



Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Os-Ma-Nfvo reference point - Interface and Information Model Specification

Disclaimer

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

Reference

RGS/NFV-IFA013ed231

Keywordsinterface, management, MANO, NFV,
orchestration, virtualisation**ETSI**

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

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

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

Important notice

The present document can be downloaded from:

<http://www.etsi.org/standards-search>

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

<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:

<https://portal.etsi.org/People/CommiteeSupportStaff.aspx>

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.

© ETSI 2017.
All rights reserved.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members.
3GPP™ and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

oneM2M logo is protected for the benefit of its Members.

GSM® and the GSM logo are trademarks registered and owned by the GSM Association.

Contents

Intellectual Property Rights	13
Foreword.....	13
Modal verbs terminology.....	13
1 Scope	14
2 References	14
2.1 Normative references	14
2.2 Informative references.....	14
3 Definitions and abbreviations.....	15
3.1 Definitions	15
3.2 Abbreviations	15
4 Overview of interfaces and information elements associated to the Os-Ma-nfvo reference point.....	15
4.1 Introduction	15
4.2 Relation to other NFV group specifications	16
4.3 Conventions.....	16
5 Reference point and interface requirements	17
5.1 Introduction	17
5.2 Os-Ma-nfvo reference point requirements.....	17
5.3 Interface requirements.....	17
5.3.1 NSD Management interface requirements	17
5.3.2 NS Lifecycle Management interface requirements.....	18
5.3.3 Void	20
5.3.4 NS Performance Management interface requirements.....	20
5.3.5 NS Fault Management interface requirements.....	21
5.3.6 VNF Package Management interface requirements.....	21
6 OSS exposed interfaces	22
7 NFVO exposed interfaces	22
7.1 Introduction	22
7.2 NSD Management interface	22
7.2.1 Description.....	22
7.2.2 On-board NSD operation	23
7.2.2.1 Description	23
7.2.2.2 Input parameters.....	23
7.2.2.3 Output parameters	23
7.2.2.4 Operation results	23
7.2.3 Enable NSD operation	24
7.2.3.1 Description	24
7.2.3.2 Input parameters.....	24
7.2.3.3 Output parameters.....	24
7.2.3.4 Operation results	24
7.2.4 Disable NSD operation	24
7.2.4.1 Description	24
7.2.4.2 Input parameters.....	25
7.2.4.3 Output parameters	25
7.2.4.4 Operation results	25
7.2.5 Update NSD operation.....	25
7.2.5.1 Description	25
7.2.5.2 Input parameters.....	25
7.2.5.3 Output parameters	26
7.2.5.4 Operation results	26
7.2.6 Delete NSD operation.....	26
7.2.6.1 Description	26
7.2.6.2 Input parameters.....	26

7.2.6.3	Output parameters	27
7.2.6.4	Operation results	27
7.2.7	Query NSD operation	27
7.2.7.1	Description	27
7.2.7.2	Input parameters	27
7.2.7.3	Output parameters	27
7.2.7.4	Operation results	28
7.2.8	On-board PNF operation	28
7.2.8.1	Description	28
7.2.8.2	Input parameters	28
7.2.8.3	Output parameters	28
7.2.8.4	Operation results	28
7.2.9	Update PNF operation	28
7.2.9.1	Description	28
7.2.9.2	Input parameters	29
7.2.9.3	Output parameters	29
7.2.9.4	Operation results	29
7.2.10	Delete PNF operation	29
7.2.10.1	Description	29
7.2.10.2	Input parameters	30
7.2.10.3	Output parameters	30
7.2.10.4	Operation results	30
7.2.11	Query PNF operation	30
7.2.11.1	Description	30
7.2.11.2	Input parameters	31
7.2.11.3	Output parameters	31
7.2.11.4	Operation results	31
7.2.12	Subscribe operation	31
7.2.12.1	Description	31
7.2.12.2	Input parameters	32
7.2.12.3	Output parameters	32
7.2.12.4	Operation results	32
7.2.13	Notify operation	32
7.2.13.1	Description	32
7.3	NS Lifecycle Management interface	33
7.3.1	Description	33
7.3.2	Create NS Identifier operation	33
7.3.2.1	Description	33
7.3.2.2	Input parameters	33
7.3.2.3	Output parameters	34
7.3.2.4	Operation results	34
7.3.3	Instantiate NS operation	34
7.3.3.1	Description	34
7.3.3.2	Input parameters	34
7.3.3.3	Output parameters	36
7.3.3.4	Operation results	36
7.3.4	Scale NS operation	36
7.3.4.1	Description	36
7.3.4.2	Input parameters	37
7.3.4.3	Output parameters	37
7.3.4.4	Operation results	37
7.3.5	Update NS operation	38
7.3.5.1	Introduction	38
7.3.5.2	Input parameters	39
7.3.5.3	Output parameters	41
7.3.5.4	Operation results	41
7.3.6	Query NS operation	41
7.3.6.1	Description	41
7.3.6.2	Input parameters	41
7.3.6.3	Output parameters	42
7.3.6.4	Operation results	42
7.3.7	Terminate NS operation	42

