

**3<sup>rd</sup> ETSI NFV Plugtests  
Sophia Antipolis, France**

---



**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/chaicor/ETSI\\_support.asp](http://portal.etsi.org/chaicor/ETSI_support.asp)

---

**Copyright Notification**

---

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI. The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2018.  
All rights reserved.

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

# Contents

Foreword.....	8
Introduction .....	8
1 Scope.....	9
2 References .....	9
2.1 Normative references .....	9
2.2 Informative references .....	9
3 Definitions, symbols and abbreviations .....	10
3.1 Definitions .....	10
3.2 Symbols .....	10
3.3 Abbreviations.....	10
4 Test Suite Structure .....	10
4.1 Naming Convention .....	10
4.2 Test Summary.....	16
4.2.1 Single Vendor NS .....	16
4.2.1.0 SUT Configuration .....	16
4.2.1.1 ONBOARD .....	16
4.2.1.2 INSTANTIATE.....	16
4.2.1.3 SCALE NS .....	16
4.2.1.3.1 SCALE NS MANUALLY .....	16
4.2.1.3.2 SCALE NS FROM VNF INDICATOR.....	16
4.2.1.3.3 SCALE NS FROM VIM KPI .....	17
4.2.1.3.4 SCALE NS FROM VNF REQ.....	17
4.2.1.4 SCALE VNF .....	17
4.2.1.4.1 SCALE VNF MANUALLY .....	17
4.2.1.4.2 SCALE VNF FROM VNF INDICATOR.....	17
4.2.1.4.3 SCALE VNF FROM VIM KPI .....	17
4.2.1.4.4 SCALE VNF FROM VNF REQ.....	18
4.2.1.5 SCALE NS TO LEVEL .....	18
4.2.1.5.1 SCALE NS TO LEVEL MANUALLY .....	18
4.2.1.5.2 SCALE NS TO LEVEL FROM VNF INDICATOR.....	18
4.2.1.5.3 SCALE NS TO LEVEL FROM VIM KPI .....	18
4.2.1.6 SCALE VNF TO LEVEL.....	18
4.2.1.6.1 SCALE VNF TO LEVEL MANUALLY .....	18
4.2.1.6.2 SCALE VNF TO LEVEL FROM VNF INDICATOR.....	18
4.2.1.6.3 SCALE VNF TO LEVEL FROM VIM KPI.....	19
4.2.1.7 UPDATE VNF .....	19
4.2.1.8 FAULT MANAGEMENT .....	19
4.2.1.8.1 FAULT MANAGEMENT - VR.....	19
4.2.1.8.2 FAULT MANAGEMENT - VNF.....	19
4.2.1.9 PERFORMANCE MANAGEMENT .....	19
4.2.1.9.1 PERFORMANCE MANAGEMENT - VR .....	19
4.2.1.9.2 PERFORMANCE MANAGEMENT – VNF KPI.....	20
4.2.1.10 TERMINATE .....	20
4.2.1.11 DELETE .....	20
4.2.2 Multi Vendor NS.....	20
4.2.2.0 SUT Configuration .....	20
4.2.2.1 ONBOARD .....	20
4.2.2.2 INSTANTIATE.....	20
4.2.2.3 SCALE NS .....	21
4.2.2.3.1 SCALE NS MANUALLY .....	21
4.2.2.3.2 SCALE NS FROM VNF INDICATOR.....	21
4.2.2.3.3 SCALE NS FROM VIM KPI .....	21
4.2.2.3.4 SCALE NS FROM VNF REQ.....	21
4.2.2.4 SCALE VNF .....	21
4.2.2.4.1 SCALE VNF MANUALLY .....	21

4.2.2.4.2	SCALE VNF FROM VNF INDICATOR.....	22
4.2.2.4.3	SCALE VNF FROM VIM KPI .....	22
4.2.2.4.4	SCALE VNF FROM VNF REQ.....	22
4.2.2.5	SCALE NS TO LEVEL .....	22
4.2.2.5.1	SCALE NS TO LEVEL MANUALLY .....	22
4.2.2.5.2	SCALE NS TO LEVEL FROM VNF INDICATOR.....	23
4.2.2.5.3	SCALE NS TO LEVEL FROM VIM KPI .....	23
4.2.2.6	SCALE VNF TO LEVEL.....	23
4.2.2.6.1	SCALE VNF TO LEVEL MANUALLY .....	23
4.2.2.6.2	SCALE VNF TO LEVEL FROM VNF INDICATOR.....	23
4.2.2.6.3	SCALE VNF TO LEVEL FROM VIM KPI.....	23
4.2.2.7	UPDATE VNF .....	23
4.2.2.8	FAULT MANAGEMENT .....	23
4.2.2.8.1	FAULT MANAGEMENT - VR.....	23
4.2.2.8.2	FAULT MANAGEMENT - VNF.....	24
4.2.2.9	PERFORMANCE MANAGEMENT.....	24
4.2.2.9.1	PERFORMANCE MANAGEMENT - VR .....	24
4.2.2.9.2	PERFORMANCE MANAGEMENT – VNF KPI.....	24
4.2.2.10	TERMINATE .....	24
4.2.2.11	DELETE .....	25
4.2.3	Multi Vendor NS with EPA .....	25
4.2.3.0	SUT Configuration .....	25
4.2.3.1	ONBOARD .....	25
4.2.3.2	INSTANTIATE.....	25
4.2.3.3	SCALE NS MANUALLY.....	25
4.2.3.4	SCALE VNF MANUALLY.....	25
4.2.3.5	TERMINATE .....	26
4.2.3.6	DELETE .....	26
4.2.4	Multi Vendor NS with SFC.....	26
4.2.4.0	SUT Configuration .....	26
4.2.4.1	ONBOARD .....	26
4.2.4.2	INSTANTIATE.....	26
4.2.4.3	TERMINATE .....	26
4.2.4.4	DELETE .....	26
4.2.5	Multi Site.....	27
4.2.5.0	SUT Configuration .....	27
4.2.5.1	ONBOARD .....	27
4.2.5.2	INSTANTIATE.....	27
4.2.5.3	SCALE NS MANUALLY.....	27
4.2.5.4	SCALE VNF MANUALLY.....	27
4.2.5.5	FAULT MANAGEMENT .....	27
4.2.5.5.1	FAULT MANAGEMENT – VR .....	27
4.2.5.5.2	FAULT MANAGEMENT – VNF.....	28
4.2.5.6	PERFORMANCE MANAGEMENT.....	28
4.2.5.6.1	PERFORMANCE MANAGEMENT – VR.....	28
4.2.5.6.2	PERFORMANCE MANAGEMENT – VNF .....	28
4.2.5.7	TERMINATE .....	28
4.2.5.8	DELETE .....	28
4.2.6	S-VNFM-D .....	29
4.2.6.0	SUT Configuration .....	29
4.2.6.1	ONBOARD .....	29
4.2.6.2	INSTANTIATE.....	29
4.2.6.3	SCALE NS MANUALLY.....	29
4.2.6.4	SCALE VNF MANUALLY.....	29
4.2.6.5	PERFORMANCE MANAGEMENT.....	29
4.2.6.5.1	PERFORMANCE MANAGEMENT – VNF VR.....	29
4.2.6.5.2	PERFORMANCE MANAGEMENT – VNF KPI.....	30
4.2.6.6	FAULT MANAGEMENT .....	30
4.2.6.6.1	FAULT MANAGEMENT – VNF VR .....	30
4.2.6.6.2	FAULT MANAGEMENT – VNF.....	30
4.2.6.7	TERMINATE .....	30
4.2.6.8	DELETE .....	30

4.2.7	S-VNFM-I.....	31
4.2.7.0	SUT Configuration .....	31
4.2.7.1	ONBOARD .....	31
4.2.7.2	INSTANTIATE.....	31
4.2.7.3	SCALE NS MANUALLY.....	31
4.2.7.4	SCALE VNF MANUALLY.....	31
4.2.7.5	PERFORMANCE MANAGEMENT.....	31
4.2.7.5.1	PERFORMANCE MANAGEMENT – VR.....	31
4.2.7.5.2	PERFORMANCE MANAGEMENT – VNF KPI.....	32
4.2.7.6	FAULT MANAGEMENT .....	32
4.2.7.6.1	FAULT MANAGEMENT – VR .....	32
4.2.7.6.2	FAULT MANAGEMENT – VNF.....	32
4.2.7.7	TERMINATE.....	32
4.2.7.8	DELETE.....	32
4.2.8	Auto LCM Validation .....	33
4.2.8.0	SUT Configuration .....	33
4.2.8.1	ONBOARD .....	33
4.2.8.2	INSTANTIATE.....	33
4.2.8.3	SCALE NS MANUALLY.....	33
4.2.8.4	SCALE VNF MANUALLY.....	33
4.2.8.5	UPDATE VNF .....	33
4.2.8.6	FAULT MANAGEMENT – VR.....	33
4.2.8.7	TERMINATE.....	34
4.2.8.8	DELETE.....	34
5	System Under Test Configurations .....	35
5.1	SUT_SINGLE-VENDOR_NS.....	35
5.2	SUT_MULTI-VENDOR_NS .....	36
5.3	SUT_AUTO-LCM-VALIDATION.....	37
5.4	SUT_MULTI-SITE .....	38
5.5	SUT_S-VNFM-D .....	39
5.6	SUT_S-VNFM-I.....	40
6	Interoperability Test Descriptions.....	41
6.1	NS .....	41
6.1.1	ONBOARD.....	41
6.1.1.1	TD_NFV_ONBOARD_VNF_PKG_001 .....	41
6.1.1.2	TD_NFV_ONBOARD_NSD_001 .....	41
6.1.2	INSTANTIATE.....	42
6.1.2.1	TD_NFV_NS_LCM_INSTANTIATE_001 .....	42
6.1.3	SCALE NS.....	43
6.1.3.1	SCALE NS MANUALLY.....	43
6.1.3.1.1	TD_NFV_NS_LCM_SCALE_OUT_001.....	43
6.1.3.1.2	TD_NFV_NS_LCM_SCALE_IN_001.....	43
6.1.3.2	SCALE NS FROM VNF INDICATOR .....	44
6.1.3.2.1	TD_NFV_NS_LCM_SCALE_OUT_002a.....	44
6.1.3.2.2	TD_NFV_NS_LCM_SCALE_IN_002a.....	45
6.1.3.2.3	TD_NFV_NS_LCM_SCALE_OUT_002b.....	46
6.1.3.2.4	TD_NFV_NS_LCM_SCALE_IN_002b.....	47
6.1.3.3	SCALE NS FROM VIM KPI.....	47
6.1.3.3.1	TD_NFV_NS_LCM_SCALE_OUT_003.....	47
6.1.3.3.2	TD_NFV_NS_LCM_SCALE_IN_003.....	48
6.1.3.4	SCALE NS FROM VNF REQUEST .....	49
6.1.3.4.1	TD_NFV_NS_LCM_SCALE_OUT_004.....	49
6.1.3.4.2	TD_NFV_NS_LCM_SCALE_IN_004.....	50
6.1.4	SCALE VNF .....	51
6.1.4.1	SCALE VNF MANUALLY.....	51
6.1.4.1.1	TD_NFV_NS_LCM_SCALE_OUT_VNF_001 .....	51
6.1.4.1.2	TD_NFV_NS_LCM_SCALE_IN_VNF_001.....	51
6.1.4.2	SCALE VNF FROM VNF INDICATOR .....	52
6.1.4.2.1	TD_NFV_NS_LCM_SCALE_OUT_VNF_002a.....	52
6.1.4.2.2	TD_NFV_NS_LCM_SCALE_IN_VNF_002a.....	53

6.1.4.2.3	TD_NFV_NS_LCM_SCALE_OUT_VNF_002b.....	54
6.1.4.2.4	TD_NFV_NS_LCM_SCALE_IN_VNF_002b.....	54
6.1.4.3	SCALE VNF FROM VIM KPI.....	55
6.1.4.3.1	TD_NFV_NS_LCM_SCALE_OUT_VNF_003.....	55
6.1.4.3.2	TD_NFV_NS_LCM_SCALE_IN_VNF_003.....	56
6.1.4.4	SCALE VNF FROM VNF REQUEST.....	57
6.1.4.4.1	TD_NFV_NS_LCM_SCALE_OUT_VNF_004.....	57
6.1.4.4.2	TD_NFV_NS_LCM_SCALE_IN_VNF_004.....	58
6.1.5	SCALE NS TO LEVEL.....	58
6.1.5.1	SCALE NS TO LEVEL MANUALLY.....	58
6.1.5.1.1	TD_NFV_NS_LCM_SCALE_TO_LEVEL_001.....	58
6.1.5.2	SCALE NS TO LEVEL FROM VNF INDICATOR.....	59
6.1.5.2.1	TD_NFV_NS_LCM_SCALE_TO_LEVEL_002.....	59
6.1.5.3	SCALE NS TO LEVEL FROM VIM KPI.....	60
6.1.5.3.1	TD_NFV_NS_LCM_SCALE_TO_LEVEL_003.....	60
6.1.6	SCALE VNF TO LEVEL.....	61
6.1.6.1	SCALE VNF TO LEVEL MANUALLY.....	61
6.1.6.1.1	TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_001.....	61
6.1.6.2	SCALE VNF TO LEVEL FROM VNF INDICATOR.....	62
6.1.6.2.1	TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_002.....	62
6.1.6.3	SCALE VNF TO LEVEL FROM VIM KPI.....	63
6.1.6.3.1	TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_003.....	63
6.1.7	UPDATE VNF.....	64
6.1.7.1	TD_NFV_NS_LCM_UPDATE_STOP_VNF_001.....	64
6.1.7.2	TD_NFV_NS_LCM_UPDATE_START_VNF_001.....	65
6.1.8	FAULT MANAGEMENT.....	65
6.1.8.1	FAULT MANAGEMENT – VR.....	65
6.1.8.1.1	TD_NFV_FM_VR_ALARM_001.....	65
6.1.8.1.2	TD_NFV_FM_VR_CLEAR_001.....	66
6.1.8.2	FAULT MANAGEMENT – VNF.....	67
6.1.8.2.1	TD_NFV_FM_VNF_ALARM_001.....	67
6.1.8.2.2	TD_NFV_FM_VNF_CLEAR_001.....	67
6.1.9	PERFORMANCE MANAGEMENT.....	68
6.1.9.1	PERFORMANCE MANAGEMENT – VR.....	68
6.1.9.1.1	TD_NFV_PM_VR_CREATE_MONITOR_001.....	68
6.1.9.1.2	TD_NFV_PM_VR_CREATE_THRESHOLD_001.....	69
6.1.9.1.3	TD_NFV_PM_VR_DELETE_MONITOR_001.....	69
6.1.9.1.4	TD_NFV_PM_VR_DELETE_THRESHOLD_001.....	70
6.1.9.2	PERFORMANCE MANAGEMENT – VNF KPI.....	71
6.1.9.2.1	TD_NFV_PM_VNF_KPI_CREATE_MONITOR_001.....	71
6.1.9.2.2	TD_NFV_PM_VNF_KPI_DELETE_MONITOR_001.....	71
6.1.9.2.3	TD_NFV_PM_VNF_KPI_CREATE_THRESHOLD_001.....	72
6.1.9.2.4	TD_NFV_PM_VNF_KPI_DELETE_THRESHOLD_001.....	73
6.1.10	TERMINATE.....	73
6.1.10.1	TD_NFV_NS_LCM_TERMINATE_001.....	73
6.1.11	DELETE.....	74
6.1.11.1	TD_NFV_TEARDOWN_DELETE_NSD_001.....	74
6.1.11.2	TD_NFV_TEARDOWN_DELETE_VNF_PKG_001.....	75
6.2	EPA.....	75
6.2.1	INSTANTIATE.....	75
6.2.1.1	TD_NFV_EPA_NS_LCM_INSTANTIATE_001.....	75
6.2.2	SCALE NS.....	76
6.2.2.1	SCALE NS MANUALLY.....	76
6.2.2.1.1	TD_NFV_EPA_NS_LCM_SCALE_OUT_001.....	76
6.2.2.1.2	TD_NFV_EPA_NS_LCM_SCALE_IN_001.....	77
6.2.3	SCALE VNF.....	78
6.2.3.1	SCALE VNF MANUALLY.....	78
6.2.3.1.1	TD_NFV_EPA_NS_LCM_SCALE_OUT_VNF_001.....	78
6.2.3.1.2	TD_NFV_EPA_NS_LCM_SCALE_IN_VNF_001.....	79
6.3	SFC.....	80
6.3.1	INSTANTIATE.....	80
6.3.1.1	TD_NFV_SFC_NS_LCM_INSTANTIATE_001.....	80

6.3.2	TERMINATE.....	81
6.3.2.1	TD_NFV_SFC_NS_LCM_TERMINATE_001 .....	81
6.4	MULTI SITE .....	81
6.4.1	INSTANTIATE.....	81
6.4.1.1	TD_NFV_MULTISITE_NS_LCM_INSTANTIATE_001 .....	81
6.4.2	SCALE NS MANUALLY .....	82
6.4.2.1	TD_NFV_MULTISITE_NS_LCM_SCALE_OUT_001 .....	82
6.4.2.2	TD_NFV_MULTISITE_NS_LCM_SCALE_IN_001 .....	83
6.4.3	SCALE VNF MANUALLY .....	84
6.4.3.1	TD_NFV_MULTISITE_NS_LCM_SCALE_OUT_VNF_001.....	84
6.4.3.2	TD_NFV_MULTISITE_NS_LCM_SCALE_IN_VNF_001.....	85
6.4.4	TERMINATE.....	86
6.4.4.1	TD_NFV_MULTISITE_NS_LCM_TERMINATE_001 .....	86
6.5	s-VNFM-D.....	86
6.5.1	INSTANTIATE.....	86
6.5.1.1	TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001 .....	86
6.5.2	SCALE NS .....	87
6.5.2.1	SCALE NS MANUALLY .....	87
6.5.2.1.1	TD_NFV_S-VNFM-D_NS_LCM_SCALE_OUT_001.....	87
6.5.2.1.2	TD_NFV_S-VNFM-D_NS_LCM_SCALE_IN_001.....	88
6.5.3	SCALE VNF .....	89
6.5.3.1	SCALE VNF MANUALLY .....	89
6.5.3.1.1	TD_NFV_S-VNFM-D_NS_LCM_SCALE_OUT_VNF_001.....	89
6.5.3.1.2	TD_NFV_S-VNFM-D_NS_LCM_SCALE_IN_VNF_001.....	89
6.5.4	PERFORMANCE MANAGEMENT.....	90
6.5.4.1	PERFORMANCE MANAGEMENT – VNF VR .....	90
6.5.4.1.1	TD_NFV_S-VNFM-D_PM_VNF_VR_CREATE_MONITOR_001 .....	90
6.5.4.1.2	TD_NFV_S-VNFM-D_PM_VNF_VR_CREATE_THRESHOLD_001.....	91
6.5.4.1.3	TD_NFV_S-VNFM-D_PM_VNF_VR_DELETE_MONITOR_001 .....	92
6.5.4.1.4	TD_NFV_S-VNFM-D_PM_VNF_VR_DELETE_THRESHOLD_001 .....	92
6.5.4.2	PERFORMANCE MANAGEMENT – VNF KPI.....	93
6.5.4.2.1	TD_NFV_S-VNFM-D_PM_VNF_KPI_CREATE_MONITOR_001 .....	93
6.5.4.2.2	TD_NFV_S-VNFM-D_PM_VNF_KPI_DELETE_MONITOR_001 .....	94
6.5.4.2.3	TD_NFV_S-VNFM-D_PM_VNF_KPI_CREATE_THRESHOLD_001 .....	94
6.5.4.2.4	TD_NFV_S-VNFM-D_PM_VNF_KPI_DELETE_THRESHOLD_001 .....	95
6.5.5	FAULT MANAGEMENT .....	96
6.5.5.1	FAULT MANAGEMENT – VNF VR .....	96
6.5.5.1.1	TD_NFV_S-VNFM-D_FM_VNF_VR_ALARM_001.....	96
6.5.5.1.2	TD_NFV_S-VNFM-D_FM_VNF_VR_CLEAR_001.....	97
6.5.5.2	FAULT MANAGEMENT – VNF.....	98
6.5.5.2.1	TD_NFV_S-VNFM-D_FM_VNF_ALARM_001 .....	98
6.5.5.2.2	TD_NFV_S-VNFM-D_FM_VNF_CLEAR_001 .....	98
	Annex A:Interoperability Feature Statements .....	99
A.1	IFS for MANO.....	99
A.2	IFS for VIM/NFVI.....	100
A.3	IFS for VNF .....	100
A.4	IFS for VNFM .....	101
	History .....	102

---

## Foreword

This Test Plan has been produced by ETSI Centre for Testing and Interoperability during the preparation of the 3<sup>rd</sup> ETSI NFV Plugtests.

