequence	Step	Type	Description		Result
	1	stimulus	Trigger NS scale in (by removing VNFC instances (VMs)) in MANO with a VIM KPI		
	2	IOP check	Verify that the scale out (by removing VM(s)) procedure has been started in MANO		
	3	IOP check	Verify that the impacted VM(s) have been terminated		
	4	IOP check	Verify that the impacted VM related resources have been released by the VIM		
	5	IOP check	Verify that the remaining VM(s) are still running and reachable through the management network		
	6	IOP check	Verify that the remaining VM(s) and VL(s) are still connected according to the descriptors		
	7	IOP check	Verify that NS has been scaled in by running the end-to-end functional test		
IOP Verdict					

6.2.4.7 TD_NFV_NS_LCM_SCALE_OUT_VNF_004

Interoperability Test Description					
Identifier	TD_NFV_NS_LCM_SCALE_OUT_VNF_004				
Test Purpose	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered in MANO by a VNF/EM request				
Configuration	SUT_1_NS_1_ENDPOINT_SCALE_VNF SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF				
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)				
Applicability	• MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • VIM_NFVI can send scale out request to MANO • MANO supports triggering scale out when the scale out request is received from VNF/EM • MANO supports scale out by adding VNFC instances (VMs) • NS/VNF supports scale out by adding VNFC instances (VMs)				
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)				

Test Sequence	Step	Type	Description	Result
	1	stimulus	Trigger the VNF/EM to send a scale out (by adding VNFC instances (VMs)) request to MANO	
	2	IOP check	Verify that the scale out (by adding VM(s)) procedure has been started in MANO	
	3	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors	
	4	IOP check	Verify that the additional VM(s) have been deployed (i.e by querying the VIM)	
	5	IOP check	Verify that the additional VM(s) are running and are reachable through the management network	
	6	IOP check	Verify that the additional VM(s) are connected to the VL(s) according to the descriptors	
	7	IOP check	Verify that NS has been scaled out by running the end-to-end functional test	
IOP Verdict				

6.2.4.8 TD_NFV_NS_LCM_SCALE_IN_VNF_004

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_SCALE_IN_VNF_004			
Test Purpose	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered in MANO by a VNF/EM request			
Configuration	SUT_1_NS_1_ENDPOINT_SCALE_VNF SUT_1_NS_1_MIDDLEPOINT_SCALE_VNF			
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.2.3, 6.3.3)			
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports terminating virtualised resources • VIM_NFVI can send scale in request to MANO • MANO supports triggering scale in when the scale in request is received from VNF/EM • MANO supports scale in by removing VNFC instances (VMs) • NS/VNF supports scale in by removing VNFC instances (VMs)			
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • NS has been scaled out by adding VM(s)			

Test Sequence	Step	Type	Description	Result
	1	stimulus	Trigger the VNF/EM to send a scale out (by removing VNFC instances (VMs)) request to MANO	
	2	IOP check	Verify that the scale out (by removing VM(s)) procedure has been started in MANO	
	3	IOP check	Verify that the impacted VM(s) have been terminated	
	4	IOP check	Verify that the impacted VM related resources have been released by the VIM	
	5	IOP check	Verify that the remaining VM(s) are still running and reachable through the management network	
	6	IOP check	Verify that the remaining VM(s) and VL(s) are still connected according to the descriptors	
	7	IOP check	Verify that NS has been scaled in by running the end-to-end functional test	
IOP Verdict				

6.2.5 UPDATE

6.2.5.1 TD_NFV_NS_LCM_UPDATE_STOP_VNF_001

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_UPDATE_STOP_VNF_001			
Test Purpose	To verify that a VNF running in a NS can be successfully stopped by MANO			
Configuration	SUT_1_NS_1_ENDPOINT SUT_1_NS_1_MIDDLEPOINT			
References	ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.5)			
Applicability	• MANO can request VIM_NFVI to stop VM(s) • VIM_NFVI supports stoping running VM(s)			
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • VNF instance(s) in the NS are running			