7.3.7.1	Description	42
7.3.7.2	Input parameters	43
7.3.7.3	Output parameters	43
7.3.7.4	Operation results	43
7.3.8	Delete NS Identifier operation	43
7.3.8.1	Description	43
7.3.8.2	Input parameters	44
7.3.8.3	Output parameters	44
7.3.8.4	Operation results	44
7.3.9	Heal NS operation	44
7.3.9.1	Description	44
7.3.9.2	Input parameters	45
7.3.9.3	Output parameters	45
7.3.9.4	Operation results	45
7.3.10	Get Operation Status operation	45
7.3.10.1	Description	45
7.3.10.2	Input parameters	46
7.3.10.3	Output Parameters	46
7.3.10.4	Operation results	46
7.3.11	Subscribe operation	46
7.3.11.1	Description	46
7.3.11.2	Input parameters	46
7.3.11.3	Output parameters	46
7.3.11.4	Operation results	47
7.3.12	Notify operation	47
7.3.12.1	Description	47
7.3.13	Terminate Subscription operation	47
7.3.13.1	Description	47
7.3.13.2	Input parameters	47
7.3.13.3	Output parameters	48
7.3.13.4	Operation results	48
7.3.14	Query Subscription Info operation	48
7.3.14.1	Description	48
7.3.14.2	Input parameters	48
7.3.14.3	Output parameters	48
7.3.14.4	Operation results	49
7.4	Void	49
7.5	NS Performance Management interface	49
7.5.1	Description	49
7.5.2	Create PM Job operation	49
7.5.2.1	Description	49
7.5.2.2	Input parameters	50
7.5.2.3	Output parameters	50
7.5.2.4	Operation results	50
7.5.3	Delete PM Jobs operation	51
7.5.3.1	Description	51
7.5.3.2	Input parameters	51
7.5.3.3	Output parameters	51
7.5.3.4	Operation results	51
7.5.4	Subscribe operation	51
7.5.4.1	Description	51
7.5.4.2	Input parameters	52
7.5.4.3	Output parameters	52
7.5.4.4	Operation results	52
7.5.5	Notify operation	52
7.5.5.1	Description	52
7.5.6	Query PM Job operation	52
7.5.6.1	Description	52
7.5.6.2	Input parameters	53
7.5.6.3	Output parameters	53
7.5.6.4	Operation results	53
7.5.7	Create Threshold operation	53

7.5.7.1	Description	53
7.5.7.2	Input parameters	53
7.5.7.3	Output parameters	54
7.5.7.4	Operation results	54
7.5.8	Delete Thresholds operation	54
7.5.8.1	Description	54
7.5.8.2	Input parameters	54
7.5.8.3	Output parameters	54
7.5.8.4	Operation results	55
7.5.9	Query Threshold operation	55
7.5.9.1	Description	55
7.5.9.2	Input parameters	55
7.4.9.3	Output parameters	55
7.5.9.4	Operation results	55
7.6	NS Fault Management interface	55
7.6.1	Description	55
7.6.2	Subscribe operation	56
7.6.2.1	Description	56
7.6.2.2	Input parameters	56
7.6.2.3	Output parameters	56
7.6.2.4	Operation results	56
7.6.3	Notify operation	56
7.6.3.1	Description	56
7.6.4	Get Alarm List operation	57
7.6.4.1	Description	57
7.6.4.2	Input parameters	57
7.6.4.3	Output parameters	57
7.6.4.4	Operation results	57
7.7	VNF Package management interface	58
7.7.1	Description	58
7.7.2	On-board VNF Package operation	58
7.7.2.1	Description	58
7.7.2.2	Input parameters	58
7.7.2.3	Output parameters	59
7.7.2.4	Operation results	59
7.7.3	Enable VNF Package operation	59
7.7.3.1	Description	59
7.7.3.2	Input parameters	59
7.7.3.3	Output parameters	59
7.7.3.4	Operation results	60
7.7.4	Disable VNF Package operation	60
7.7.4.1	Description	60
7.7.4.2	Input parameters	60
7.7.4.3	Output parameters	60
7.7.4.4	Operation results	60
7.7.5	Delete VNF Package operation	60
7.7.5.1	Description	60
7.7.5.2	Input parameters	61
7.7.5.3	Output parameters	61
7.7.5.4	Operation results	61
7.7.6	Query On-boarded VNF Package Information operation	61
7.7.6.1	Description	61
7.7.6.2	Input parameters	62
7.7.6.3	Output parameters	62
7.7.6.4	Operation results	62
7.7.7	Subscribe operation	62
7.7.7.1	Description	62
7.7.7.2	Input parameters	63
7.7.7.3	Output parameters	63
7.7.7.4	Operation results	63
7.7.8	Notify operation	63
7.7.8.1	Description	63