---

## Introduction

The present document describes the Interoperability Test Plan that was followed during the 3<sup>rd</sup> ETSI NFV Plugtests held from 28<sup>th</sup> of May to 8<sup>th</sup> of June 2018 in Sophia Antipolis, France.

The Test Plan was developed following the interoperability testing methodology and guidelines defined by ETSI NFV in [TST002] and [TST007] and building on the learnings of the 1<sup>st</sup> and 2<sup>nd</sup> NFV Plugtests.

---

# 1 Scope

The goal of this document is to support the interoperability test sessions run during the 3<sup>rd</sup> 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), possibly providing EM and /or specific VNF Manager (VNFM) functionality
- Management and Orchestration (MANO) solutions, providing pre-integrated NFV Orchestrator (NFVO) and generic VNFM functionality
- NFV Platforms providing pre-integrated NFV Infrastructure (NFVI) and Virtual Infrastructure Manager (VIM) functionality

This document includes several System Under Test Configurations to illustrate how Functions Under Test are combined to provide different end-to-end functionality

The NS 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 NS derived from the examples 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"
- [TST007] ETSI GS NFV-TST 007: "Network Function Virtualization (NFV); Testing; Guidelines on Interoperability Testing for MANO"

---

## 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>\_<SUBGROUP>\_<OPERATION>\_<NN>

Where <ROOT> is NFV, and <NN> corresponds to a sequential number starting at 001.

The possible values for <GROUP>, <SUBGROUP>, <OPERATION> are summarized in the tables below:

<GROUP>	<SUBGROUP>	<OPERATION>	Description
	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)
		SCALE_IN	Scale In by removing VNF instance(s)
		SCALE_OUT_VNF	Scale Out by adding VNFC instance(s)
		SCALE_IN_VNF	Scale In by removing VNFC instance(s)
		SCALE_TO_LEVEL	Scale To Level by changing number of VNF instance(s)
		SCALE_TO_LEVEL_VNF	Scale To Level by changing number of VNFC instance(s)
		UPDATE VNF	Start and Stop VNF instances
		TERMINATE	Terminate Network Service
	FM_VR	ALARM	Virtualised resource fault alarm event propagation to MANO
		CLEAR	Virtualised resource fault clearance event propagation to MANO
	FM_VNF	ALARM	VNF fault alarm event detection by MANO
		CLEAR	VNF fault clearance event detection by MANO
	PM_VR	CREATE_MONITOR	Monitoring of virtualised resource performance metrics
		CREATE_THRESHOLD	Monitoring of virtualised resource performance metrics using thresholds based notifications
		DELETE_MONITOR	Deletion of monitoring of virtualised resource performance metrics
		DELETE_THRESHOLD	Deletion of monitoring of virtualised resource performance metrics using thresholds based notifications
	PM_VNF_KPI	CREATE_MONITOR	Monitoring of VNF indicators
		CREATE_THRESHOLD	Monitoring of VNF indicators using thresholds based notifications
		DELETE_MONITOR	Deletion of monitoring of VNF indicators
		DELETE_THRESHOLD	Deletion of monitoring of VNF indicators using thresholds based notifications

	DELETE	NSD	Deletion of Network Service Descriptor
		VNF_PKG	Deletion of VNF Package

EPA	NS_LCM	INSTANTIATE	Instantiate Network Service with EPA requirements
		SCALE_OUT	Scale Out by adding VNF instance(s) with EPA requirements
		SCALE_IN	Scale In by removing VNF instance(s) with EPA requirements
		SCALE_OUT_VNF	Scale Out by adding VNFC instance(s) with EPA requirements
		SCALE_IN_VNF	Scale In by removing VNFC instance(s) with EPA requirements

SFC	NS_LCM	INSTANTIATE	Instantiate Network Service with SFC based on NSH
		TERMINATE	Terminate Network Service

MULTISITE	NS_LCM	INSTANTIATE	Instantiate Multi Site Network Service
		SCALE_OUT	Multi Site Scale Out by adding VNF instance(s)
		SCALE_IN	Multi Site Scale In by removing VNF instance(s)
		SCALE_OUT_VNF	Multi Site Scale Out by adding VNFC instance(s)
		SCALE_IN_VNF	Multi Site Scale In by removing VNFC instance(s)
		TERMINATE	Terminate Multi Site Network Service

S_VNFM-D	NS_LCM	INSTANTIATE	Instantiate Network Service through specific VNFM in direct mode
		SCALE_OUT	Scale Out by adding VNF instance(s) through specific VNFM in direct mode
		SCALE_IN	Scale In by removing VNF instance(s) through specific VNFM in direct mode
		SCALE_OUT_VNF	Scale Out by adding VNFC instance(s) through specific VNFM in direct mode
		SCALE_IN_VNF	Scale In by removing VNFC instance(s) through specific VNFM in direct mode
	FM_VNF_VR	ALARM	Virtualised resource fault alarm event propagation to MANO from VNFM
		CLEAR	Virtualised resource fault clearance event propagation to MANO from VFM
	FM_VNF	ALARM	VNF fault alarm event detection by MANO from VNFM
		CLEAR	VNF fault clearance event detection by MANO from VNFM
	PM_VNF_VR	CREATE_MONITOR	Monitoring of virtualised resource performance metrics through VNFM
		CREATE_THRESHOLD	Monitoring of virtualised resource performance metrics using thresholds based notifications through VNFM
		DELETE_MONITOR	Deletion of monitoring of virtualised resource performance metrics through VNFM
		DELETE_THRESHOLD	Deletion of monitoring of virtualised resource performance metrics using thresholds based notifications through VNFM
	PM_VNF_KPI	CREATE_MONITOR	Monitoring of VNF indicators through VNFM
		CREATE_THRESHOLD	Monitoring of VNF indicators using thresholds based notifications through VNFM
		DELETE_MONITOR	Deletion of monitoring of VNF indicators through VNFM
		DELETE_THRESHOLD	Deletion of monitoring of VNF indicators using thresholds based notifications through VNFM

## 4.2 Test Summary

### 4.2.1 Single Vendor NS

#### 4.2.1.0 SUT Configuration

The System Under Test Configuration applicable to this group is:

- SUT\_SINGLE-VENDOR\_NS

See Clause 5 for further details

#### 4.2.1.1 ONBOARD

Test Id	Test Purpose
TD_NFV_ONBOARD_VNF_PKG_001	To on-board a VNF Package
TD_NFV_ONBOARD_NSD_001	To onboard a NSD

#### 4.2.1.2 INSTANTIATE

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

#### 4.2.1.3 SCALE NS

##### 4.2.1.3.1 SCALE NS MANUALLY

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_IN_001	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator

##### 4.2.1.3.2 SCALE NS FROM VNF INDICATOR

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_OUT_002a	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by a VNF Indicator notification
TD_NFV_NS_LCM_SCALE_IN_002a	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by a VNF Indicator notification
TD_NFV_NS_LCM_SCALE_OUT_002b	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by querying a VNF Indicator
TD_NFV_NS_LCM_SCALE_IN_002b	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by querying a VNF Indicator

## 4.2.1.3.3 SCALE NS FROM VIM KPI

Test Id	Test Purpose
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

## 4.2.1.3.4 SCALE NS FROM VNF REQ

Test Id	Test Purpose
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

## 4.2.1.4 SCALE VNF

## 4.2.1.4.1 SCALE VNF MANUALLY

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

## 4.2.1.4.2 SCALE VNF FROM VNF INDICATOR

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_OUT_VNF_002a	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 VNF Indicator notification
TD_NFV_NS_LCM_SCALE_IN_VNF_002a	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered automatically in MANO by a VNF Indicator notification
TD_NFV_NS_LCM_SCALE_OUT_VNF_002b	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered automatically in MANO by querying a VNF Indicator
TD_NFV_NS_LCM_SCALE_IN_VNF_002b	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered automatically in MANO by querying a VNF Indicator

## 4.2.1.4.3 SCALE VNF FROM VIM KPI

Test Id	Test Purpose
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 removing VNFC instances (VMs))

	when triggered automatically in MANO by a VIM KPI
--	---

#### 4.2.1.4.4 SCALE VNF FROM VNF REQ

Test Id	Test Purpose
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

#### 4.2.1.5 SCALE NS TO LEVEL

##### 4.2.1.5.1 SCALE NS TO LEVEL MANUALLY

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_001	Verify that an NS can be successfully scaled to another existing instantiation level by changing the number of VNF instances when triggered by a MANO operator

##### 4.2.1.5.2 SCALE NS TO LEVEL FROM VNF INDICATOR

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_002	Verify that an NS can be successfully scaled to another existing instantiation level by changing the number of VNF instances when triggered automatically by a VNF indicator

##### 4.2.1.5.3 SCALE NS TO LEVEL FROM VIM KPI

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_003	Verify that an NS can be successfully scaled to another existing instantiation level by changing the number of VNF instances when triggered automatically by a VIM KPI

#### 4.2.1.6 SCALE VNF TO LEVEL

##### 4.2.1.6.1 SCALE VNF TO LEVEL MANUALLY

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_001	Verify that a VNF in a NS can be successfully scaled to another existing instantiation level by changing the number of VNFC instances when triggered by a MANO operator

##### 4.2.1.6.2 SCALE VNF TO LEVEL FROM VNF INDICATOR

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_002	Verify that a VNF in a NS can be successfully scaled to another existing instantiation level by changing the number of VNFC instances when triggered automatically triggered by a VNF indicator

## 4.2.1.6.3 SCALE VNF TO LEVEL FROM VIM KPI

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_003	Verify that a VNF in a NS can be successfully scaled to another existing instantiation level by changing the number of VNFC instances when triggered automatically by a VIM KPI

## 4.2.1.7 UPDATE VNF

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

## 4.2.1.8 FAULT MANAGEMENT

## 4.2.1.8.1 FAULT MANAGEMENT - VR

Test Id	Test Purpose
TD_NFV_FM_VR_ALARM_001	Verify that a fault alarm event propagates to the MANO when a virtualised resource that is required for the NS fails.
TD_NFV_FM_VR_CLEAR_001	Verify that a fault clearance event propagates to the MANO when a failed virtualised resource that is required for the NS is recovered

## 4.2.1.8.2 FAULT MANAGEMENT - VNF

Test Id	Test Purpose
TD_NFV_FM_VNF_ALARM_001	Verify that a VNF fault alarm event is detected by the MANO when a fault occurs on a VNF part of a NS
TD_NFV_FM_VNF_CLEAR_001	Verify that a VNF fault alarm clearance event is detected by the MANO when a fault is cleared on a VNF part of a NS by resolving the causing problem

## 4.2.1.9 PERFORMANCE MANAGEMENT

## 4.2.1.9.1 PERFORMANCE MANAGEMENT - VR

Test Id	Test Purpose
TD_NFV_PM_VR_CREATE_MONITOR_001	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored
TD_NFV_PM_VR_CREATE_THRESHOLD_001	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored using thresholds based notifications
TD_NFV_PM_VR_DELETE_MONITOR_001	To verify that monitoring of performance metrics of one or more virtualised resources that are allocated to a NS instance can be stopped
TD_NFV_PM_VR_DELETE_THRESHOLD_001	To verify that performance monitoring thresholds created for one or more virtualised resources that are allocated to a NS instance can be deleted

## 4.2.1.9.2 PERFORMANCE MANAGEMENT – VNF KPI

Test Id	Test Purpose
TD_NFV_PM_VNF_KPI_CREATE_MONITOR_001	To verify that a VNF indicator related to a NS instance can be monitored
TD_NFV_PM_VNF_KPI_DELETE_MONITOR_001	To verify that monitoring of a VNF indicator related to a NS instance can be stopped
TD_NFV_PM_VNF_KPI_CREATE_THRESHOLD_001	To verify that a VNF indicator related to a NS instance can be monitored using thresholds based notifications
TD_NFV_PM_VNF_KPI_DELETE_THRESHOLD_001	To verify that performance monitoring thresholds created for one or more VNF indicator related to a NS instance can be deleted

## 4.2.1.10 TERMINATE

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

## 4.2.1.11 DELETE

Test Id	Test Purpose
TD_NFV_DELETE_NSD_001	To delete a NSD
TD_NFV_DELETE_VNF_PKG_001	To delete a VNF Package

## 4.2.2 Multi Vendor NS

## 4.2.2.0 SUT Configuration

The System Under Test Configuration applicable to this group is:

- SUT\_MULTI-VENDOR\_NS

See Clause 5 for further details

## 4.2.2.1 ONBOARD

Test Id	Test Purpose
TD_NFV_ONBOARD_VNF_PKG_001	To on-board a VNF Package
TD_NFV_ONBOARD_NSD_001	To onboard a NSD

## 4.2.2.2 INSTANTIATE

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

### 4.2.2.3 SCALE NS

#### 4.2.2.3.1 SCALE NS MANUALLY

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_IN_001	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator

#### 4.2.2.3.2 SCALE NS FROM VNF INDICATOR

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_OUT_002a	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by a VNF Indicator notification
TD_NFV_NS_LCM_SCALE_IN_002a	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by a VNF Indicator notification
TD_NFV_NS_LCM_SCALE_OUT_002b	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by querying a VNF Indicator
TD_NFV_NS_LCM_SCALE_IN_002b	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by querying a VNF Indicator

#### 4.2.2.3.3 SCALE NS FROM VIM KPI

Test Id	Test Purpose
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

#### 4.2.2.3.4 SCALE NS FROM VNF REQ

Test Id	Test Purpose
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

### 4.2.2.4 SCALE VNF

#### 4.2.2.4.1 SCALE VNF MANUALLY

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
--	--

#### 4.2.2.4.2 SCALE VNF FROM VNF INDICATOR

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_OUT_VNF_002a	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 VNF Indicator notification
TD_NFV_NS_LCM_SCALE_IN_VNF_002a	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered automatically in MANO by a VNF Indicator notification
TD_NFV_NS_LCM_SCALE_OUT_VNF_002b	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered automatically in MANO by querying a VNF Indicator
TD_NFV_NS_LCM_SCALE_IN_VNF_002b	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered automatically in MANO by querying a VNF Indicator

#### 4.2.2.4.3 SCALE VNF FROM VIM KPI

Test Id	Test Purpose
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 removing VNFC instances (VMs)) when triggered automatically in MANO by a VIM KPI

#### 4.2.2.4.4 SCALE VNF FROM VNF REQ

Test Id	Test Purpose
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

#### 4.2.2.5 SCALE NS TO LEVEL

##### 4.2.2.5.1 SCALE NS TO LEVEL MANUALLY

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_001	Verify that an NS can be successfully scaled to another existing instantiation level by changing the number of VNF instances when triggered by a MANO operator

## 4.2.2.5.2 SCALE NS TO LEVEL FROM VNF INDICATOR

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_002	Verify that an NS can be successfully scaled to another existing instantiation level by changing the number of VNF instances when triggered automatically by a VNF indicator

## 4.2.2.5.3 SCALE NS TO LEVEL FROM VIM KPI

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_003	Verify that an NS can be successfully scaled to another existing instantiation level by changing the number of VNF instances when triggered automatically by a VIM KPI

## 4.2.2.6 SCALE VNF TO LEVEL

## 4.2.2.6.1 SCALE VNF TO LEVEL MANUALLY

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_001	Verify that a VNF in a NS can be successfully scaled to another existing instantiation level by changing the number of VNFC instances when triggered by a MANO operator

## 4.2.2.6.2 SCALE VNF TO LEVEL FROM VNF INDICATOR

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_002	Verify that a VNF in a NS can be successfully scaled to another existing instantiation level by changing the number of VNFC instances when triggered automatically triggered by a VNF indicator

## 4.2.2.6.3 SCALE VNF TO LEVEL FROM VIM KPI

Test Id	Test Purpose
TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_003	Verify that a VNF in a NS can be successfully scaled to another existing instantiation level by changing the number of VNFC instances when triggered automatically by a VIM KPI

## 4.2.2.7 UPDATE VNF

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

## 4.2.2.8 FAULT MANAGEMENT

## 4.2.2.8.1 FAULT MANAGEMENT - VR

Test Id	Test Purpose
---------	--------------

TD_NFV_FM_VR_ALARM_001	Verify that a fault alarm event propagates to the MANO when a virtualised resource that is required for the NS fails.
TD_NFV_FM_VR_CLEAR_001	Verify that a fault clearance event propagates to the MANO when a failed virtualised resource that is required for the NS is recovered

#### 4.2.2.8.2 FAULT MANAGEMENT - VNF

Test Id	Test Purpose
TD_NFV_FM_VNF_ALARM_001	Verify that a VNF fault alarm event is detected by the MANO when a fault occurs on a VNF part of a NS
TD_NFV_FM_VNF_CLEAR_001	Verify that a VNF fault alarm clearance event is detected by the MANO when a fault is cleared on a VNF part of a NS by resolving the causing problem

#### 4.2.2.9 PERFORMANCE MANAGEMENT

##### 4.2.2.9.1 PERFORMANCE MANAGEMENT - VR

Test Id	Test Purpose
TD_NFV_PM_VR_CREATE_MONITOR_001	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored
TD_NFV_PM_VR_CREATE_THRESHOLD_001	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored using thresholds based notifications
TD_NFV_PM_VR_DELETE_MONITOR_001	To verify that monitoring of performance metrics of one or more virtualised resources that are allocated to a NS instance can be stopped
TD_NFV_PM_VR_DELETE_THRESHOLD_001	To verify that performance monitoring thresholds created for one or more virtualised resources that are allocated to a NS instance can be deleted

##### 4.2.2.9.2 PERFORMANCE MANAGEMENT – VNF KPI

Test Id	Test Purpose
TD_NFV_PM_VNF_KPI_CREATE_MONITOR_001	To verify that a VNF indicator related to a NS instance can be monitored
TD_NFV_PM_VNF_KPI_DELETE_MONITOR_001	To verify that monitoring of a VNF indicator related to a NS instance can be stopped
TD_NFV_PM_VNF_KPI_CREATE_THRESHOLD_001	To verify that a VNF indicator related to a NS instance can be monitored using thresholds based notifications
TD_NFV_PM_VNF_KPI_DELETE_THRESHOLD_001	To verify that performance monitoring thresholds created for one or more VNF indicator related to a NS instance can be deleted

##### 4.2.2.10 TERMINATE

Test Id	Test Purpose
---------	--------------

TD_NFV_NS_LCM_TERMINATE_001	To verify that a NS can be successfully terminated
-----------------------------	--

#### 4.2.2.11 DELETE

Test Id	Test Purpose
TD_NFV_DELETE_NSD_001	To delete a NSD
TD_NFV_DELETE_VNF_PKG_001	To delete a VNF Package

### 4.2.3 Multi Vendor NS with EPA

#### 4.2.3.0 SUT Configuration

The System Under Test Configuration applicable to this group is:

- SUT\_MULTI-VENDOR\_NS

See Clause 5 for further details

#### 4.2.3.1 ONBOARD

Test Id	Test Purpose
TD_NFV_ONBOARD_VNF_PKG_001	To on-board a VNF Package
TD_NFV_ONBOARD_NSD_001	To onboard a NSD

#### 4.2.3.2 INSTANTIATE

Test Id	Test Purpose
TD_NFV_EPA_NS_LCM_INSTANTIATE_001	To verify that an NS can be successfully instantiated with EPA requirements

#### 4.2.3.3 SCALE NS MANUALLY

Test Id	Test Purpose
TD_NFV_EPA_NS_LCM_SCALE_OUT_001	To verify that a NS can be successfully scaled out with EPA requirements (by adding VNF instances) if triggered by a MANO operator
TD_NFV_EPA_NS_LCM_SCALE_IN_001	To verify that a NS can be successfully scaled in with EPA requirements (by removing VNF instances) if triggered by a MANO operator