Test Sequence	Step	Type	Description	Result
	1	stimulus	Trigger the VNF(s) stop operation in MANO	

Interoperability Test Description				
	2	IOP check	Verify the VNF(s) state inside the NS is "Stopped" on MANO (query, display, ...)	
	3	IOP check	Verify that individual VM(s) inside the VNF(s) are stopped on VIM (i.e query or display the state from VIM)	
IOP Verdict				

6.2.5.2   TD_NFV_NS_LCM_UPDATE_START_VNF_001

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_UPDATE_START_VNF_001			
Test Purpose	To verify that a stopped VNF in a NS can be successfully re-started by MANO			
Configuration	SUT_1_NS_1_ENDPOINT SUT_1_NS_1_MIDDLEPOINT			
References	ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.5)			
Applicability	• MANO can request VIM_NFVI to re-start VM(s) • VIM_NFVI supports re-starting VM(s)			
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • One VNF inside the NS has been stopped (TD_NFV_NS_LCM_UPDATE_STOP_VNF_001)			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	stimulus	Trigger the VNF(s) start operation in MANO	
	2	IOP check	Verify the VNF(s) state inside the NS is "Started" on MANO (i.e query, display, ...)	
	3	IOP check	Verify that individual VM(s) inside the VNF(s) are started on VIM (i.e query or display the state from VIM)	
	4	IOP check	Verify that the NS is successfully recovered by running the end-to-end functional test	
IOP Verdict				

6.2.5.3   TD_NFV_NS_LCM_UPDATE_ADD_VNF_VL_001

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_UPDATE_ADD_VNF_VL_001			
Test Purpose	To verify that VNF(s) and VL(s) can be added to a running NS			
Configuration	SUT_1_NS_1_MIDDLEPOINT_UPDATE			
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.3.4)			
Applicability	• MANO can request VIM_NFVI to allocate virtualised resources • VIM_NFVI supports allocating virtualised resources • MANO supports triggering NS update with an operator's action • MANO supports adding VNFs to a running NS • MANO supports adding VLs to a running NS			
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • Test VNF can loadshare traffic among several VNFs			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	stimulus	The MANO operator triggers NS update adding VNF(s) and VL(s) to a running NS	

Interoperability Test Description			
	2	IOP check	Verify that the requested resources have been allocated by the VIM according to the descriptors
	3	IOP check	Verify that the additional VNF(s) and VL(s) have been deployed
	4	IOP check	Verify that the additional VNF(s) are running and reachable through the management network
	5	IOP check	Verify that the additional VNF(s) are connected to the VL(s) according to the descriptors
	6	IOP check	Verify that NS has been updated by running the end-to-end functional test
IOP Verdict			

6.2.5.4   TD_NFV_NS_LCM_UPDATE_REM_VNF_VL_001

Interoperability Test Description	
Identifier	TD_NFV_NS_LCM_UPDATE_REM_VNF_VL_001
Test Purpose	To verify that VNF(s) and VL(s) can be removed from a running NS
Configuration	SUT_1_NS_1_MIDDLEPOINT_UPDATE
References	ETSI GS NFV-IFA005 V2.1.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.1.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.3.4)
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports terminating virtualised resources • MANO supports triggering NS update with an operator's action • MANO supports removing VNF(s) from a running NS • MANO supports removing VL(s) from a running NS
Pre-test conditions	• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) • NS has been updated by adding VNF(s) and VL(s) (TD_NFV_NS_LCM_UPDATE_ADD_VNF_VL_001)

Test Sequence	Step	Type	Description	Result
	1	stimulus	The MANO operator triggers NS update removing VNF(s) and VL(s) from a running NS	
	2	IOP check	Verify that the specified VNF(s) and VL(s) have been terminated	
	3	IOP check	Verify that the impacted resources have been released by the VIM	
	4	IOP check	Verify that the remaining VNF(s) are still running and reachable	
	5	IOP check	Verify that the remaining VNF (s) and VL(s) are still connected according to the descriptors	
	6	IOP check	Verify that NS has been updated by running the end-to-end functional test	
IOP Verdict				