#### 4.2.3.4 SCALE VNF MANUALLY

Test Id	Test Purpose
TD_NFV_EPA_NS_LCM_SCALE_OUT_VNF_001	To verify that a VNF in a NS can be successfully scaled out with EPA requirements (by adding VNFC instances (VMs)) when triggered by a MANO operator
TD_NFV_EPA_NS_LCM_SCALE_IN_VNF_001	To verify that a VNF in a NS can be successfully scaled in with EPA requirements (by removing VNFC instances (VMs)) when triggered by a MANO operator

#### 4.2.3.5 TERMINATE

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

#### 4.2.3.6 DELETE

Test Id	Test Purpose
TD_NFV_DELETE_NSD_001	To delete a NSD
TD_NFV_DELETE_VNF_PKG_001	To delete a VNF Package

### 4.2.4 Multi Vendor NS with SFC

#### 4.2.4.0 SUT Configuration

The System Under Test Configuration applicable to this group is:

- SUT\_MULTI-VENDOR\_NS

See Clause 5 for further details

#### 4.2.4.1 ONBOARD

Test Id	Test Purpose
TD_NFV_ONBOARD_VNF_PKG_001	To on-board a VNF Package
TD_NFV_ONBOARD_NSD_001	To onboard a NSD

#### 4.2.4.2 INSTANTIATE

Test Id	Test Purpose
TD_NFV_SFC_NS_LCM_INSTANTIATE_001	To verify that an NS with NSH based SFC can be successfully instantiated

**Table 1: Test Group**

#### 4.2.4.3 TERMINATE

Test Id	Test Purpose
TD_NFV_SFC_NS_LCM_TERMINATE_001	To verify that a NS can be successfully terminated

**Table 2: Test Group**

#### 4.2.4.4 DELETE

Test Id	Test Purpose
TD_NFV_DELETE_NSD_001	To delete a NSD
TD_NFV_DELETE_VNF_PKG_001	To delete a VNF Package

## 4.2.5 Multi Site

### 4.2.5.0 SUT Configuration

The System Under Test Configuration applicable to this group is:

- SUT\_MULTI-SITE

See Clause 5 for further details

#### 4.2.5.1 ONBOARD

Test Id	Test Purpose
TD_NFV_ONBOARD_VNF_PKG_001	To on-board a VNF Package
TD_NFV_ONBOARD_NSD_001	To on-board a Multi-Site NSD

#### 4.2.5.2 INSTANTIATE

Test Id	Test Purpose
TD_NFV_MULTISITE_NS_LCM_INSTANTIATE_001	To verify that an NS can be successfully instantiated across different sites

#### 4.2.5.3 SCALE NS MANUALLY

Test Id	Test Purpose
TD_NFV_MULTISITE_NS_LCM_SCALE_OUT_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_MULTISITE_NS_LCM_SCALE_IN_001	To verify that a multi-site NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator

#### 4.2.5.4 SCALE VNF MANUALLY

Test Id	Test Purpose
TD_NFV_MULTISITE_NS_LCM_SCALE_OUT_VNF_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_MULTISITE_NS_LCM_SCALE_IN_VNF_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

### 4.2.5.5 FAULT MANAGEMENT

#### 4.2.5.5.1 FAULT MANAGEMENT – VR

Test Id	Test Purpose
TD_NFV_FM_VR_ALARM_001	Verify that a fault alarm event propagates to the MANO when a virtualised resource that is required for the NS fails.
TD_NFV_FM_VR_CLEAR_001	Verify that a fault clearance event propagates to the MANO when a failed virtualised resource that is required for the NS is recovered

## 4.2.5.5.2 FAULT MANAGEMENT – VNF

Test Id	Test Purpose
TD_NFV_FM_VNF_ALARM_001	Verify that a VNF fault alarm event is detected by the MANO when a fault occurs on a VNF part of a NS
TD_NFV_FM_VNF_CLEAR_001	Verify that a VNF fault alarm clearance event is detected by the MANO when a fault is cleared on a VNF part of a NS by resolving the causing problem

## 4.2.5.6 PERFORMANCE MANAGEMENT

## 4.2.5.6.1 PERFORMANCE MANAGEMENT – VR

Test Id	Test Purpose
TD_NFV_PM_VR_CREATE_MONITOR_001	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored
TD_NFV_PM_VR_CREATE_THRESHOLD_001	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored using thresholds based notifications
TD_NFV_PM_VR_DELETE_MONITOR_001	To verify that monitoring of performance metrics of one or more virtualised resources that are allocated to a NS instance can be stopped
TD_NFV_PM_VR_DELETE_THRESHOLD_001	To verify that performance monitoring thresholds created for one or more virtualised resources that are allocated to a NS instance can be deleted

## 4.2.5.6.2 PERFORMANCE MANAGEMENT – VNF

Test Id	Test Purpose
TD_NFV_PM_VNF_KPI_CREATE_MONITOR_001	To verify that a VNF indicator related to a NS instance can be monitored
TD_NFV_PM_VNF_KPI_DELETE_MONITOR_001	To verify that monitoring of a VNF indicator related to a NS instance can be stopped
TD_NFV_PM_VNF_KPI_CREATE_THRESHOLD_001	To verify that a VNF indicator related to a NS instance can be monitored using thresholds based notifications
TD_NFV_PM_VNF_KPI_DELETE_THRESHOLD_001	To verify that performance monitoring thresholds created for one or more VNF indicator related to a NS instance can be deleted

## 4.2.5.7 TERMINATE

Test Id	Test Purpose
TD_NFV_MULTISITE_NS_LCM_TERMINATE_001	To verify that a Multi Site NS can be successfully terminated

## 4.2.5.8 DELETE

Test Id	Test Purpose
TD_NFV_DELETE_NSD_001	To delete a NSD
TD_NFV_DELETE_VNF_PKG_001	To delete a VNF Package

## 4.2.6 S-VNFM-D

### 4.2.6.0 SUT Configuration

The System Under Test Configuration applicable to this group is:

- SUT\_S-VNFM-D

See Clause 5 for further details

### 4.2.6.1 ONBOARD

Test Id	Test Purpose
TD_NFV_ONBOARD_VNF_PKG_001	To on-board a VNF Package
TD_NFV_ONBOARD_NSD_001	To on-board a NSD

**Table 3: Test Group VNFM-D\_ONBOARD**

### 4.2.6.2 INSTANTIATE

Test Id	Test Purpose
TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001	To verify that an NS can be successfully instantiated

### 4.2.6.3 SCALE NS MANUALLY

Test Id	Test Purpose
TD_NFV_S-VNFM-D_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_S-VNFM-D_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

### 4.2.6.4 SCALE VNF MANUALLY

Test Id	Test Purpose
TD_NFV_S-VNFM-D_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_S-VNFM-D_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

### 4.2.6.5 PERFORMANCE MANAGEMENT

#### 4.2.6.5.1 PERFORMANCE MANAGEMENT – VNF VR

Test Id	Test Purpose
TD_NFV_S-VNFM-D_PM_VNF_VR_CREATE_MONITOR_001	To verify that the performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be monitored through external VNFM
TD_NFV_S-VNFM-D_PM_VNF_VR_CREATE_THRESHOLD_001	To verify that the performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be monitored using thresholds based notifications through external VNFM

TD_NFV_S-VNFM-D_PM_VNF_VR_DELETE_MONITOR_001	To 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 through external VNFM
TD_NFV_S-VNFM-D_PM_VNF_VR_DELETE_THRESHOLD_001	To verify that a performance monitoring threshold created for a virtualised resource that is allocated to a VNF instance inside a NS instance can be deleted through external VNFM

#### 4.2.6.5.2 PERFORMANCE MANAGEMENT – VNF KPI

Test Id	Test Purpose
TD_NFV_S-VNFM-D_PM_VNF_KPI_CREATE_MONITOR_001	To verify that a VNF indicator related to a NS instance can be monitored through external VNFM
TD_NFV_S-VNFM-D_PM_VNF_KPI_DELETE_MONITOR_001	To verify that monitoring of a VNF indicator related to a NS instance can be stopped through external VNFM
TD_NFV_S-VNFM-D_PM_VNF_KPI_CREATE_THRESHOLD_001	To verify that a VNF indicator related to a NS instance can be monitored using thresholds based notifications through external VNFM
TD_NFV_S-VNFM-D_PM_VNF_KPI_DELETE_THRESHOLD_001	To verify that performance monitoring thresholds created for one or more VNF indicator related to a NS instance can be deleted through external VNFM

#### 4.2.6.6 FAULT MANAGEMENT

##### 4.2.6.6.1 FAULT MANAGEMENT – VNF VR

Test Id	Test Purpose
TD_NFV_S-VNFM-D_FM_VNF_VR_ALARM_001	Verify that a VNF fault alarm notification propagates via the specific VNFM to the MANO when a VNF fault is triggered by a failed virtualised resource
TD_NFV_S-VNFM-D_FM_VNF_VR_CLEAR_001	Verify that a VNF fault alarm clearance notification propagates via the specific VNFM to the MANO when a VNF fault is cleared by resolving the causing problem on the failed virtualised resource

##### 4.2.6.6.2 FAULT MANAGEMENT – VNF

Test Id	Test Purpose
TD_NFV_S-VNFM-D_FM_VNF_ALARM_001	Verify that a VNF fault alarm notification propagates via the VNFM to the MANO when a fault occurs on a VNF part of a NS
TD_NFV_S-VNFM-D_FM_VNF_CLEAR_001	Verify that a VNF fault alarm clearance notification propagates via the VNFM to the MANO when a fault is cleared on a VNF part of a NS by resolving the causing problem

#### 4.2.6.7 TERMINATE

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

#### 4.2.6.8 DELETE

Test Id	Test Purpose
TD_NFV_DELETE_NSD_001	To delete a NSD

TD_NFV_DELETE_VNF_PKG_001	To delete a VNF Package
---------------------------	-------------------------

## 4.2.7 S-VNFM-I

### 4.2.7.0 SUT Configuration

The System Under Test Configuration applicable to this group is:

- SUT\_S-VNFM-I

See Clause 5 for further details

### 4.2.7.1 ONBOARD

Test Id	Test Purpose
TD_NFV_ONBOARD_VNF_PKG_001	To on-board a VNF Package
TD_NFV_ONBOARD_NSD_001	To on-board a NSD

### 4.2.7.2 INSTANTIATE

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

### 4.2.7.3 SCALE NS MANUALLY

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_IN_001	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator

### 4.2.7.4 SCALE VNF MANUALLY

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

### 4.2.7.5 PERFORMANCE MANAGEMENT

#### 4.2.7.5.1 PERFORMANCE MANAGEMENT – VR

Test Id	Test Purpose
TD_NFV_PM_VR_CREATE_MONITOR_001	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored
TD_NFV_PM_VR_CREATE_THRESHOLD_001	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored using thresholds based notifications

TD_NFV_PM_VR_DELETE_MONITOR_001	To verify that monitoring of performance metrics of one or more virtualised resources that are allocated to a NS instance can be stopped
TD_NFV_PM_VR_DELETE_THRESHOLD_001	To verify that performance monitoring thresholds created for one or more virtualised resources that are allocated to a NS instance can be deleted

#### 4.2.7.5.2 PERFORMANCE MANAGEMENT – VNF KPI

Test Id	Test Purpose
TD_NFV_PM_VNF_KPI_CREATE_MONITOR_001	To verify that a VNF indicator related to a NS instance can be monitored through external VNFM
TD_NFV_PM_VNF_KPI_DELETE_MONITOR_001	To verify that monitoring of a VNF indicator related to a NS instance can be stopped through external VNFM
TD_NFV_PM_VNF_KPI_CREATE_THRESHOLD_001	To verify that a VNF indicator related to a NS instance can be monitored using thresholds based notifications through external VNFM
TD_NFV_PM_VNF_KPI_DELETE_THRESHOLD_001	To verify that performance monitoring thresholds created for one or more VNF indicator related to a NS instance can be deleted through external VNFM

#### 4.2.7.6 FAULT MANAGEMENT

##### 4.2.7.6.1 FAULT MANAGEMENT – VR

Test Id	Test Purpose
TD_NFV_FM_VR_ALARM_001	Verify that a fault alarm event propagates to the MANO when a virtualised resource that is required for the NS fails.
TD_NFV_FM_VR_CLEAR_001	Verify that a fault clearance event propagates to the MANO when a failed virtualised resource that is required for the NS is recovered

##### 4.2.7.6.2 FAULT MANAGEMENT – VNF

Test Id	Test Purpose
TD_NFV_FM_VNF_ALARM_001	Verify that a VNF fault alarm notification propagates via the VNFM to the MANO when a fault occurs on a VNF part of a NS
TD_NFV_FM_VNF_CLEAR_001	Verify that a VNF fault alarm clearance notification propagates via the VNFM to the MANO when a fault is cleared on a VNF part of a NS by resolving the causing problem

#### 4.2.7.7 TERMINATE

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

#### 4.2.7.8 DELETE

Test Id	Test Purpose
TD_NFV_DELETE_NSD_001	To delete a NSD
TD_NFV_DELETE_VNF_PKG_001	To delete a VNF Package

## 4.2.8 Auto LCM Validation

### 4.2.8.0 SUT Configuration

The System Under Test Configuration applicable to this group is:

- SUT\_AUTO-LCM-VALIDATION

See Clause 5 for further details

#### 4.2.8.1 ONBOARD

Test Id	Test Purpose
TD_NFV_ONBOARD_NSD_001	To on-board a NSD

#### 4.2.8.2 INSTANTIATE

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

#### 4.2.8.3 SCALE NS MANUALLY

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_IN_001	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator

#### 4.2.8.4 SCALE VNF MANUALLY

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

#### 4.2.8.5 UPDATE VNF

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

#### 4.2.8.6 FAULT MANAGEMENT – VR

Test Id	Test Purpose
TD_NFV_FM_VR_ALARM_001	Verify that a fault alarm event propagates to the MANO when a virtualised resource that is required for the NS fails.

TD_NFV_FM_VR_CLEAR_001	Verify that a fault clearance event propagates to the MANO when a failed virtualised resource that is required for the NS is recovered
------------------------	--

#### 4.2.8.7 TERMINATE

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

#### 4.2.8.8 DELETE

Test Id	Test Purpose
TD_NFV_DELETE_NSD_001	To delete a NSD

## 5 System Under Test Configurations

### 5.1 SUT\_SINGLE-VENDOR\_NS

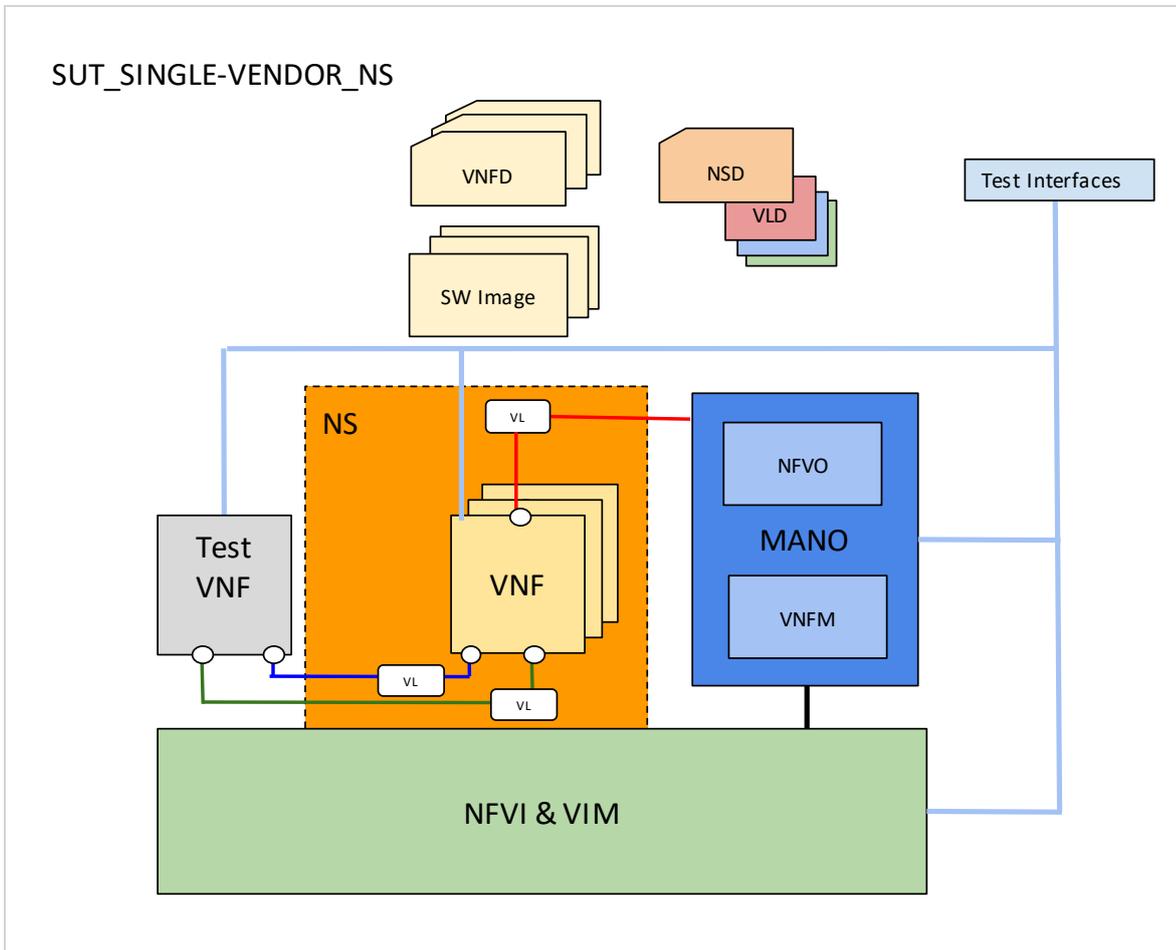
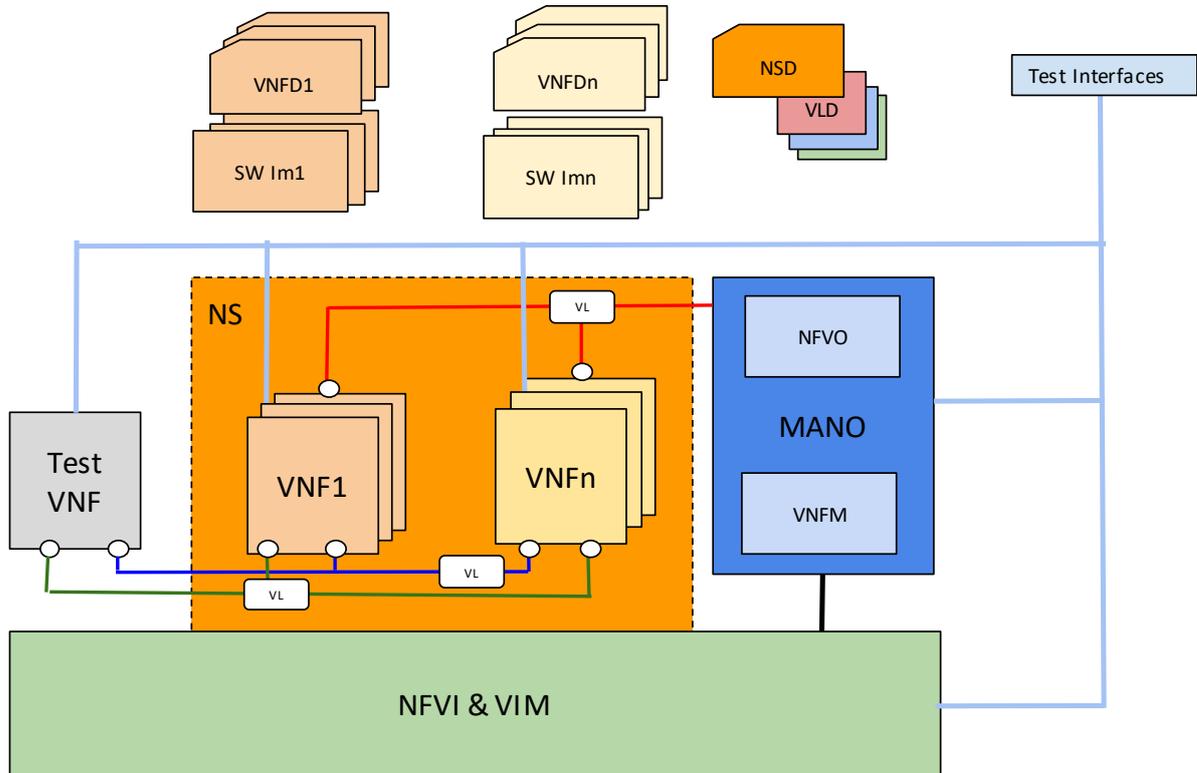


Figure 1: Single Vendor NS SUT Configuration

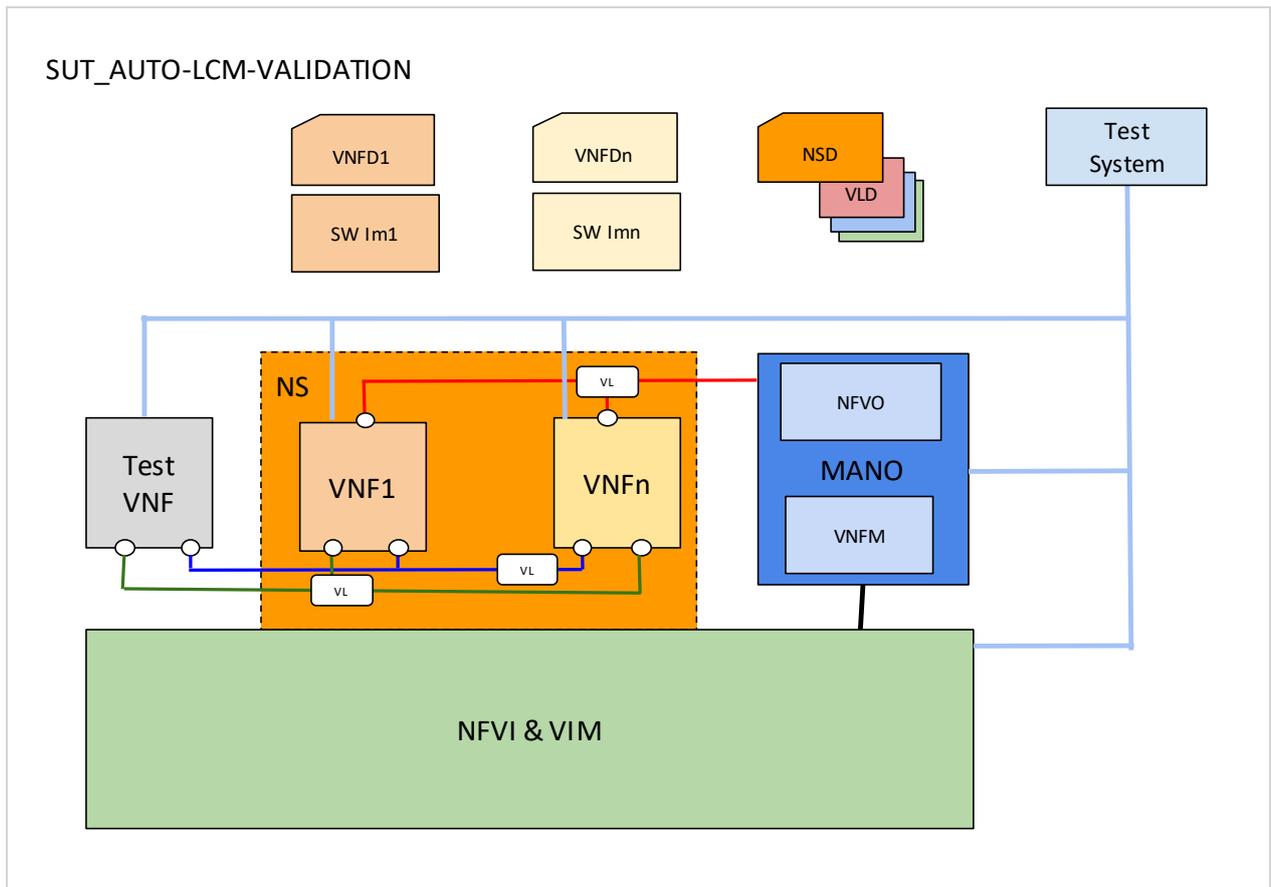
## 5.2 SUT\_MULTI-VENDOR\_NS

SUT\_MULTI-VENDOR\_NS



**Figure 2: Multi Vendor NS SUT Configuration**

## 5.3 SUT\_AUTO-LCM-VALIDATION



**Figure 3: Auto LCM Validation SUT Configuration**

## 5.4 SUT\_MULTI-SITE

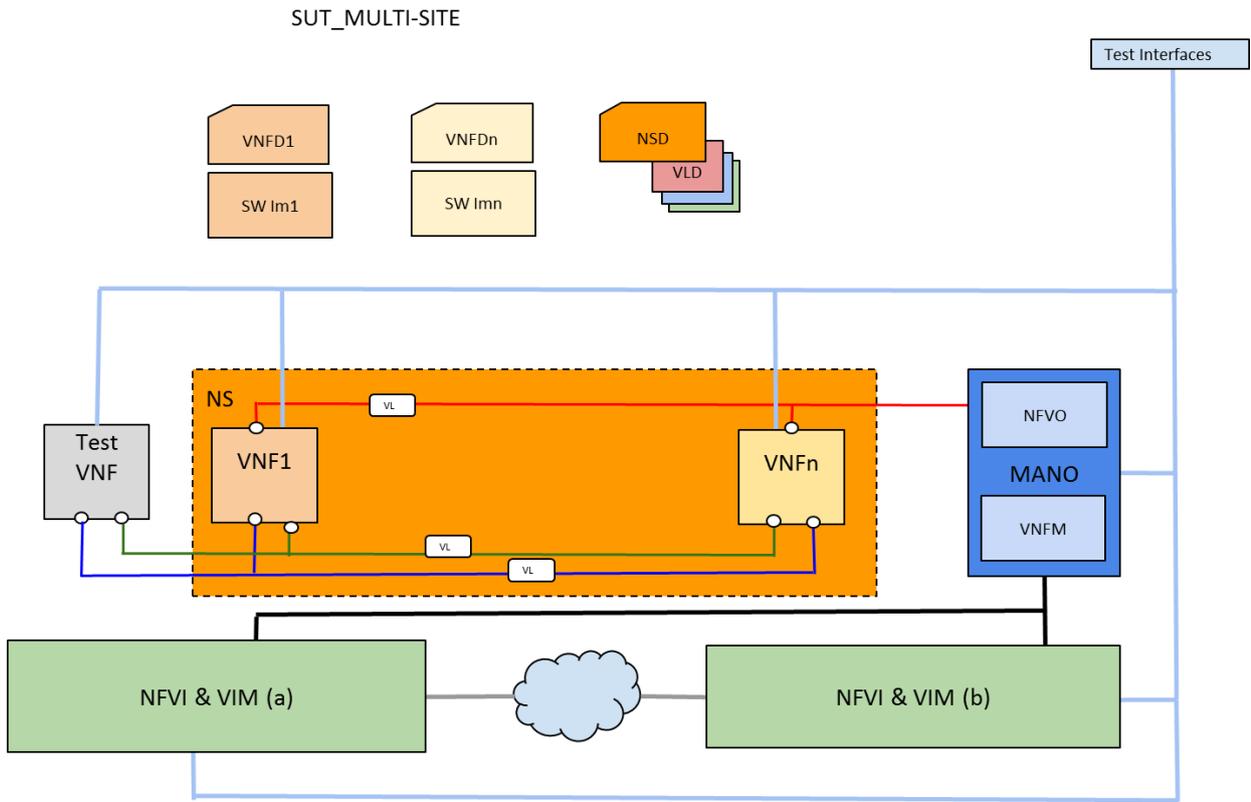


Figure 4: Multi Site SUT Configuration





## 6 Interoperability Test Descriptions

### 6.1 NS

#### 6.1.1 ONBOARD

##### 6.1.1.1 TD\_NFV\_ONBOARD\_VNF\_PKG\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_ONBOARD_VNF_PKG_001			
<b>Test Purpose</b>	To on-board a VNF Package			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-D SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clauses 7.7.2)			
<b>Applicability</b>				
<b>Pre-test conditions</b>	* VNF Package resides on a repository reachable by MANO * VNF Package is complete and consumable by MANO			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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, ...)	
<b>IOP Verdict</b>				

##### 6.1.1.2 TD\_NFV\_ONBOARD\_NSD\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_ONBOARD_NSD_001
<b>Test Purpose</b>	To onboard a NSD
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-D SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clauses 7.2.2)
<b>Applicability</b>	

<b>Pre-test conditions</b>	* NSD and referenced VLD and VNFFGDs exist and can be reached and consumed by MANO * Required VNF Packages have been on-boarded (TD_NFV_ONBOARD_VNF_PKG_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

## 6.1.2 INSTANTIATE

### 6.1.2.1 TD\_NFV\_NS\_LCM\_INSTANTIATE\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_INSTANTIATE_001			
<b>Test Purpose</b>	To verify that an NS can be successfully instantiated			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.3) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.3)			
<b>Applicability</b>				
<b>Pre-test conditions</b>	* 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			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

### 6.1.3 SCALE NS

#### 6.1.3.1 SCALE NS MANUALLY

##### 6.1.3.1.1 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_001			
<b>Test Purpose</b>	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered by a MANO operator			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	*[IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances *[IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 getting 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	
<b>IOP Verdict</b>				

##### 6.1.3.1.2 TD\_NFV\_NS\_LCM\_SCALE\_IN\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_001

<b>Test Purpose</b>	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	*[IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances *[IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_MULTIVENDOR_NS_LCM_INSTANTIATE_001) * NS has been scaled out by adding VNF instances			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

### 6.1.3.2 SCALE NS FROM VNF INDICATOR

#### 6.1.3.2.1 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_002a

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_002a
<b>Test Purpose</b>	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by a VNF Indicator notification
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_S-VNFM-I
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.3)
<b>Applicability</b>	* [IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM * [IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM * [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances * [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances * [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO
<b>Pre-test conditions</b>	* 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 to send the targeted VNF indicator notification to MANO until the configured threshold is crossed	
	2	IOP Check	Verify that the scale out (by adding VNF instance(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 instance(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	
<b>IOP Verdict</b>				

## 6.1.3.2.2 TD\_NFV\_NS\_LCM\_SCALE\_IN\_002a

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_002a			
<b>Test Purpose</b>	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by a VNF Indicator notification			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.3)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances</li> <li>* [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances</li> <li>* [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>* NS has been scaled out by adding VNF instances</li> <li>* MANO is configured to trigger SCALE IN (by removing VNF instances) when a given VNF Indicator value crosses a certain threshold</li> </ul>			
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Trigger the VNF to send the targeted VNF indicator notification to MANO until the configured threshold is crossed	
	2	IOP Check	Verify that the scale in (by removing VNF instance(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	
<b>IOP Verdict</b>				

## 6.1.3.2.3 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_002b

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_002b			
<b>Test Purpose</b>	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by querying a VNF Indicator			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances</li> <li>* [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances</li> <li>* [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>* MANO is configured to trigger SCALE OUT (by adding VNF instances) when a given VNF Indicator value crosses a certain threshold</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1a	Stimulus	In the VNF, trigger the target VNF indicator to cross the configured auto-scaling threshold value for scale out operation	
	1b	Stimulus	Trigger MANO to query the VNF for retrieving a new value of the VNF indicator	
	2	IOP Check	Verify that the scale out (by adding VNF instance(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 instance(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	
<b>IOP Verdict</b>				

## 6.1.3.2.4 TD\_NFV\_NS\_LCM\_SCALE\_IN\_002b

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_002b			
<b>Test Purpose</b>	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by querying a VNF Indicator			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances</li> <li>* [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances</li> <li>* [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>* NS has been scaled out by adding VNF instances</li> <li>* MANO is configured to trigger SCALE IN (by removing VNF instances) when a given VNF Indicator value crosses a certain threshold</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1a	Stimulus	In the VNF, trigger the target VNF indicator to cross the configured auto-scaling threshold value for scale in operation	
	1b	Stimulus	Trigger MANO to query the VNF for retrieving a new value of the VNF indicator	
	2	IOP Check	Verify that the scale in (by removing VNF instance(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	
<b>IOP Verdict</b>				

## 6.1.3.3 SCALE NS FROM VIM KPI

## 6.1.3.3.1 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_003

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_003
<b>Test Purpose</b>	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered automatically in MANO by a VIM KPI

<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM</li> <li>* [IFS_NFV_MANO_20] MANO supports automatic scaling out/in triggered by KPIs from VIM</li> <li>* [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances</li> <li>* [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances</li> <li>* [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>* MANO is configured to trigger SCALE OUT (by adding VNF instances) when a given VIM KPI value crosses a certain threshold</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 instance(s)) procedure has been started in MANO	
	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 instance(s) are running and reachable through the management network	
	5	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)	
	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	
<b>IOP Verdict</b>				

## 6.1.3.3.2 TD\_NFV\_NS\_LCM\_SCALE\_IN\_003

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_003
<b>Test Purpose</b>	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered automatically in MANO by a VIM KPI
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM</li> <li>* [IFS_NFV_MANO_20] MANO supports automatic scaling out/in triggered by KPIs from VIM</li> <li>* [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances</li> <li>* [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances</li> </ul>

		* [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM * [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM * [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM		
<b>Pre-test conditions</b>		* 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		
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 in (by removing VNF instance(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	
<b>IOP Verdict</b>				

### 6.1.3.4 SCALE NS FROM VNF REQUEST

#### 6.1.3.4.1 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_004

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_004			
<b>Test Purpose</b>	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered in MANO by a VNF/EM request			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_16] MANO supports scaling out/in request from VNF/EM * [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances * [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances * [IFS_NFV_VNF_8] VNF/EM can request scaling to MANO			
<b>Pre-test conditions</b>		* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)		
	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>

<b>Test Sequence</b>	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 instance(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 instance(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	
<b>IOP Verdict</b>				

## 6.1.3.4.2 TD\_NFV\_NS\_LCM\_SCALE\_IN\_004

<b>Interoperability Test Description</b>				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_004			
<b>Test Purpose</b>	To verify that a NS can successfully scale in (by removing VNF instances) if triggered in MANO by a VNF/EM request			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_16] MANO supports scaling out/in request from VNF/EM * [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances * [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances * [IFS_NFV_VNF_8] VNF/EM can request scaling to MANO			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * NS has been scaled out by adding VNF instances			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 removing VNF instance(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	
<b>IOP Verdict</b>				

## 6.1.4 SCALE VNF

### 6.1.4.1 SCALE VNF MANUALLY

#### 6.1.4.1.1 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_VNF\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_VNF_001			
<b>Test Purpose</b>	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered by a MANO operator			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	*[IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances *[IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

#### 6.1.4.1.2 TD\_NFV\_NS\_LCM\_SCALE\_IN\_VNF\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_VNF_001
<b>Test Purpose</b>	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered by a MANO operator
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION

<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	*[IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances *[IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances			
<b>Pre-test conditions</b>	* NS has been scaled out by adding VM			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

#### 6.1.4.2 SCALE VNF FROM VNF INDICATOR

##### 6.1.4.2.1 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_VNF\_002a

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_VNF_002a			
<b>Test Purpose</b>	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 VNF Indicator notification			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.3)			
<b>Applicability</b>	*[IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM *[IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM *[IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances *[IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances *[IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO			
<b>Pre-test conditions</b>	* 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			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger the VNF to send the targeted VNF indicator notification 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	
	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	
<b>IOP Verdict</b>				

## 6.1.4.2.2 TD\_NFV\_NS\_LCM\_SCALE\_IN\_VNF\_002a

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_VNF_002a			
<b>Test Purpose</b>	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered automatically in MANO by a VNF Indicator notification			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.3)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>* NS has been scaled out by adding VM(s)</li> <li>* MANO is configured to trigger SCALE IN (by removing VM(s)) when a given VNF Indicator value crosses a certain threshold</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger the VNF to send the targeted VNF indicator notification 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	
<b>IOP Verdict</b>				

## 6.1.4.2.3 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_VNF\_002b

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_VNF_002b			
<b>Test Purpose</b>	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered automatically in MANO by querying a VNF Indicator			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>* MANO is configured to trigger SCALE OUT (by adding VM(s)) when a given VNF Indicator value crosses a certain threshold</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1a	Stimulus	In the VNF, trigger the target VNF indicator to cross the configured auto-scaling threshold value for scale out operation	
	1b	Stimulus	Trigger MANO to query the VNF for retrieving a new value of the VNF indicator	
	2	IOP Check	Verify that the scale out (by adding VNFC instances (VMs)) procedure has been started in MANO	
	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	
<b>IOP Verdict</b>				

## 6.1.4.2.4 TD\_NFV\_NS\_LCM\_SCALE\_IN\_VNF\_002b

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_VNF_002b
<b>Test Purpose</b>	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered automatically in MANO by querying a VNF Indicator
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.4)

<b>Applicability</b>		<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO</li> </ul>		
<b>Pre-test conditions</b>		<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>* NS has been scaled out by adding VM(s)</li> <li>* MANO is configured to trigger SCALE IN (by removing VM(s)) when a given VNF Indicator value crosses a certain threshold</li> </ul>		
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1a	Stimulus	In the VNF, trigger the target VNF indicator to cross the configured auto-scaling threshold value for scale in operation	
	1b	Stimulus	Trigger MANO to query the VNF for retrieving a new value of the VNF indicator	
	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	
<b>IOP Verdict</b>				

### 6.1.4.3 SCALE VNF FROM VIM KPI

#### 6.1.4.3.1 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_VNF\_003

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_VNF_003
<b>Test Purpose</b>	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
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM</li> <li>* [IFS_NFV_MANO_20] MANO supports automatic scaling out/in triggered by KPIs from VIM</li> <li>* [IFS_NFV_MANO_15] MANO supports scaling by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM</li> </ul>

		* [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM		
<b>Pre-test conditions</b>		* 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		
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
	6	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test	
<b>IOP Verdict</b>				

## 6.1.4.3.2 TD\_NFV\_NS\_LCM\_SCALE\_IN\_VNF\_003

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_VNF_003			
<b>Test Purpose</b>	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered automatically in MANO by a VIM KPI			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM</li> <li>* [IFS_NFV_MANO_20] MANO supports automatic scaling out/in triggered by KPIs from VIM</li> <li>* [IFS_NFV_MANO_15] MANO supports scaling by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM</li> </ul>			
<b>Pre-test conditions</b>	* 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			
	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>

<b>Test Sequence</b>	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	
<b>IOP Verdict</b>				

#### 6.1.4.4 SCALE VNF FROM VNF REQUEST

##### 6.1.4.4.1 TD\_NFV\_NS\_LCM\_SCALE\_OUT\_VNF\_004

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_OUT_VNF_004			
<b>Test Purpose</b>	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			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_16] MANO supports scaling out/in request from VNF/EM</li> <li>* [IFS_NFV_MANO_15] MANO supports scaling by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_8] VNF/EM can request scaling to MANO</li> </ul>			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	

<b>IOP Verdict</b>		
--------------------	--	--

#### 6.1.4.4.2 TD\_NFV\_NS\_LCM\_SCALE\_IN\_VNF\_004

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_IN_VNF_004			
<b>Test Purpose</b>	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			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_16] MANO supports scaling out/in request from VNF/EM</li> <li>* [IFS_NFV_MANO_15] MANO supports scaling by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_8] VNF/EM can request scaling to MANO</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>* NS has been scaled out by adding VM(s)</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger the VNF/EM to send a scale in (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	
<b>IOP Verdict</b>				

### 6.1.5 SCALE NS TO LEVEL

#### 6.1.5.1 SCALE NS TO LEVEL MANUALLY

##### 6.1.5.1.1 TD\_NFV\_NS\_LCM\_SCALE\_TO\_LEVEL\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_TO_LEVEL_001
<b>Test Purpose</b>	Verify that an NS can be successfully scaled to another existing instantiation level by changing the number of VNF instances when triggered by a MANO operator
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS

	SUT_MULTI-VENDOR_NS		
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)		
<b>Applicability</b>	*[IFS_NFV_MANO_33] MANO supports NS scaling among levels by changing the number of VNF instances *[IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances		
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>• The NS initial deployment size should support scaling to or from a specified level</li> <li>• Current status of NS supports scale to level by changing the number of VNF instances</li> </ul>		
<b>Test Sequence</b>	<b>Step</b>		<b>Result</b>
	1	Stimulus	Trigger NS scale to another existing instantiation level in MANO with an operator action
	2	IOP Check	Verify that the number of VNF instance(s) has changed for the NS by querying the MANO
	3	IOP Check	Verify that the resources allocated by the VIM have changed according to the descriptors
	4	IOP Check	Verify that all VNF instance(s) are running and reachable via the management network
	5	IOP Check	Verify that the additional VNF instance(s) have been configured according to the descriptors
	6	IOP Check	Verify that all VNF instance(s) are connected to the VL(s) according to the descriptors
	7	IOP Check	Verify that the MANO indicates the scaling operation result as successful
	8	IOP Check	Verify that NS has been scaled by running the end-to-end functional test in relevance to the VNF scale and capacity
<b>IOP Verdict</b>			

## 6.1.5.2 SCALE NS TO LEVEL FROM VNF INDICATOR

### 6.1.5.2.1 TD\_NFV\_NS\_LCM\_SCALE\_TO\_LEVEL\_002

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_TO_LEVEL_002a
<b>Test Purpose</b>	Verify that an NS can be successfully scaled to another existing instantiation level by changing the number of VNF instances when triggered automatically by a VNF indicator
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.3)
<b>Applicability</b>	* [IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM * [IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM * [IFS_NFV_MANO_33] MANO supports NS scaling among levels by changing the number of VNF instances * [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances

	* [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO		
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>• NS is instantiated (TD_NFV_SCALE-LEVEL_NS_LCM_INSTANTIATE_001)</li> <li>• The NS initial deployment size should support scaling to or from a specified level</li> <li>• MANO is configured to trigger “scale to level by changing the number of VNF instances” when a given VNF indicator value crosses a certain threshold</li> <li>• Current status of NS supports scale to level by changing the number of VNF instances</li> </ul>		
<b>Test Sequence</b>	<b>Step</b>		<b>Result</b>
	1	Stimulus	Trigger the EM/VNF to send the targeted VNF indicator to the MANO until the configured threshold is crossed
	2	IOP Check	Verify that the “scale to level by adding VNF instance(s)” procedure has been started in MANO
	3	IOP Check	Verify that the number of VNF instance(s) has changed for the NS by querying the MANO
	4	IOP Check	Verify that the resources allocated by the VIM have changed according to the descriptors
	5	IOP Check	Verify that all VNF instance(s) are running and reachable via the management network
	6	IOP Check	Verify that the additional VNF instance(s) have been configured according to the descriptors
	7	IOP Check	Verify that all VNF instance(s) are connected to the VL(s) according to the descriptors
	8	IOP Check	Verify that the MANO indicates the scaling operation result as successful
	9	IOP Check	Verify that NS has been scaled by running the end-to-end functional test in relevance to the VNF scale and capacity
<b>IOP Verdict</b>			

### 6.1.5.3 SCALE NS TO LEVEL FROM VIM KPI

#### 6.1.5.3.1 TD\_NFV\_NS\_LCM\_SCALE\_TO\_LEVEL\_003

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_TO_LEVEL_003
<b>Test Purpose</b>	Verify that an NS can be successfully scaled to another existing instantiation level by changing the number of VNF instances when triggered automatically by a VIM KPI
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM</li> <li>* [IFS_NFV_MANO_20] MANO supports automatic scaling out/in triggered by KPIs from VIM</li> <li>* [IFS_NFV_MANO_33] MANO supports NS scaling among levels by changing the number of VNF instances</li> <li>* [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances</li> <li>* [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM</li> </ul>

* [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM * [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM				
<b>Pre-test conditions</b>		<ul style="list-style-type: none"> <li>• NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>• The NS initial deployment size should support scaling to or from a specified level</li> <li>• MANO is configured to trigger “scale to level by changing the number of VNF instances” when a given VIM KPI value crosses a certain threshold</li> <li>• Current status of NS supports scale to level by changing the number of VNF instances</li> </ul>		
<b>Test Sequence</b>	<b>Step</b>			<b>Result</b>
	1	Stimulus	Trigger the VIM to send the targeted KPI to the MANO until the configured threshold is crossed	
	2	IOP Check	Verify that the “scale to level by adding VNF instance(s)” procedure has been started in MANO	
	3	IOP Check	Verify that the number of VNF instance(s) has changed for the NS by querying the MANO	
	4	IOP Check	Verify that the resources allocated by the VIM have changed according to the descriptors	
	5	IOP Check	Verify that all VNF instance(s) are running and reachable via the management network	
	6	IOP Check	Verify that the additional VNF instance(s) have been configured according to the descriptors	
	7	IOP Check	Verify that all VNF instance(s) are connected to the VL(s) according to the descriptors	
	8	IOP Check	Verify that the MANO indicates the scaling operation result as successful	
	9	IOP Check	Verify that NS has been scaled by running the end-to-end functional test in relevance to the VNF scale and capacity	
<b>IOP Verdict</b>				

## 6.1.6 SCALE VNF TO LEVEL

### 6.1.6.1 SCALE VNF TO LEVEL MANUALLY

#### 6.1.6.1.1 TD\_NFV\_NS\_LCM\_SCALE\_TO\_LEVEL\_VNF\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_001
<b>Test Purpose</b>	Verify that a VNF in a NS can be successfully scaled to another existing instantiation level by changing the number of VNFC instances when triggered by a MANO operator
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)
<b>Applicability</b>	*[IFS_NFV_MANO_34] MANO supports VNF scaling among levels by changing the number of VNFC instances

		*[IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances		
<b>Pre-test conditions</b>		<ul style="list-style-type: none"> <li>NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>The NS initial deployment size should support scaling to a specified level</li> <li>NS/VNF supports scale to level by adding/removing VNFC instances</li> </ul>		
<b>Test Sequence</b>	<b>Step</b>			<b>Result</b>
	1	Stimulus	Trigger NS scale by scaling to another existing instantiation level a VNF in the NS in MANO with an operator action	
	2	IOP Check	Verify that the number of VNFC instance(s) has changed for the VNF by querying the MANO	
	3	IOP Check	Verify that the resources allocated by the VIM have changed according to the descriptors	
	4	IOP Check	Verify that all VNFC instance(s) are running and reachable via the management network	
	5	IOP Check	Verify that the VNF configuration has been updated to include the additional VNFC instances according to the descriptors	
	6	IOP Check	Verify that all VNFC instance(s) are connected to the VL(s) according to the descriptors	
	7	IOP Check	Verify that the MANO indicates the scaling operation result as successful	
8	IOP Check	Verify that NS has been scaled by running the end-to-end functional test in relevance to the VNF scale and capacity		
<b>IOP Verdict</b>				

## 6.1.6.2 SCALE VNF TO LEVEL FROM VNF INDICATOR

### 6.1.6.2.1 TD\_NFV\_NS\_LCM\_SCALE\_TO\_LEVEL\_VNF\_002

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_002
<b>Test Purpose</b>	Verify that a VNF in a NS can be successfully scaled to another existing instantiation level by changing the number of VNFC instances when triggered automatically triggered by a VNF indicator
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA008 V2.3.1 (clause 6.3.3)
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_17] MANO supports receiving VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_18] MANO supports automatic scaling triggered by VNF indicators from VNF/EM</li> <li>* [IFS_NFV_MANO_34] MANO supports VNF scaling among levels by changing the number of VNFC instances</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO</li> </ul>
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)</li> <li>The VNF initial deployment size should support scaling to a specified level</li> </ul>

		<ul style="list-style-type: none"> <li>MANO is configured to trigger “scale to level by adding VNFC instances” when a given VNF indicator value crosses a certain threshold</li> <li>NS/VNF supports scale to level by adding VNFC instances</li> </ul>		
Test Sequence	Step			Result
	1	Stimulus	Trigger the EM/VNF to send the targeted VNF indicator to the MANO until the configured threshold is crossed	
	2	IOP Check	Verify that the “scale to level by adding VNFC instance(s)” procedure has been started in MANO	
	3	IOP Check	Verify that the additional VNFC instance(s) have been deployed by querying the MANO	
	4	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	5	IOP Check	Verify that the additional VNFC instance(s) are running and are reachable through their management network	
	6	IOP Check	Verify that the VNF configuration has been updated to include the additional VNFC instances according to the descriptors	
	7	IOP Check	Verify that the additional VNFC instances(s) are connected to the VL(s) according to the descriptors	
	8	IOP Check	Verify that the MANO indicates the scaling operation result as successful	
	9	IOP Check	Verify that NS has been scaled to level by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

### 6.1.6.3 SCALE VNF TO LEVEL FROM VIM KPI

#### 6.1.6.3.1 TD\_NFV\_NS\_LCM\_SCALE\_TO\_LEVEL\_VNF\_003

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_SCALE_TO_LEVEL_VNF_003
<b>Test Purpose</b>	Verify that a VNF in a NS can be successfully scaled to another existing instantiation level by changing the number of VNFC instances when triggered automatically by a VIM KPI
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2, 7.7) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM</li> <li>* [IFS_NFV_MANO_20] MANO supports automatic scaling out/in triggered by KPIs from VIM</li> <li>* [IFS_NFV_MANO_34] MANO supports VNF scaling among levels by changing the number of VNFC instances</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM</li> </ul>

<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>• NS is instantiated (TD_NFV_SCALE-LEVEL_NS_LCM_INSTANTIATE_001)</li> <li>• The VNF initial deployment size should support scaling to a specified level</li> <li>• MANO is configured to trigger “scale to level by adding VNFC instances” when a given VIM KPI value crosses a certain threshold</li> <li>• NS/VNF supports scale to level by adding VNFC instances</li> </ul>			
Test Sequence	Step			Result
	1	Stimulus	Trigger the VIM to send the targeted KPI to the MANO until the configured threshold is crossed	
	2	IOP Check	Verify that the “scale to level by adding VNFC instance(s)” procedure has been started in MANO	
	3	IOP Check	Verify that the additional VNFC instance(s) have been deployed by querying the MANO	
	4	IOP Check	Verify that the additional resources have been allocated by the VIM according to the descriptors	
	5	IOP Check	Verify that the additional VNFC instance(s) are running and are reachable through their management network	
	6	IOP Check	Verify that the VNF configuration has been updated to include the additional VNFC instances according to the descriptors	
	7	IOP Check	Verify that the additional VNFC instances(s) are connected to the VL(s) 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 out by running the end-to-end functional test in relevance to the VNF scale and capacity	
IOP Verdict				

## 6.1.7 UPDATE VNF

### 6.1.7.1 TD\_NFV\_NS\_LCM\_UPDATE\_STOP\_VNF\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_UPDATE_STOP_VNF_001
<b>Test Purpose</b>	To verify that a VNF running in a NS can be successfully stopped by MANO
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_AUTO-LCM-VALIDATION
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.5) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.11)
<b>Applicability</b>	* [IFS_NFV_MANO_32] MANO can request to start/stop VNFs/VNFCs to the VIM * [IFS_NFV_VIM_NFVI_10] NFVI/VIM supports start/stop of VMs/VNFCs

<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * VNF instance(s) in the NS are running			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger the VNF(s) stop operation in MANO	
	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 shutdown on VIM (i.e query or display the state from VIM)	
<b>IOP Verdict</b>				

### 6.1.7.2 TD\_NFV\_NS\_LCM\_UPDATE\_START\_VNF\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_NS_LCM_UPDATE_START_VNF_001			
<b>Test Purpose</b>	To verify that a stopped VNF in a NS can be successfully re-started by MANO			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_AUTO-LCM-VALIDATION			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.5) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.11)			
<b>Applicability</b>	* [IFS_NFV_MANO_32] MANO can request to start/stop VNFs/VNFCs to the VIM * [IFS_NFV_VIM_NFVI_10] NFVI/VIM supports start/stop of VMs/VNFCs			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_MULTIVENDOR_NS_LCM_INSTANTIATE_001) * One VNF inside the NS has been stopped (TD_NFV_NS_LCM_UPDATE_STOP_VNF_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

## 6.1.8 FAULT MANAGEMENT

### 6.1.8.1 FAULT MANAGEMENT – VR

#### 6.1.8.1.1 TD\_NFV\_FM\_VR\_ALARM\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_FM_VR_ALARM_001
<b>Test Purpose</b>	Verify that a fault alarm event propagates to the MANO when a virtualised resource that is required for the NS fails.

<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clauses 5.3.10, 7.6.2, 8.6.2) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.5, 7.6.3, 8.5.2)			
<b>Applicability</b>	* [IFS_NFV_MANO_23] MANO supports receiving VM/VNFC faults/alarms from VIM * [IFS_NFV_VIM_NFVI_6] NFVI/VIM exposes VM/VNFC faults/alarms to MANO/VNFM			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * MANO is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) virtualised resources fault alarms from the VIM			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger a fault on a virtualised resource that is required for the NS instance 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 the virtualised resource alarm is collected or received by the MANO and a correspondent NS fault alarm is created, e.g. by querying the NS fault alarms database or checking the GUI where applicable	
<b>IOP Verdict</b>				

## 6.1.8.1.2 TD\_NFV\_FM\_VR\_CLEAR\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_FM_VR_CLEAR_001			
<b>Test Purpose</b>	Verify that a fault clearance event propagates to the MANO when a failed virtualised resource that is required for the NS is recovered			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clauses 5.3.10, 7.6.2, 8.6.3) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.5, 7.6.3, 8.5.3)			
<b>Applicability</b>	* [IFS_NFV_MANO_23] MANO supports receiving VM/VNFC faults/alarms from VIM * [IFS_NFV_VIM_NFVI_6] NFVI/VIM exposes VM/VNFC faults/alarms to MANO/VNFM			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * MANO is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) virtualised resources fault alarms from the VIM * NS fault alarm is created on the MANO by failing a virtualised resource that is required for the NS (TD_NFV_FM_VR_ALARM)			
	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>

<b>Test Sequence</b>	1	Stimulus	Resolve the failure of the virtualised resource that triggered the alarm on the NS instance (e.g. reconnect the NIC allocated to the correspondent NS network resource)	
	2	IOP Check	Verify that the correspondent 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 is cleared on the MANO, e.g. by querying the NS fault alarms database or checking the GUI where applicable	
<b>IOP Verdict</b>				

## 6.1.8.2 FAULT MANAGEMENT – VNF

### 6.1.8.2.1 TD\_NFV\_FM\_VNF\_ALARM\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_FM_VNF_ALARM_001			
<b>Test Purpose</b>	Verify that a VNF fault alarm event is detected by the MANO when a fault occurs on a VNF part of a NS			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.7, 6.4.6, 7.5.3, 8.4.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.5, 7.6.3, 8.5.2)			
<b>Applicability</b>	* MANO supports receiving VNF faults/alarms * VNF/EM exposes VNF fault alarms to MANO			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * MANO is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) VNF fault alarms from the VNF/EM			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger a fault on a VNF that is part of the relevant NS instance (e.g. disabling the management connectivity between VNF and MANO as an emulation of a VNF that is not alive any longer)	
	2	IOP Check	Verify that the VNF fault is detected by MANO, e.g. by querying the list of VNF fault alarms	
	3	IOP Check	Verify that the VNF fault alarm is processed by MANO and a correspondent NS fault alarm is created, e.g. by querying the NS fault alarms database or checking the GUI where applicable	
<b>IOP Verdict</b>				

### 6.1.8.2.2 TD\_NFV\_FM\_VNF\_CLEAR\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_FM_VNF_CLEAR_001
<b>Test Purpose</b>	Verify that a VNF fault alarm clearance event is detected by the MANO when a fault is cleared on a VNF part of a NS by resolving the causing problem
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS

	SUT_MULTI-SITE SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.7, 6.4.6, 7.5.3, 8.4.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.5, 7.6.3, 8.5.3)			
<b>Applicability</b>	* MANO supports receiving VNF faults/alarms * VNF/EM exposes VNF fault alarms to MANO			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * MANO is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) VNF fault alarms from the VNF/EM * NS fault alarm is created on the MANO by failing a VNF that is required for the NS (TD_NFV_FM_VNF_ALARM)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Resolve the failure of the VNF that triggered the alarm on the NS instance (e.g. re-enable the management connectivity between MANO and the VNF)	
	2	IOP Check	Verify that the fault clear event is detected by MANO, e.g. by querying the list of VNF fault alarms	
	3	IOP Check	Verify that the VNF fault clear is processed by MANO and the correspondent NS fault alarm is cleared, e.g. by querying the NS fault alarms database or checking the GUI where applicable	
<b>IOP Verdict</b>				

## 6.1.9 PERFORMANCE MANAGEMENT

### 6.1.9.1 PERFORMANCE MANAGEMENT – VR

#### 6.1.9.1.1 TD\_NFV\_PM\_VR\_CREATE\_MONITOR\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_PM_VR_CREATE_MONITOR_001			
<b>Test Purpose</b>	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clauses 5.3.9, 7.7.2, 7.7.5, 7.7.6, 8.5) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.2, 7.5.4, 7.5.5, 8.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM * [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO * [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO * [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * Monitoring parameters (e.g. performance metrics, metric groups) are defined in the NS descriptor (e.g. CPU usage, memory usage, etc.)			
	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>

<b>Test Sequence</b>	1	Stimulus	Trigger MANO to monitor one or more performance metrics defined in the NS descriptor and referring to virtualised resources that are allocated to the target NS instance	
	2	IOP Check	Verify that the performance metrics are getting collected by the VIM	
	3	IOP Check	Verify that performance metrics for the monitored virtualised resource are properly collected by MANO, e.g. by quering MANO performance metrics database (if any) or checking directly the MANO Graphical User Interface (if applicable)	
<b>IOP Verdict</b>				

## 6.1.9.1.2 TD\_NFV\_PM\_VR\_CREATE\_THRESHOLD\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_PM_VR_CREATE_THRESHOLD_001			
<b>Test Purpose</b>	To verify that performance metrics of one or more virtualised resources that are allocated to a NS instance can be monitored using thresholds based notifications			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clauses 5.3.9, 7.7.6, 7.7.7, 8.5) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.7, 8.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM * [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO * [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO * [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * Monitoring parameters (e.g. performance metrics, metric groups) are defined in the NS Descriptor (e.g. CPU usage, memory usage, etc.) and one or more of them are already monitored by MANO (TD_NFV_PM_VR_CREATE_MONITOR) * Monitoring performance metrics thresholds are defined			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to create a monitoring threshold for a performance metric of a virtualised resource allocated to the target NS instance that is already monitored	
	2	Stimulus	(if required and not implicitly carried out at step 1) Trigger MANO to subscribe to the monitoring threshold crossing notification for the performance monitoring threshold created	
	3	Stimulus	Trigger the monitored virtualised resource to cross the specified threshold (e.g. by increasing resource utilisation levels in the related virtualization container)	
	4	IOP Check	Verify that a "threshold crossed" notification for the monitored virtualised resource was processed by MANO (e.g. quering the related MANO monitoring database) and properly dispatched (e.g. checking the MANO Graphical User Interface, if applicable)	
<b>IOP Verdict</b>				

## 6.1.9.1.3 TD\_NFV\_PM\_VR\_DELETE\_MONITOR\_001

Interoperability Test Description				
-----------------------------------	--	--	--	--