6.2.5   TERMINATE

6.5.1.1   TD_NFV_NS_LCM_TERMINATE_001

Interoperability Test Description				
Identifier	TD_NFV_NS_LCM_TERMINATE_001			
Test Purpose	To verify that a NS can be successfully terminated			
Configuration	SUT_1_NS_1_ENDPOINT SUT_1_NS_1_MIDDLEPOINT			
References	ETSI GS NFV-IFA013 V2.1.1 (clause 7.3.7) ETSI GS NFV-IFA005 V2.1.1 (clause 7.2.6) ETSI GS NFV-IFA008 V2.1.1 (clause 7.2.7) ETSI GS NFV-IFA010 V2.1.1 (clause 6.3.5)			
Applicability	• MANO can request VIM_NFVI to terminate virtualised resources • VIM_NFVI supports terminating virtualised resources • (If required by NSD) MANO can request VIM_NFVI to delete NFP(s) • (If required by NSD) VIM_NFVI supports deleting NFP(s)			
Pre-test conditions	• NS has been instantiated			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	stimulus	Trigger NS termination in MANO	
	2	IOP check	Verify that all the VNF instance(s) have been terminated	
	3	IOP check	Verify that the resources that were allocated to the NS and VNF(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 NS instance does no longer exist	
IOP Verdict				

6.3 TEARDOWN

6.3.1 DELETE

6.3.1.1 TD_NFV_TEARDOWN_DELETE_NSD_001

Interoperability Test Description				
Identifier	TD_NFV_TEARDOWN_DELETE_NSD_001			
Test Purpose	To delete a NSD			
Configuration	SUT_1_NS_1_ENDPOINT SUT_1_NS_1_MIDDLEPOINT			
References	ETSI GS NFV-IFA013 V2.1.1 (clause 7.2.6)			
Applicability				
Pre-test conditions	• NSD is created in MANO (TD_NFV_SETUP_CREATE_NSD_001) • All NSs associated with the NSD have been terminated			
Test Sequence	**Step**	**Type**	**Description**	**Result**
	1	stimulus	Trigger the deletion of NSD on MANO (i.e using tools produced by MANO)	
	2	IOP check	Verify that the NSD and referenced VLD(s) and VNFFGD(s) no longer exists on MANO	
IOP Verdict				

6.3.1.2   TD_NFV_TEARDOWN_DELETE_VNF_PKG_001

Interoperability Test Description				
Identifier	TD_NFV_TEARDOWN_DELETE_VNF_PKG_001			
Test Purpose	To delete a VNF Package			
Configuration	SUT_1_NS_1_ENDPOINT SUT_1_NS_1_MIDDLEPOINT			
References	ETSI GS NFV-IFA013 V2.1.1 (clause 7.7) ETSI GS NFV-IFA010 V2.1.1 (clauses 6.5.1, 7.4.1)			
Applicability				
Pre-test conditions	• VNF package has been on-boarded in MANO (TD_NFV_SETUP_VNF_PKG_ONB_001)			

Test Sequence	Step	Type	Description	Result
	1	stimulus	Trigger the deletion of the VNF package on MANO	
	2	IOP check	Verify that the VNF Package information has been deleted from MANO	
IOP Verdict				

Annex A:Interoperability Feature Statement Pro-Forma

A.1   IFS for MANO

IFS ID	Description	Support
IFS_MANO_SWIM_ADD	MANO can request VIM to add a SW image	
IFS_MANO_VRM_ALLOCATE	MANO can request VIM to allocate virtualised resources	
IFS_MANO_VRM_TERMINATE	MANO can request VIM to terminate virtualised resources	
IFS_MANO_NFPM_CREATE	MANO can request VIM to create NFP(s)	
IFS_MANO_NFPM_DELETE	MANO can request VIM to delete NFP(s)	
IFS_MANO_SCALE_OUT	MANO supports scale out by adding VNF instances	
IFS_MANO_SCALE_OUT_VNF	MANO supports scale out by adding VNFC instances	
IFS_MANO_SCALE_IN	MANO supports scale in by removing VNF instances	
IFS_MANO_SCALE_IN_VNF	MANO supports scale in by removing VNFC instances	
IFS_MANO_SCALE_OUT_BY_OPERATOR	MANO supports triggering scale out with an operator's action	
IFS_MANO_SCALE_IN_BY_OPERATOR	MANO supports triggering scale in with an operator's action	
IFS_MANO_RECEIVE_VNF_INDICATOR	MANO supports receiving VNF indicators from VNF/EM	
IFS_MANO_SCALE_OUT_BY_VNF_INDICATOR	MANO supports triggering scale out when a given VNF Indicator value crosses a certain threshold	
IFS_MANO_SCALE_IN_BY_VNF_INDICATOR	MANO supports triggering scale in when a given VNF Indicator value crosses a certain threshold	
IFS_MANO_RECEIVE_VIM_KPI	MANO supports receiving VR related KPIs from VIM	
IFS_MANO_SCALE_OUT_BY_VIM_KPI	MANO supports triggering scale out when a given KPI crosses a certain threshold	
IFS_MANO_SCALE_IN_BY_VIM_KPI	MANO supports triggering scale in when a given KPI crosses a certain threshold	
IFS_MANO_SCALE_OUT_BY_VNF_REQUEST	MANO supports triggering scale out when the scale out request is received from VNF/EM	
IFS_MANO_SCALE_IN_BY_VNF_REQUEST	MANO supports triggering scale in when the scale in request is received from VNF/EM	
IFS_MANO_STOP_VNFC	MANO can request VIM to stop VNFC (VMs)	
IFS_MANO_RESTART_VNFC	MANO can request VIM to re-start VNFC (VM)	
IFS_MANO_NS_UPDATE_OPERATOR	MANO supports triggering NS update with an operator's action	
IFS_MANO_ADD_VNF_RUNNING_NS	MANO supports adding VNFs to a running NS	
IFS_MANO_REMOVE_VNF_RUNNING_NS	MANO supports removing VNF(s) from a running NS	
IFS_MANO_ADD_VL_RUNNING_NS	MANO supports adding VLs to a running NS	
IFS_MANO_REMOVE_VL_RUNNING_NS	MANO supports removing VL(s) from a running NS	

A.2   IFS for VIM/NFVI

IFS ID	Description	Support
IFS_VIM_SWIM_ADD	VIM supports adding a SW image	
IFS_VIM_VRM_ALLOCATE	VIM supports allocating virtualised resources	
IFS_VIM_VRM_TERMINATE	VIM supports terminating virtualised resources	
IFS_VIM_NFPM_CREATE	VIM supports creating NFP(s)	
IFS_VIM_NFPM_DELETE	VIM supports deleting NFP(s)	
IFS_VIM_SEND_KPI	VIM can send VR related KPIs to MANO	
IFS_VIM_STOP_VNFC	VIM supports stoping running VNFCs (VMs)	
IFS_VIM_RESTART_VNFC	VIM supports re-starting VNFCs (VMs)	

A.3   IFS for NS/VNF/EM

IFS ID	Description	Support
IFS_VNF_NS_SCALE_OUT	NS/VNF supports scale out by adding VNF instances	
IFS_VNF_NS_SCALE_OUT_VNF	NS/VNF supports scale out by adding VNFC instances	
IFS_VNF_NS_SCALE_IN	NS/VNF supports scale in by removing VNF instances	
IFS_VNF_NS_SCALE_IN_VNF	NS/VNF supports scale in by removing VNFC instances	
IFS_VNF_EM_SEND_VNF_INDICATOR	VNF/EM can send VNF indicator values to MANO	
IFS_VNF_EM_SEND_SCALE_OUT_REQUEST	VNF/EM can send scale out request to MANO	
IFS_VNF_EM_SEND_SCALE_IN_REQUEST	VNF/EM can send scale in request to MANO	