<b>Identifier</b>	TD_NFV_PM_VR_DELETE_MONITOR_001			
<b>Test Purpose</b>	To verify that monitoring of performance metrics of one or more virtualised resources that are allocated to a NS instance can be stopped			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clauses 5.3.9, 7.7.4, 7.7.5, 7.7.6, 8.5) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.3, 7.5.4, 7.5.5, 8.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM * [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO * [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO * [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * Monitoring parameters (e.g. performance metrics, metric groups) are defined in the NS Descriptor (e.g. CPU usage, memory usage, etc.) and one or more of them are already monitored by MANO (TD_NFV_PM_VR_CREATE_MONITOR)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to stop monitoring a performance metric of a virtualised resource that is allocated to the target NS instance	
	2	IOP Check	Verify that the selected performance metric is no more collected by the VIM	
	3	IOP Check	Verify that the performance metric is no more collected by MANO, e.g. by querying MANO performance metrics database (if any) or checking directly the MANO Graphical User Interface (if applicable)	
<b>IOP Verdict</b>				

## 6.1.9.1.4 TD\_NFV\_PM\_VR\_DELETE\_THRESHOLD\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_PM_VR_DELETE_THRESHOLD_001			
<b>Test Purpose</b>	To verify that performance monitoring thresholds created for one or more virtualised resources that are allocated to a NS instance can be deleted			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clauses 5.3.9, 7.7.6, 7.7.9, 8.5) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.5, 7.5.8, 8.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_19] MANO supports receiving VM/VNFC KPIs from VIM * [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO * [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO * [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * A performance monitoring threshold for a virtualised resource that is part of the NS instance is created by MANO (TD_NFV_PM_VR_CREATE_THRESHOLD)			
	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>

<b>Test Sequence</b>	1	Stimulus	Trigger MANO to delete a monitoring threshold related to a performance metric of a virtualised resource that is allocated to the target NS instance	
	2	IOP Check	Verify that the relevant virtual resource performance monitoring threshold has been deleted on the VIM	
	3	Stimulus	Trigger the virtualised resource to cross the specified threshold (e.g. by increasing resource utilisation levels in the related virtualisation container)	
	4	IOP Check	Verify that MANO is not receiving and therefore not processing (and dispatching) any monitoring threshold crossed notification, e.g. by quering the MANO Graphical User Interface	
<b>IOP Verdict</b>				

## 6.1.9.2 PERFORMANCE MANAGEMENT – VNF KPI

### 6.1.9.2.1 TD\_NFV\_PM\_VNF\_KPI\_CREATE\_MONITOR\_001

<b>Interoperability Test Description</b>				
<b>Identifier</b>	TD_NFV_PM_VNF_KPI_CREATE_MONITOR_001			
<b>Test Purpose</b>	To verify that a VNF indicator related to a NS instance can be monitored			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA008 V2.3.1 (clauses 5.3.1.4, 5.4.1.4, 6.3.2, 6.3.3, 6.3.4, 8.2.2, 8.2.3, 8.2.4, 9.6) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.9, 7.7.2, 7.7.3, 7.7.4, 8.10)			
<b>Applicability</b>	* MANO supports receiving or collecting VNF KPIs * VNF/EM can send indicators (KPIs) to MANO asynchronously or by explicit request			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * One or more VNF indicators are defined in the related VNF Descriptors and referenced in the NS Descriptor			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to monitor a VNF indicator defined in the NS descriptor and related to a VNF instance inside the target NS instance	
	2	IOP Check	Verify that VNF indicator monitored values are properly collected by MANO, e.g. by quering MANO VNF indicator database (if any) or checking directly the MANO Graphical User Interface (if applicable)	
<b>IOP Verdict</b>				

### 6.1.9.2.2 TD\_NFV\_PM\_VNF\_KPI\_DELETE\_MONITOR\_001

<b>Interoperability Test Description</b>	
<b>Identifier</b>	TD_NFV_PM_VNF_KPI_DELETE_MONITOR_001
<b>Test Purpose</b>	To verify that monitoring of a VNF indicator related to a NS instance can be stopped
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I

<b>References</b>	ETSI GS NFV-IFA008 V2.3.1 (clauses 5.3.1.4, 5.4.1.4, 6.3.5, 8.2.5, 9.6) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.9, 7.7.5, 8.10)			
<b>Applicability</b>	* MANO supports receiving or collecting VNF KPIs * VNF/EM can send indicators (KPIs) to MANO asynchronously or by explicit request			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * One or more VNF indicators are defined in the related VNF Descriptors and referenced in the NS Descriptor and one or more of them are already monitored by MANO (TD_NFV__PM_VNF_KPI_CREATE_MONITOR)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to stop monitoring a VNF indicator for a VNF instance inside the target NS instance	
	2	IOP Check	Verify that the VNF indicator is no more collected and processed by MANO, e.g. by quering MANO VNF indicator database (if any) or checking directly the MANO Graphical User Interface (if applicable)	
<b>IOP Verdict</b>				

## 6.1.9.2.3 TD\_NFV\_PM\_VNF\_KPI\_CREATE\_THRESHOLD\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_PM_VNF_KPI_CREATE_THRESHOLD_001			
<b>Test Purpose</b>	To verify that a VNF indicator related to a NS instance can be monitored using thresholds based notifications			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.7, 8.4)			
<b>Applicability</b>	* MANO supports receiving or collecting VNF KPIs * VNF/EM can send indicators (KPIs) to MANO asynchronously or by explicit request			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * One or more VNF indicators are defined in the related VNF Descriptors and referenced in the NS Descriptor and one or more of them are already monitored by MANO (TD_NFV__PM_VNF_KPI_CREATE_MONITOR) * VNF monitoring indicators metrics thresholds are define			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to create a monitoring threshold for a VNF indicator related to the relevant VNF instance inside the target NS instance that is already monitored	
	2	Stimulus	(if required and not implicitly carried out at step 1) Trigger MANO to subscribe to the VNF indicator threshold crossing notification for the VNF indicator threshold created	
	3	Stimulus	Trigger the VNF indicator to cross the specified threshold (e.g. by accessing the related VNF instance and manually stimulate the VNF indicator change)	

	4	IOP Check	Verify that a “threshold crossed” notification for the VNF indicator was generated and processed by MANO (e.g. querying the related MANO monitoring database) and properly dispatched (e.g. checking the MANO Graphical User Interface, if applicable)	
<b>IOP Verdict</b>				

#### 6.1.9.2.4 TD\_NFV\_PM\_VNF\_KPI\_DELETE\_THRESHOLD\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_PM_VNF_KPI_DELETE_THRESHOLD_001			
<b>Test Purpose</b>	To verify that performance monitoring thresholds created for one or more VNF indicator related to a NS instance can be deleted			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.5, 7.5.8, 8.4)			
<b>Applicability</b>	* MANO supports receiving or collecting VNF KPIs * VNF/EM can send indicators (KPIs) to MANO asynchronously or by explicit request			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001) * A performance monitoring threshold for a VNF indicator related to a VNF instance allocated to the relevant NS instance is created by MANO (TD_NFV_PM_VNF_KPI_CREATE_THRESHOLD)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to delete a monitoring threshold linked to a performance metric of a VNF indicator related to a VNF instance that is allocated to the target NS instance	
	2	IOP Check	Verify that the relevant VNF indicator performance monitoring threshold has been deleted on MANO (e.g. querying its monitoring service, if applicable)	
	3	Stimulus	Trigger the VNF indicator to cross the deleted threshold (e.g. by accessing the related VNF instance and manually stimulate the VNF indicator change)	
	4	IOP Check	Verify that MANO is not receiving and therefore not processing (and dispatching) any VNF indicator threshold crossed notification, e.g. querying the related MANO monitoring database or checking the MANO Graphical User Interface, if applicable	
<b>IOP Verdict</b>				

### 6.1.10 TERMINATE

#### 6.1.10.1 TD\_NFV\_NS\_LCM\_TERMINATE\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_NS_LCM_TERMINATE_001
<b>Test Purpose</b>	To verify that a NS can be successfully terminated
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS

	SUT_S-VNFM-D SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.7) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.5, clause 7.4.1.5, clause 7.5.1.5) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.5, clause 7.4.1.5, clause 7.5.1.5) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.7)			
<b>Applicability</b>				
<b>Pre-test conditions</b>	* NS has been instantiated			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

## 6.1.11 DELETE

### 6.1.11.1 TD\_NFV\_TEARDOWN\_DELETE\_NSD\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_TEARDOWN_DELETE_NSD_001			
<b>Test Purpose</b>	To delete a NSD			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-D SUT_S-VNFM-I SUT_AUTO-LCM-VALIDATION			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.2.6)			
<b>Applicability</b>				
<b>Pre-test conditions</b>	* NSD is created in MANO (TD_NFV_ONBOARD_NSD_001) * All NSs associated with the NSD have been terminated			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

## 6.1.11.2 TD\_NFV\_TEARDOWN\_DELETE\_VNF\_PKG\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_TEARDOWN_DELETE_VNF_PKG_001			
<b>Test Purpose</b>	To delete a VNF Package			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS SUT_MULTI-SITE SUT_S-VNFM-D SUT_S-VNFM-I			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.7.5)			
<b>Applicability</b>				
<b>Pre-test conditions</b>	* VNF package has been on-boarded in MANO (TD_NFV_ONBOARD_VNF_PKG_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

## 6.2 EPA

## 6.2.1 INSTANTIATE

## 6.2.1.1 TD\_NFV\_EPA\_NS\_LCM\_INSTANTIATE\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_EPA_NS_LCM_INSTANTIATE_001
<b>Test Purpose</b>	To verify that an NS can be successfully instantiated with EPA requirements
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.3) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.3)
<b>Applicability</b>	* [IFS_NFV_MANO_28] MANO supports deploying VNFs with EPA requirements towards NFVI/VIM * [IFS_NFV_VIM_NFVI_1] NFVI/VIM supports EPA attributes * [IFS_NFV_VNF_1] VNF requires EPA
<b>Pre-test conditions</b>	* NSD, VLD(s), VNFFGD(s) and VNF Package(s) have been on-boarded in MANO * On-boarded VNFD(s) include EPA requirements * The software image repository is reachable by the VIMs * The required resources are available on the NFVIs

	Step	Type	Description	Result
<b>Test Sequence</b>	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 required EPA attributes have been configured as expected, e.g. checking configuration of: - SR-IOV - CPU pinning - NUMA topology - PCI passthrough - ....	
	7	IOP Check	Verify that the VNF(s) are running and reachable through the management network	
	8	IOP Check	Verify that the VNF(s) have been configured according to VNFD(s) (i.e by obtaining a result from the management interface)	
	9	IOP Check	Verify that the VNF(s), VL(s) and VNFFG(s) have been connected according to the Descriptors	
	10	IOP Check	Verify that the NS is successfully instantiated by running the end-to-end functional test	
	11	IOP Check	Verify that the EPA requirements are matched in the NS instance and the running VNFs (e.g. performance check)	
<b>IOP Verdict</b>				

## 6.2.2 SCALE NS

### 6.2.2.1 SCALE NS MANUALLY

#### 6.2.2.1.1 TD\_NFV\_EPA\_NS\_LCM\_SCALE\_OUT\_001

<b>Interoperability Test Description</b>	
<b>Identifier</b>	TD_NFV_EPA_NS_LCM_SCALE_OUT_001
<b>Test Purpose</b>	To verify that a NS can be successfully scaled out with EPA requirements (by adding VNF instances) if triggered automatically by a MANO operator
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)
<b>Applicability</b>	* [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances * [IFS_NFV_MANO_28] MANO supports deploying VNFs with EPA requirements towards NFVI/VIM * [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances * [IFS_NFV_VNF_1] VNF requires EPA * [IFS_NFV_VIM_NFVI_1] NFVI/VIM supports EPA attributes
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_EPA_NS_LCM_INSTANTIATE_EPA_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 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 required EPA attributes have been configured as expected, e.g. checking configuration of: - SR-IOV - CPU pinning - NUMA topology - PCI passthrough - ....	
	5	IOP Check	Verify that the additional VNF instance(s) have been deployed	
	6	IOP Check	Verify that the additional VNF instances(s) are running and reachable from the management network	
	7	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors (i.e. by getting a result through the management interface)	
	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 EPA requirements are matched in the scaled VNF(s) (e.g. performance check)	
	10	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test	
<b>IOP Verdict</b>				

## 6.2.2.1.2 TD\_NFV\_EPA\_NS\_LCM\_SCALE\_IN\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_EPA_NS_LCM_SCALE_IN_001			
<b>Test Purpose</b>	To verify that a NS can be successfully scaled in with EPA requirements (by removing VNF instances) if triggered automatically by a MANO operator			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances * [IFS_NFV_MANO_28] MANO supports deploying VNFs with EPA requirements towards NFVI/VIM * [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances * [IFS_NFV_VNF_1] VNF requires EPA * [IFS_NFV_VIM_NFVI_1] NFVI/VIM supports EPA attributes			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_EPA_NS_LCM_INSTANTIATE_001) * NS has been scaled out by adding VNF instances with EPA requirements			
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 scale in (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 EPA configurations of VNF(s) to be scaled-in have been deallocated/released as expected, e.g. checking de-configuration of: - SR-IOV - CPU pinning - NUMA topology - PCI passthrough - ....	
	5	IOP Check	Verify that the impacted VNF related resources have been released by the VIM	
	6	IOP Check	Verify that the remaining VNF instances(s) are still running and reachable through the management network	
	7	IOP Check	Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors	
	10	IOP Check	Verify that NS has been scaled in by running the end-to-end functional test	
<b>IOP Verdict</b>				

## 6.2.3 SCALE VNF

### 6.2.3.1 SCALE VNF MANUALLY

#### 6.2.3.1.1 TD\_NFV\_EPA\_NS\_LCM\_SCALE\_OUT\_VNF\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_EPA_NS_LCM_SCALE_OUT_VNF_001			
<b>Test Purpose</b>	To verify that a NS can be successfully scaled out with EPA requirements (by adding VNF instances) if triggered automatically by a MANO operator			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_MANO_28] MANO supports deploying VNFs with EPA requirements towards NFVI/VIM</li> <li>* [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances</li> <li>* [IFS_NFV_VNF_1] VNF requires EPA</li> <li>* [IFS_NFV_VIM_NFVI_1] NFVI/VIM supports EPA attributes</li> </ul>			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_EPA_NS_LCM_INSTANTIATE_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger NS scale out (by adding VNF instances) in MANO with an operator action	
	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 required EPA attributes have been configured as expected, e.g. checking configuration of: - SR-IOV - CPU pinning - NUMA topology - PCI passthrough - .....	
	5	IOP Check	Verify that the additional VNF instance(s) have been deployed	
	6	IOP Check	Verify that the additional VNF instances(s) are running and reachable from the management network	
	7	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors (i.e. by getting a result through the management interface)	
	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 EPA requirements are matched in the scaled VNF(s) (e.g. performance check)	
	10	IOP Check	Verify that NS has been scaled out by running the end-to-end functional test	
<b>IOP Verdict</b>				

## 6.2.3.1.2 TD\_NFV\_EPA\_NS\_LCM\_SCALE\_IN\_VNF\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_EPA_NS_LCM_SCALE_IN_VNF_001			
<b>Test Purpose</b>	To verify that a NS can be successfully scaled in with EPA requirements (by removing VNF instances) if triggered automatically by a MANO operator			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances * [IFS_NFV_MANO_28] MANO supports deploying VNFs with EPA requirements towards NFVI/VIM * [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances * [IFS_NFV_VNF_1] VNF requires EPA * [IFS_NFV_VIM_NFVI_1] NFVI/VIM supports EPA attributes			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_EPA_NS_LCM_INSTANTIATE_001) * NS has been scaled out by adding VNF instances with EPA requirements			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger NS scale in (by removing VNFs) in MANO with an operator action	
	2	IOP Check	Verify that the scale in (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 EPA configurations of VNF(s) to be scaled-in have been deallocated/released as expected, e.g. checking de-configuration of: - SR-IOV - CPU pinning - NUMA topology - PCI passthrough - .....	

	5	IOP Check	Verify that the impacted VNF related resources have been released by the VIM	
	6	IOP Check	Verify that the remaining VNF instances(s) are still running and reachable through the management network	
	7	IOP Check	Verify that the remaining VNF instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors	
	10	IOP Check	Verify that NS has been scaled in by running the end-to-end functional test	
<b>IOP Verdict</b>				

## 6.3 SFC

### 6.3.1 INSTANTIATE

#### 6.3.1.1 TD\_NFV\_SFC\_NS\_LCM\_INSTANTIATE\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_SFC_NS_LCM_INSTANTIATE_001			
<b>Test Purpose</b>	To verify that an NS with NSH based SFC can be successfully instantiated			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.3) ETSI GS NFV-IFA005 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA006 V2.3.1 (clause 7.2.3) ETSI GS NFV-IFA008 V2.3.1 (clause 6.2.2) ETSI GS NFV-IFA010 V2.3.1 (clause 6.3.2) IETF RFC 7665 SFC <a href="https://datatracker.ietf.org/doc/rfc7665/">https://datatracker.ietf.org/doc/rfc7665/</a> IETF NSH draft <a href="https://datatracker.ietf.org/doc/rfc8300/">https://datatracker.ietf.org/doc/rfc8300/</a>			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>• [IFS_NFV_MANO_30] MANO supports provisioning of network forwarding paths</li> <li>• [IFS_NFV_VIM_NFVI_9] NFVI/VIM exposes network forwarding path functionalities to MANO</li> <li>• NFVI/VIM supports NSH</li> <li>• VNF supports Network Service Headers (NSH) encapsulation</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NSD, VLD(s), VNFFGD(s) and VNF Package(s) have been on-boarded in MANO</li> <li>* The software image repository is reachable by the VIM</li> <li>* The required resources are available on the NFVI</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 (NSH Traffic)	
<b>IOP Verdict</b>				

## 6.3.2 TERMINATE

### 6.3.2.1 TD\_NFV\_SFC\_NS\_LCM\_TERMINATE\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_SFC_NS_LCM_TERMINATE_001			
<b>Test Purpose</b>	To verify that a NS can be successfully terminated			
<b>Configuration</b>	SUT_SINGLE-VENDOR_NS SUT_MULTI-VENDOR_NS			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.7) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.5, clause 7.4.1.5, clause 7.5.1.5) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.5, clause 7.4.1.5, clause 7.5.1.5) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.7)			
<b>Applicability</b>				
<b>Pre-test conditions</b>	* NS has been instantiated			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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	
<b>IOP Verdict</b>				

## 6.4 MULTI SITE

### 6.4.1 INSTANTIATE

#### 6.4.1.1 TD\_NFV\_MULTISITE\_NS\_LCM\_INSTANTIATE\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_MULTISITE_NS_LCM_INSTANTIATE_001
<b>Test Purpose</b>	To verify that an NS can be successfully instantiated across different sites

<b>Configuration</b>	SUT_MULTI-SITE			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.3) ETSI GS NFV-IFA005 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA006 V2.3.1 (clause 7.2.3) ETSI GS NFV-IFA008 V2.3.1 (clause 6.2.3) ETSI GS NFV-IFA010 V2.3.1 (clause 6.3.2) ETSI GS NFV-IFA022 V0.8.0 (clause 5.2)			
<b>Applicability</b>	* [IFS_NFV_MANO_1] MANO supports multi-site deployments (i.e. two or more geographically distributed sites managed by different VIM instances) * [IFS_NFV_VIM_NFVI_2] NFVI/VIM supports multi-site deployments			
<b>Pre-test conditions</b>	* NSD, VLD(s), VNFFGD(s) and VNF Package(s) have been on-boarded in MANO * The software image repository is reachable by the VIMs * The required resources are available on the NFVIs			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger multi-site NS instantiation in MANO	
	2	IOP Check	Verify that the software images have been onboarded in the VIMs	
	3	IOP Check	Verify that the requested resources have been allocated by the VIMs 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) have been deployed according to the multi-site location constraints	
	7	IOP Check	Verify that the VNF(s) are running and reachable through the management network	
	8	IOP Check	Verify that the VNF(s) have been configured according to VNFD(s) (i.e by obtaining a result from the management interface)	
	9	IOP Check	Verify that the VNF(s), VL(s) and VNFFG(s) have been connected according to the Descriptors	
	10	IOP Check	Verify that the VNF(s) have multi-site connectivity through the multi-site VL(s)	
	11	IOP Check	Verify that the multi-site NS is successfully instantiated by running the end-to-end functional test	
<b>IOP Verdict</b>				

## 6.4.2 SCALE NS MANUALLY

### 6.4.2.1 TD\_NFV\_MULTISITE\_NS\_LCM\_SCALE\_OUT\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_MULTISITE_NS_LCM_SCALE_OUT_001
<b>Test Purpose</b>	To verify that a multi-site NS can be successfully scaled out (by adding VNF instances) if triggered by a MANO operator
<b>Configuration</b>	SUT_MULTI-SITE

<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.3.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.3.1 (clauses 6.2.3, 6.3.3) ETSI GS NFV-IFA022 V0.8.0 (clause 5.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_1] MANO supports multi-site deployments (i.e. two or more geographically distributed sites managed by different VIM instances) *[IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances * [IFS_NFV_VIM_NFVI_2] NFVI/VIM supports multi-site deployments * [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances			
<b>Pre-test conditions</b>	* Multi-site NS is instantiated (TD_NFV_MULTISITE_NS_LCM_INSTANTIATE_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger multi-site 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 according to the descriptors (VMs, VLs, CPs...)	
	4	IOP Check	Verify that the additional VNF instance(s) have been deployed according to the multi-site location constraints	
	5	IOP Check	Verify that the additional VNF instances(s) are running and reachable from the management network	
	6	IOP Check	Verify that the additional VNF instances(s) have been configured according to the descriptors (i.e. by getting a result through 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 the additional VNF instance(s) have multi-site connectivity through the multi-site VL(s)	
	9	IOP Check	Verify that multi-site NS has been scaled out by running the end-to-end functional test	
<b>IOP Verdict</b>				

#### 6.4.2.2 TD\_NFV\_MULTISITE\_NS\_LCM\_SCALE\_IN\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_MULTISITE_NS_LCM_SCALE_IN_001
<b>Test Purpose</b>	To verify that a multi-site NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator
<b>Configuration</b>	SUT_MULTI-SITE
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.3.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA008 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.3.1 (clauses 6.2.3, 6.3.3) ETSI GS NFV-IFA022 V0.8.0 (clause 5.4)
<b>Applicability</b>	* [IFS_NFV_MANO_1] MANO supports multi-site deployments (i.e. two or more geographically distributed sites managed by different VIM instances) *[IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances * [IFS_NFV_VIM_NFVI_2] NFVI/VIM supports multi-site deployments * [IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances

Pre-test conditions				
* Multi-site NS is instantiated (TD_NFV_MULTISITE_NS_LCM_INSTANTIATE_001) * Multi-site NS has been scaled out by adding VNF instances (TD_NFV_MULTISITE_NS_LCM_SCALE_OUT_001)				
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Trigger multi-site 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 and not running in the correspondant VIM site / instance	
	3	IOP Check	Verify that the impacted VNF related resources have been released by the proper VIM site / VIM instance	
	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 the remaining VNF instance(s) have still multi-site connectivity through the multi-site VL(s)	
	7	IOP Check	Verify that multi-site NS has been scaled in by running the end-to-end functional test	
IOP Verdict				

### 6.4.3 SCALE VNF MANUALLY

#### 6.4.3.1 TD\_NFV\_MULTISITE\_NS\_LCM\_SCALE\_OUT\_VNF\_001

Interoperability Test Description				
Identifier	TD_NFV_MULTISITE_NS_LCM_SCALE_OUT_VNF_001			
Test Purpose	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			
Configuration	SUT_MULTI-SITE			
References	ETSI GS NFV-IFA005 V2.3.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.3.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA008 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.3.1 (clauses 6.2.3, 6.3.3) ETSI GS NFV-IFA022 V0.8.0 (clause 5.4)			
Applicability	* [IFS_NFV_MANO_1] MANO supports multi-site deployments (i.e. two or more geographically distributed sites managed by different VIM instances) * [IFS_NFV_MANO_15] MANO supports scaling by adding/removing VNFC instances * [IFS_NFV_VIM_NFVI_2] NFVI/VIM supports multi-site deployments * [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances			
Pre-test conditions				
* Multi-Site NS is instantiated (TD_NFV_MULTISITE_NS_LCM_INSTANTIATE_001) * MANO is configured to trigger SCALE OUT (by adding VNFC instances) when triggered by a MANO operator				
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Trigger multi-site 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 VNFC instance(s) have been deployed according to the descriptors (VMs, VLs, CPs...)	
	4	IOP Check	Verify that the additional VNFC instance(s) have been deployed according to the multi-site location constraints	
	5	IOP Check	Verify that the additional VNFC instances(s) are running and reachable from the management network	
	6	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)	
	7	IOP Check	Verify that the additional VNFC instances(s), VL(s) and VNFFG(s) are connected according to the Descriptors	
	8	IOP Check	Verify that the additional VNFC instance(s) have multi-site connectivity through the multi-site VL(s)	
	9	IOP Check	Verify that NS has been scaled in by running the end-to-end functional test	
<b>IOP Verdict</b>				

#### 6.4.3.2 TD\_NFV\_MULTISITE\_NS\_LCM\_SCALE\_IN\_VNF\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_MULTISITE_NS_LCM_SCALE_IN_VNF_001			
<b>Test Purpose</b>	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			
<b>Configuration</b>	SUT_MULTI-SITE			
<b>References</b>	ETSI GS NFV-IFA005 V2.3.1 (clause 5.3.4) ETSI GS NFV-IFA006 V2.3.1 (clauses 7.3.1, 7.4.1) ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA008 V2.3.1 (clause 7.2.4) ETSI GS NFV-IFA010 V2.3.1 (clauses 6.2.3, 6.3.3) ETSI GS NFV-IFA022 V0.8.0 (clause 5.4)			
<b>Applicability</b>	* [IFS_NFV_MANO_1] MANO supports multi-site deployments (i.e. two or more geographically distributed sites managed by different VIM instances) * [IFS_NFV_MANO_15] MANO supports scaling by adding/removing VNFC instances * [IFS_NFV_VIM_NFVI_2] NFVI/VIM supports multi-site deployments * [IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances			
<b>Pre-test conditions</b>	* Multi-site NS is instantiated (TD_NFV_MULTISITE_NS_LCM_INSTANTIATE_001) * Multi-site NS has been scaled out by adding VNFC instances (TD_NFV_MULTISITE_NS_LCM_SCALE_OUT_VNF_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger NS scale in (by removing VNFC instances (VMs) from a VNF in the NS) in MANO with an operator action	
	2	IOP Check	Verify that the impacted VNFC instance(s) have been terminated and not running in the correspondent VIM site / instance	
	3	IOP Check	Verify that the impacted VNFC related resources have been released by the proper VIM site / VIM instance	
	4	IOP Check	Verify that the remaining VNFC instances(s) are still running and reachable through the management network	
	5	IOP Check	Verify that the remaining VNFC instances(s), VL(s) and VNFFG(s) are still connected according to the descriptors	
	6	IOP Check	Verify that the remaining VNFC instance(s) have still multi-site connectivity through the multi-site VL(s)	

	7	IOP Check	Verify that multi-site NS has been scaled in by running the end-to-end functional test	
<b>IOP Verdict</b>				

## 6.4.4 TERMINATE

### 6.4.4.1 TD\_NFV\_MULTISITE\_NS\_LCM\_TERMINATE\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_MULTISITE_NS_LCM_TERMINATE_001			
<b>Test Purpose</b>	To verify that a Multi Site NS can be successfully terminated			
<b>Configuration</b>	SUT_MULTI-SITE			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.7) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.5, 7.4.1.5, 7.5.1.5) ETSI GS NFV-IFA008 V2.3.1 (clause 7.2.7)			
<b>Applicability</b>	* [IFS_NFV_MANO_1] MANO supports multi-site deployments (i.e. two or more geographically distributed sites managed by different VIM instances) *[IFS_NFV_MANO_15] MANO supports scaling by adding/removing VNFC instances			
<b>Pre-test conditions</b>	* Multi Site NS has been instantiated			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger Multi Site NS termination in MANO	
	2	IOP Check	Verify that all the VNF instance(s) have been terminated in the given sites	
	3	IOP Check	Verify that the resources that were allocated to the Multi Site NS and VNF(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	
<b>IOP Verdict</b>				

## 6.5 s-VNFM-D

### 6.5.1 INSTANTIATE

#### 6.5.1.1 TD\_NFV\_S-VNFM-D\_NS\_LCM\_INSTANTIATE\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001
<b>Test Purpose</b>	To verify that an NS can be successfully instantiated
<b>Configuration</b>	SUT_S-VNFM-D

<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.3) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.3)			
<b>Applicability</b>				
<b>Pre-test conditions</b>	* 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			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 VNFM in the VIM according to the descriptors	
	4	IOP Check	Verify that the VNF(s) have been deployed by the VNFM 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	
<b>IOP Verdict</b>				

## 6.5.2 SCALE NS

### 6.5.2.1 SCALE NS MANUALLY

#### 6.5.2.1.1 TD\_NFV\_S-VNFM-D\_NS\_LCM\_SCALE\_OUT\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_S-VNFM-D_NS_LCM_SCALE_OUT_001
<b>Test Purpose</b>	To verify that a NS can be successfully scaled out (by adding VNF instances) if triggered by a MANO operator
<b>Configuration</b>	SUT_S-VNFM-D
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)
<b>Applicability</b>	*[IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances *[IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances

<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_NS_LCM_INSTANTIATE_001)			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 VNFM in the VIM according to the descriptors	
	3	IOP Check	Verify that the additional VNF instance(s) have been deployed by the VNFM	
	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 getting 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	
<b>IOP Verdict</b>				

## 6.5.2.1.2 TD\_NFV\_S-VNFM-D\_NS\_LCM\_SCALE\_IN\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_NS_LCM_SCALE_IN_001			
<b>Test Purpose</b>	To verify that a NS can be successfully scaled in (by removing VNF instances) if triggered by a MANO operator			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	*[IFS_NFV_MANO_14] MANO supports scaling by adding/removing VNF instances *[IFS_NFV_VNF_4] VNF can scale out/in by adding/removing VNF instances			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001) * NS has been scaled out by adding VNF instances			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 by the VNFM	
	3	IOP Check	Verify that the impacted VNF related resources have been released by the VNFM in 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	
<b>IOP Verdict</b>				

## 6.5.3 SCALE VNF

### 6.5.3.1 SCALE VNF MANUALLY

#### 6.5.3.1.1 TD\_NFV\_S-VNFM-D\_NS\_LCM\_SCALE\_OUT\_VNF\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_NS_LCM_SCALE_OUT_VNF_001			
<b>Test Purpose</b>	To verify that a VNF in a NS can be successfully scaled out (by adding VNFC instances (VMs)) when triggered by a MANO operator			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	*[IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances *[IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_S-VNFM-D_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 VNFM in the VIM according to the descriptors	
	3	IOP Check	Verify that the additional VM(s) have been deployed by the VNFM (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	
<b>IOP Verdict</b>				

#### 6.5.3.1.2 TD\_NFV\_S-VNFM-D\_NS\_LCM\_SCALE\_IN\_VNF\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_NS_LCM_SCALE_IN_VNF_001			
<b>Test Purpose</b>	To verify that a VNF in a NS can be successfully scaled in (by removing VNFC instances (VMs)) when triggered by a MANO operator			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clause 7.3.4) ETSI GS NFV-IFA005 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA006 V2.3.1 (clause 7.3.1.2, 7.4.1.2, 7.5.1.2) ETSI GS NFV-IFA007 V2.3.1 (clause 7.2.4)			
<b>Applicability</b>	*[IFS_NFV_MANO_15] MANO supports scaling out/in by adding/removing VNFC instances *[IFS_NFV_VNF_5] VNF can scale out/in by adding/removing VNFC instances			

<b>Pre-test conditions</b>	* NS has been scaled out by adding VM			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 by the VNFM	
	3	IOP Check	Verify that the impacted VM related resources have been released by the VNFM in 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	
<b>IOP Verdict</b>				

## 6.5.4 PERFORMANCE MANAGEMENT

### 6.5.4.1 PERFORMANCE MANAGEMENT – VNF VR

#### 6.5.4.1.1 TD\_NFV\_S-VNFM-D\_PM\_VNF\_VR\_CREATE\_MONITOR\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_PM_VNF_VR_CREATE_MONITOR_001			
<b>Test Purpose</b>	To verify that the performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be monitored through external VNFM			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA006 V2.3.1 (clauses 5.3.8, 7.7.2, 7.7.5, 7.7.6, 8.5) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.6, 7.4.2, 7.4.4, 7.4.5, 8.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.2, 7.5.4, 7.5.5, 8.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_21] MANO supports receiving VNF KPIs from external VNFMs</li> <li>* [IFS_NFV_VNFM_18] VNFM exposes VNF KPIs and indicators towards MANO</li> <li>* [IFS_NFV_VNFM_14] VNFM supports receiving VM/VNFC KPIs from VIM</li> <li>* [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001)</li> <li>* VNF monitoring parameters (e.g. performance metrics, metric groups) are defined in the NS descriptor (e.g. CPU usage, memory usage, etc.)</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to monitor a VNF performance metric defined in the NS descriptor and referring to a virtualised resource allocated to the relevant VNF instance inside the target NS instance	

	2	IOP Check	Verify that the VNFM is triggered by the MANO to start monitor the given VNF performance metrics on the VIM	
	3	IOP Check	Verify that the performance metric for the monitored virtualised resource is properly collected and processed by the VNFM, and exposed to the MANO, e.g. by querying its performance metrics database (if any)	
	4	IOP Check	In turn, verify that the VNF performance metric is collected and processed by the MANO from the VNFM, e.g. by querying its performance metrics database (if any) or checking its Graphical User Interface (if applicable)	
<b>IOP Verdict</b>				

## 6.5.4.1.2 TD\_NFV\_S-VNFM-D\_PM\_VNF\_VR\_CREATE\_THRESHOLD\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-I_PM_VNF_VR_CREATE_THRESHOLD_001			
<b>Test Purpose</b>	To verify that the performance metrics of a virtualised resource that is allocated to a VNF instance inside a NS instance can be monitored using thresholds based notifications through external VNFM			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA006 V2.3.1 (clauses 5.3.8, 7.7.5, 7.7.6, 7.7.7, 8.5) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.6, 7.4.4, 7.4.5, 7.4.7, 8.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.7, 8.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_21] MANO supports receiving VNF KPIs from external VNFMs</li> <li>* [IFS_NFV_VNFM_18] VNFM exposes VNF KPIs and indicators towards MANO</li> <li>* [IFS_NFV_VNFM_14] VNFM supports receiving VM/VNFC KPIs from VIM</li> <li>* [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001)</li> <li>* VNF monitoring parameters (e.g. performance metrics, metric groups) are defined in the NS Descriptor (e.g. CPU usage, memory usage, etc.) and one or more of them are already monitored by MANO through VNFM (TD_NFV_S-VNFM-D_PM_VNF_VR_CREATE_MONITOR)</li> <li>* VNF monitoring performance metrics thresholds are defined</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to create a monitoring threshold for a VNF performance metric of a virtualised resource allocated to the relevant VNF instance inside the target NS instance that is already monitored through VNFM	
	2	IOP Check	Verify that the VNFM is triggered by the MANO to create the monitor threshold for the given VNF performance metrics on the VIM	
	3	Stimulus	(if required and not implicitly carried out at step 1) Trigger MANO to subscribe to the VNF monitoring threshold crossing notification for the performance monitoring threshold created on the VNFM	
	4	Stimulus	Trigger the virtualised resource to cross the specified threshold (e.g. by increasing resource utilisation levels in the virtualization container)	
	5	IOP Check	(If applicable) Verify that the “threshold crossed” notification for the monitored virtualised resource was generated by the VIM and collected by the VNFM	
	6	IOP Check	Verify that a “threshold crossed” notification for the VNF monitored virtualised resource was generated by the VNFM and processed by MANO (e.g. querying the related MANO monitoring database) and properly dispatched (e.g. checking the MANO Graphical User Interface, if applicable)	

<b>IOP Verdict</b>	
--------------------	--

## 6.5.4.1.3 TD\_NFV\_S-VNFM-D\_PM\_VNF\_VR\_DELETE\_MONITOR\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_PM_VNF_VR_DELETE_MONITOR_001			
<b>Test Purpose</b>	To 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 through external VNFM			
<b>Configuration</b>	S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA006 V2.3.1 (clauses 5.3.8, 7.7.5, 7.7.6, 7.7.7, 8.5) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.6, 7.4.4, 7.4.5, 7.4.7, 8.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.7, 8.4)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_21] MANO supports receiving VNF KPIs from external VNFMs</li> <li>* [IFS_NFV_VNFM_18] VNFM exposes VNF KPIs and indicators towards MANO</li> <li>* [IFS_NFV_VNFM_14] VNFM supports receiving VM/VNFC KPIs from VIM</li> <li>* [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM</li> <li>* [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001)</li> <li>* VNF monitoring parameters (e.g. performance metrics, metric groups) are defined in the NS Descriptor (e.g. CPU usage, memory usage, etc.) and one or more of them are already monitored by MANO through VNFM (TD_NFV_S-VNFM-D_PM_VNF_VR_CREATE_MONITOR)</li> <li>* VNF monitoring performance metrics thresholds are defined</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to stop monitoring a performance metric of a virtualised resource that is allocated to the target NS instance	
	2	IOP Check	Verify that the external VNFM receives in turn a request to stop monitoring the selected performance metric from the VIM	
	3	IOP Check	Verify that the selected performance metric is no more collected by the VIM	
	4	IOP Check	Verify that the performance metric is no more collected by the VNFM, e.g. by querying VNFM performance metrics database (if any)	
	5	IOP Check	Verify that the performance metric is no more reported by the VNFM and collected by MANO, e.g. by querying MANO performance metrics database (if any) or checking directly the MANO Graphical User Interface (if applicable)	
<b>IOP Verdict</b>				

## 6.5.4.1.4 TD\_NFV\_S-VNFM-D\_PM\_VNF\_VR\_DELETE\_THRESHOLD\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_S-VNFM-D_PM_VNF_VR_DELETE_THRESHOLD_001
<b>Test Purpose</b>	To verify that a performance monitoring threshold created for a virtualised resource that is allocated to a VNF instance inside a NS instance can be deleted through external VNFM
<b>Configuration</b>	S-VNFM-D

<b>References</b>	ETSI GS NFV-IFA006 V2.3.1 (clauses 5.3.8, 7.7.5, 7.7.6, 7.7.9, 8.5) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.6, 7.4.4, 7.4.5, 7.4.8, 8.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.5, 7.5.8, 8.4)
<b>Applicability</b>	* [IFS_NFV_MANO_21] MANO supports receiving VNF KPIs from external VNFMs * [IFS_NFV_VNFM_18] VNFM exposes VNF KPIs and indicators towards MANO * [IFS_NFV_VNFM_14] VNFM supports receiving VM/VNFC KPIs from VIM * [IFS_NFV_VIM_NFVI_3] NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM * [IFS_NFV_VIM_NFVI_4] NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM * [IFS_NFV_VIM_NFVI_5] NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001) * A VNF performance monitoring threshold for a virtualised resource that is allocated to a VNF instance inside a NS instance is created by MANO through VNFM (TD_NFV_S-VNFM-D_PM_VNFM_VR_CREATE_THRESHOLD)

	Step	Type	Description	Result
<b>Test Sequence</b>	1	Stimulus	Trigger MANO to delete a monitoring threshold related to a performance metric of a virtualised resource that is allocated to the target NS instance	
	2	IOP Check	Verify that the request is correctly processed by the external VNFM and the relevant virtual resource performance monitoring threshold has been deleted on the VIM	
	3	Stimulus	Trigger the virtualised resource to cross the specified threshold (e.g. by increasing resource utilisation levels in the related virtualisation container)	
	4	IOP Check	Verify that the VNFM is not receiving and therefore not processing (and dispatching) any monitoring threshold crossed notification to MANO	
	5	IOP Check	In turn, verify that MANO is not receiving any monitoring threshold crossed notification, e.g. by quering the MANO Graphical User Interface	
<b>IOP Verdict</b>				

#### 6.5.4.2 PERFORMANCE MANAGEMENT – VNF KPI

##### 6.5.4.2.1 TD\_NFV\_S-VNFM-D\_PM\_VNF\_KPI\_CREATE\_MONITOR\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_S-VNFM-D_PM_VNF_KPI_CREATE_MONITOR_001
<b>Test Purpose</b>	To verify that a VNF indicator related to a NS instance can be monitored through external VNFM
<b>Configuration</b>	SUT_S-VNFM-D
<b>References</b>	ETSI GS NFV-IFA008 V2.3.1 (clauses 5.3.1.4, 5.4.1.4, 6.3.2, 6.3.3, 6.3.4, 8.2.2, 8.2.3, 8.2.4, 9.6) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.9, 7.7.2, 7.7.3, 7.7.4, 8.10)
<b>Applicability</b>	* [IFS_NFV_MANO_21] MANO supports receiving VNF KPIs from external VNFMs * [IFS_NFV_VNFM_18] VNFM exposes VNF KPIs and indicators towards MANO * [IFS_NFV_VNFM_12] VNFM supports receiving VNF indicators from VNF/EM * [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO/VNFM
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001) * Monitoring information (e.g. VNF instance) is defined and VNF indicators are available in the NSD

	Step	Type	Description	Result
<b>Test Sequence</b>	1	Stimulus	Trigger MANO to monitor a VNF indicator defined in the NS descriptor and related to a VNF instance inside the target NS instance	
	2	IOP Check	Verify that the VNFM receives the request from MANO for starting monitoring the related VNF indicator	
	3	IOP Check	Verify that VNF indicator monitored values are properly collected by the external VNFM and dispatched to the MANO, e.g. by querying VNFM VNF indicator database (if any)	
	4	IOP Check	Verify that VNF indicator monitored values are properly collected and processed by MANO, e.g. by querying MANO VNF indicator database (if any) or checking directly the MANO Graphical User Interface (if applicable)	
<b>IOP Verdict</b>				

## 6.5.4.2.2 TD\_NFV\_S-VNFM-D\_PM\_VNF\_KPI\_DELETE\_MONITOR\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_PM_VNF_KPI_DELETE_MONITOR_001			
<b>Test Purpose</b>	To verify that monitoring of a VNF indicator related to a NS instance can be stopped through external VNFM			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA008 V2.3.1 (clauses 5.3.1.4, 5.4.1.4, 6.3.5, 8.2.5, 9.6) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.9, 7.7.5, 8.10)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_21] MANO supports receiving VNF KPIs from external VNFMs</li> <li>* [IFS_NFV_VNFM_18] VNFM exposes VNF KPIs and indicators towards MANO</li> <li>* [IFS_NFV_VNFM_12] VNFM supports receiving VNF indicators from VNF/EM</li> <li>* [IFS_NFV_VNF_9] VNF can send indicators (KPIs) to MANO/VNFM</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001)</li> <li>* A VNF indicator specified in the NSD of the target NS instance is monitored by the MANO and external VNFM (TD_NFV_S-VNFM-D_PM_VNF_KPI_CREATE_NOTIFY_001)</li> </ul>			
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Trigger MANO to stop monitoring a VNF indicator for a VNF instance inside the target NS instance	
	2	IOP Check	Verify that VNFM receives from MANO the request for stopping monitoring the VNF indicator	
	3	IOP Check	Verify that the VNF indicator is no more collected by the external VNFM and dispatched to MANO, e.g. by querying VNFM VNF indicator database (if any)	
	4	IOP Check	Verify that the VNF indicator is no more collected and processed by MANO, e.g. by querying MANO VNF indicator database (if any) or checking directly the MANO Graphical User Interface (if applicable)	
<b>IOP Verdict</b>				

## 6.5.4.2.3 TD\_NFV\_S-VNFM-D\_PM\_VNF\_KPI\_CREATE\_THRESHOLD\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_S-VNFM-D_PM_VNF_KPI_CREATE_THRESHOLD_001

<b>Test Purpose</b>	To verify that a VNF indicator related to a NS instance can be monitored using thresholds based notifications through external VNFM			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.7, 8.4)			
<b>Applicability</b>	* MANO supports receiving or collecting VNF KPIs * VNF/EM can send indicators (KPIs) to MANO asynchronously or by explicit request			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001) * One or more VNF indicators are defined in the related VNF Descriptors and referenced in the NS Descriptor and one or more of them are already monitored by MANO (TD_NFV_S-VNFM-D_PM_VNF_KPI_CREATE_MONITOR) * VNF monitoring indicators metrics thresholds are define			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger MANO to create a monitoring threshold for a VNF indicator related to the relevant VNF instance inside the target NS instance that is already monitored	
	2	IOP Check	Verify that external VNFM receive the requests to create the monitoring threshold for the given VNF indicator	
	3	IOP Check	(if required and not implicitly carried out at step 1 and 2) Trigger MANO to subscribe on the external VNFM to the VNF indicator threshold crossing notification for the VNF indicator threshold created	
	4	Stimulus	Trigger the VNF indicator to cross the specified threshold (e.g. by accessing the related VNF instance and manually stimulate the VNF indicator change)	
	5	IOP Check	Verify that a "threshold crossed" notification for the VNF indicator was generated, received by the external VNFM (e.g. querying the related VNFM monitoring database) and properly dispatched to MANO	
	6	IOP Check	Verify that the "threshold crossed" notification for the VNF indicator was collected by MANO (e.g. querying the related MANO monitoring database) and properly processed (e.g. checking the MANO Graphical User Interface, if applicable)	
<b>IOP Verdict</b>				

## 6.5.4.2.4 TD\_NFV\_S-VNFM-D\_PM\_VNF\_KPI\_DELETE\_THRESHOLD\_001

Interoperability Test Description	
<b>Identifier</b>	TD_NFV_S-VNFM-D_PM_VNF_KPI_DELETE_THRESHOLD_001
<b>Test Purpose</b>	To verify that performance monitoring thresholds created for one or more VNF indicator related to a NS instance can be deleted through external VNFM
<b>Configuration</b>	SUT_S-VNFM-D
<b>References</b>	ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.4, 7.5.5, 7.5.8, 8.4)
<b>Applicability</b>	* MANO supports receiving or collecting VNF KPIs * VNF/EM can send indicators (KPIs) to MANO asynchronously or by explicit request

Interoperability Test Description				
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001) * A performance monitoring threshold for a VNF indicator related to a VNF instance allocated to the relevant NS instance is created by MANO through VNFM (TD_NFV_S-VNFM-D_PM_VNF_KPI_CREATE_THRESHOLD)			
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Trigger MANO to delete a monitoring threshold linked to a performance metric of a VNF indicator related to a VNF instance that is allocated to the target NS instance	
	2	IOP Check	Verify that the relevant VNF indicator performance monitoring threshold has been deleted on MANO (e.g. querying its monitoring service, if applicable)	
	3	IOP Check	In turn, verify that the VNFM receives a request from MANO for monitoring threshold deletion, and that VNFM deletes the related threshold (e.g. querying the VNFM monitoring service, if applicable)	
	4	Stimulus	Trigger the VNF indicator to cross the deleted threshold (e.g. by accessing the related VNF instance and manually stimulate the VNF indicator change)	
	5	IOP Check	Verify that VNFM is not receiving and therefore not processing (and dispatching) any VNF indicator threshold crossed notification, e.g. querying the related VNFM monitoring database	
	6	IOP Check	Verify that MANO is not receiving from VNFM any VNF indicator threshold crossed notification, e.g. querying the related MANO monitoring database or checking the MANO Graphical User Interface, if applicable	
<b>IOP Verdict</b>				

## 6.5.5 FAULT MANAGEMENT

### 6.5.5.1 FAULT MANAGEMENT – VNF VR

#### 6.5.5.1.1 TD\_NFV\_S-VNFM-D\_FM\_VNF\_VR\_ALARM\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_FM_VNF_VR_ALARM_001			
<b>Test Purpose</b>	Verify that a VNF fault alarm notification propagates via the specific VNFM to the MANO when a VNF fault is triggered by a failed virtualised resource			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA006 V2.3.1 (clauses 5.3.9, 7.6.3, 8.6.2) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.7, 6.4.6, 7.5.3, 8.4.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.5, 7.6.3, 8.5.2)			
<b>Applicability</b>	* [IFS_NFV_MANO_24] MANO supports receiving VNF faults/alarms from external specific VNFM * [IFS_NFV_VNFM_16] VNFM supports receiving VM/VNFC faults/alarms from VIM * [IFS_NFV_VNFM_19] VNFM exposes VNF faults/alarms towards MANO * [IFS_NFV_VIM_NFVI_6] NFVI/VIM exposes VM/VNFC faults/alarms to MANO/VNFM			
<b>Pre-test conditions</b>	* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001) * MANO is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) VNF fault alarms from the external VNFM * VNFM is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) virtualised resource fault alarms from the VIM			
	Step	Type	Description	Result

<b>Test Sequence</b>	1	Stimulus	Trigger a failure on a virtualised resource allocated to the relevant VNF instance (e.g. power off the 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 the virtualised resource alarm is collected or received by the VNFM and a correspondant VNF fault alarm is created for the affected VNF instance on the VNFM, e.g. by querying the VNF fault alarms database	
	4	IOP Check	Verify that the VNF fault alarm is propagated from the VNFM to the MANO and properly detected and processed by the MANO, e.g. by quering the NS fault alarms database or checking the GUI where applicable	
<b>IOP Verdict</b>				

## 6.5.5.1.2 TD\_NFV\_S-VNFM-D\_FM\_VNF\_VR\_CLEAR\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_FM_VNF_VR_CLEAR_001			
<b>Test Purpose</b>	Verify that a VNF fault alarm clearance notification propagates via the specific VNFM to the MANO when a VNF fault is cleared by resolving the causing problem on the failed virtualised resource			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA006 V2.3.1 (clauses 5.3.9, 7.6.3, 8.6.3) ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.7, 6.4.6, 7.5.3, 8.4.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.5, 7.6.3, 8.5.3)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_24] MANO supports receiving VNF faults/alarms from external VNFM</li> <li>* [IFS_NFV_VNFM_16] VNFM supports receiving VM/VNFC faults/alarms from VIM</li> <li>* [IFS_NFV_VNFM_19] VNFM exposes VNF faults/alarms towards MANO</li> <li>* [IFS_NFV_VIM_NFVI_6] NFVI/VIM exposes VM/VNFC faults/alarms to MANO/VNFM</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001)</li> <li>* MANO is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) VNF fault alarms from the external VNFM</li> <li>* VNFM is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) virtualised resource fault alarms from the VIM</li> <li>* NS fault alarm is created on the MANO by failing a virtualised resource allocated to a VNF that is required for the NS (TD_NFV_S-VNFM-D_FM_VNF_VR_ALARM)</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	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 virtualised resource fault clearance is detected by the VNFM and the relevant VNF fault alarm has been cleared on the VNFM, e.g. by querying the list of VNF fault alarms	
	4	IOP Check	Verify that the VNF fault clear event is detected by MANO, and the relevant NS fault alarm has been cleared, by querying the list of NS fault alarms or checking the GUI where applicable	
<b>IOP Verdict</b>				

## 6.5.5.2 FAULT MANAGEMENT – VNF

## 6.5.5.2.1 TD\_NFV\_S-VNFM-D\_FM\_VNF\_ALARM\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_FM_VNF_ALARM_001			
<b>Test Purpose</b>	Verify that a VNF fault alarm notification propagates via the VNFM to the MANO when a fault occurs on a VNF part of a NS			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.7, 6.4.6, 7.5.3, 8.4.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.5, 7.6.3, 8.5.2)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_24] MANO supports receiving VNF faults/alarms from external specific VNFM</li> <li>* VNFM supports receiving VNF/VNFC faults/alarms from VNF/EM</li> <li>* VNF/EM exposes faults/alarms to VNFM</li> <li>* [IFS_NFV_VNFM_19] VNFM exposes VNF faults/alarms towards MANO</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001)</li> <li>* MANO is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) VNF fault alarms from the external VNFM</li> <li>* VNFM is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) VNF fault alarms from the VNF/EM</li> </ul>			
<b>Test Sequence</b>	<b>Step</b>	<b>Type</b>	<b>Description</b>	<b>Result</b>
	1	Stimulus	Trigger a fault on a VNF that is part of the relevant NS instance (e.g. disabling the management connectivity between VNF and VNFM as an emulation of a VNF that is not alive any longer)	
	2	IOP Check	Verify that the VNF fault is detected and processed by the VNFM, e.g. by querying the database of VNF fault alarms	
	3	IOP Check	Verify that the VNF fault alarm is propagated from the VNFM to the MANO and properly detected and processed by the MANO, e.g. by querying the NS fault alarms database or checking the GUI where applicable	
<b>IOP Verdict</b>				

## 6.5.5.2.2 TD\_NFV\_S-VNFM-D\_FM\_VNF\_CLEAR\_001

Interoperability Test Description				
<b>Identifier</b>	TD_NFV_S-VNFM-D_FM_VNF_CLEAR_001			
<b>Test Purpose</b>	Verify that a VNF fault alarm clearance notification propagates via the VNFM to the MANO when a fault is cleared on a VNF part of a NS by resolving the causing problem			
<b>Configuration</b>	SUT_S-VNFM-D			
<b>References</b>	ETSI GS NFV-IFA007 V2.3.1 (clauses 5.3.7, 6.4.6, 7.5.3, 8.4.7) ETSI GS NFV-IFA013 V2.3.1 (clauses 5.3.5, 7.6.3, 8.5.3)			
<b>Applicability</b>	<ul style="list-style-type: none"> <li>* [IFS_NFV_MANO_24] MANO supports receiving VNF faults/alarms from external specific VNFM</li> <li>* VNFM supports receiving VNF/VNFC faults/alarms from VNF/EM</li> <li>* VNF/EM exposes faults/alarms to VNFM</li> <li>* [IFS_NFV_VNFM_19] VNFM exposes VNF faults/alarms towards MANO</li> </ul>			
<b>Pre-test conditions</b>	<ul style="list-style-type: none"> <li>* NS is instantiated (TD_NFV_S-VNFM-D_NS_LCM_INSTANTIATE_001)</li> <li>* MANO is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) VNF fault alarms from the external VNFM</li> <li>* VNFM is configured to collect (e.g. through a subscription) or receive (e.g. through explicit queries) VNF fault alarms from the VNF/EM</li> </ul>			

	* NS fault alarm is created on the MANO by failing a VNF that is required for the NS (TD_NFV_S-VNFM-D_FM_VNF_ALARM)			
Test Sequence	Step	Type	Description	Result
	1	Stimulus	Resolve the failure of the relevant VNF that triggered the alarm on the NS instance (e.g. re-enable the management connectivity between VNFM and the VNF)	
	3	IOP Check	Verify that the VNF fault clearance is detected by the VNFM and the relevant VNF fault alarm has been cleared on the VNFM, e.g. by querying the list of VNF fault alarms	
	4	IOP Check	Verify that the VNF fault clear event is detected by MANO, and the relevant NS fault alarm has been cleared, by querying the list of NS fault alarms or checking the GUI where applicable	
<b>IOP Verdict</b>				

## Annex A: Interoperability Feature Statements

### A.1 IFS for MANO

IFS_ID	Description	Support
IFS_NFV_MANO_1	MANO supports multi-site deployments (i.e. two or more geographically distributed sites managed by different VIM instances)	
IFS_NFV_MANO_3	MANO provides generic VNFM functionality	
IFS_NFV_MANO_4	MANO supports specific VNFMs (external) in direct mode (resource management by VNFM)	
IFS_NFV_MANO_5	MANO supports specific VNFMs (external) in in-direct mode (resource management by MANO)	
IFS_NFV_MANO_14	MANO supports scaling by adding/removing VNF instances	
IFS_NFV_MANO_15	MANO supports scaling out/in by adding/removing VNFC instances	
IFS_NFV_MANO_16	MANO supports scaling out/in request from VNF/EM	
IFS_NFV_MANO_17	MANO supports receiving VNF indicators from VNF/EM	
IFS_NFV_MANO_18	MANO supports automatic scaling triggered by VNF indicators from VNF/EM	
IFS_NFV_MANO_19	MANO supports receiving VM/VNFC KPIs from VIM	
IFS_NFV_MANO_20	MANO supports automatic scaling out/in triggered by KPIs from VIM	
IFS_NFV_MANO_21	MANO supports receiving VNF KPIs from external VNFMs	
IFS_NFV_MANO_23	MANO supports receiving VM/VNFC faults/alarms from VIM	
IFS_NFV_MANO_24	MANO supports receiving VNF faults/alarms from external VNFM	
IFS_NFV_MANO_28	MANO supports deploying VNFs with EPA requirements towards NFVI/VIM	

IFS_NFV_MANO_30	MANO supports provisioning and configuration of network forwarding paths	
IFS_NFV_MANO_31	MANO supports receiving VNF faults/alarms from VNF/EM	
IFS_NFV_MANO_32	MANO can request to start/stop VNFs/VNFCs to the VIM	
IFS_NFV_MANO_33	MANO supports NS scaling among levels by changing the number of VNF instances	
IFS_NFV_MANO_34	MANO supports VNF scaling among levels by changing the number of VNFC instances	

## A.2 IFS for VIM/NFVI

IFS_ID	Description	Support
IFS_NFV_VIM_NFVI_1	NFVI/VIM supports EPA attributes	
IFS_NFV_VIM_NFVI_2	NFVI/VIM supports multi-site deployment	
IFS_NFV_VIM_NFVI_3	NFVI/VIM exposes VM/VNFC virtual compute resource KPIs to MANO/VNFM	
IFS_NFV_VIM_NFVI_4	NFVI/VIM exposes VM/VNFC virtual network resource KPIs to MANO/VNFM	
IFS_NFV_VIM_NFVI_5	NFVI/VIM exposes VM/VNFC virtual storage resource KPIs to MANO/VNFM	
IFS_NFV_VIM_NFVI_6	NFVI/VIM exposes VM/VNFC faults/alarms to MANO/VNFM	
IFS_NFV_VIM_NFVI_9	NFVI/VIM exposes network forwarding path functionalities to MANO	
IFS_NFV_VIM_NFVI_10	NFVI/VIM supports start/stop of VMs/VNFCs	

## A.3 IFS for VNF

IFS_ID	Description	Support
IFS_NFV_VNF_1	VNF requires EPA	
IFS_NFV_VNF_2	VNF has its own VNFM	
IFS_NFV_VNF_3	VNF can work with generic VNFM	
IFS_NFV_VNF_4	VNF can scale out/in by adding/removing VNF instances	
IFS_NFV_VNF_5	VNF can scale out/in by adding/removing VNFC instances	
IFS_NFV_VNF_8	VNF/EM can request scaling to MANO	
IFS_NFV_VNF_9	VNF can send indicators (KPIs) to MANO	
IFS_NFV_VNF_10	VNF can be part of multi-vendor NS	

## A.4 IFS for VNFM

IFS_ID	Description	Support
IFS_NFV_VNFM_1	VNFM supports direct mode (Resource management by VNFM)	
IFS_NFV_VNFM_2	VNFM supports in-direct mode (Resource management by MANO)	
IFS_NFV_VNFM_9	(if direct mode) VNFM supports multi-VIM deployments (i.e. different VIMs and possibly different technologies, like OpenStack and VMWare)	
IFS_NFV_VNFM_10	VNFM supports VNF scaling out/in by adding/removing VNFC instances	
IFS_NFV_VNFM_11	VNFM supports VNF scaling out/in request from VNF/EM	
IFS_NFV_VNFM_12	VNFM supports receiving VNF indicators from VNF/EM	
IFS_NFV_VNFM_13	VNFM supports automatic VNF scaling triggered by VNF indicators from VNF/EM	
IFS_NFV_VNFM_14	VNFM supports receiving VM/VNFC KPIs from VIM	
IFS_NFV_VNFM_15	VNFM supports automatic scaling out/in triggered by KPIs from VIM	
IFS_NFV_VNFM_16	VNFM supports receiving VM/VNFC faults/alarms from VIM	
IFS_NFV_VNFM_17	VNFM supports VNF/VNFC healing triggered by faults/alarms from VIM	
IFS_NFV_VNFM_18	VNFM exposes VNF KPIs and indicators towards MANO	
IFS_NFV_VNFM_19	VNFM exposes VNF faults/alarms towards MANO	

---

## History

<b>Document history</b>		
V1.0.0	30/08/2018	Publication