7.7.9	Void	64
7.7.10	Fetch On-boarded VNF Package operation	64
7.7.10.1	Description	64
7.7.10.2	Input parameters	64
7.7.10.3	Output parameters	64
7.7.10.4	Operation results	64
7.7.11	Fetch On-boarded VNF Package Artifacts operation	64
7.7.11.1	Description	64
7.7.11.2	Input parameters	65
7.7.11.3	Output parameters	65
7.7.11.4	Operation results	65
7.7.12	Abort VNF Package deletion operation	65
7.7.12.1	Description	65
7.7.12.2	Input parameters	65
7.7.12.3	Output parameters	66
7.7.12.4	Operation results	66
8	Information elements exchanged	66
8.1	Introduction	66
8.2	Information elements related to NSD Management	66
8.2.1	Introduction	66
8.2.2	NsdInfo information element	66
8.2.2.1	Description	66
8.2.2.2	Attributes	67
8.2.3	Pnfd information element	67
8.2.3.1	Description	67
8.2.3.2	Attributes	67
8.2.4	PnfdInfo information element	67
8.2.4.1	Description	67
8.2.4.2	Attributes	68
8.2.5	Nsd information element	68
8.2.5.1	Description	68
8.2.5.2	Attributes	68
8.2.6	NsdOnBoardingNotification	68
8.2.6.1	Description	68
8.2.6.2	Trigger Conditions	68
8.2.6.3	Attributes	69
8.2.7	NsdChangeNotification	69
8.2.7.1	Description	69
8.2.7.2	Trigger Conditions	69
8.2.7.3	Attributes	69
8.3	Information elements and notifications related to NS Lifecycle Management	69
8.3.1	Introduction	69
8.3.2	Information elements and notifications related to NS Lifecycle Change Notification	70
8.3.2.1	Introduction	70
8.3.2.2	NsLifecycleChangeNotification	70
8.3.2.2.1	Description	70
8.3.2.2.2	Trigger conditions	70
8.3.2.2.3	Attributes	70
8.3.2.3	AffectedVnf information element	71
8.3.2.3.1	Description	71
8.3.2.3.2	Attributes	71
8.3.2.4	AffectedPnf information element	71
8.3.2.4.1	Description	71
8.3.2.4.2	Attributes	72
8.3.2.5	AffectedVirtualLink information element	72
8.3.2.5.1	Description	72
8.3.2.5.2	Attributes	72
8.3.2.6	AffectedVnffg information element	72
8.3.2.6.1	Description	72
8.3.2.6.2	Attributes	72
8.3.2.7	AffectedNs information element	73

8.3.2.7.1	Description	73
8.3.2.7.2	Attributes	73
8.3.2.8	AffectedSap information element	73
8.3.2.8.1	Description	73
8.3.2.8.2	Attributes	73
8.3.2.9	NsIdentifierCreationNotification	73
8.3.2.9.1	Description	73
8.3.2.9.2	Trigger conditions	73
8.3.2.9.3	Attributes	73
8.3.2.10	NsIdentifierDeletionNotification	74
8.3.2.10.1	Description	74
8.3.2.10.2	Trigger conditions	74
8.3.2.10.3	Attributes	74
8.3.3	Information elements related to NsInfo	74
8.3.3.1	Introduction	74
8.3.3.2	NsInfo information element	74
8.3.3.2.1	Description	74
8.3.3.2.2	Attributes	74
8.3.3.3	VnfInfo information element	75
8.3.3.3.1	Description	75
8.3.3.3.2	Attributes	75
8.3.3.4	InstantiatedVnfInfo information element	77
8.3.3.4.1	Description	77
8.3.3.4.2	Attributes	77
8.3.3.5	VnfcResourceInfo information element	78
8.3.3.5.1	Description	78
8.3.3.5.2	Attributes	78
8.3.3.6	VnfVirtualLinkResourceInfo information element	78
8.3.3.6.1	Description	78
8.3.3.6.2	Attributes	78
8.3.3.7	VirtualStorageResourceInfo information element	79
8.3.3.7.1	Description	79
8.3.3.7.2	Attributes	79
8.3.3.8	ResourceHandle information element	79
8.3.3.8.1	Description	79
8.3.3.8.2	Attributes	79
8.3.3.9	PnfInfo information element	80
8.3.3.9.1	Description	80
8.3.3.9.2	Attributes	80
8.3.3.10	NsVirtualLinkInfo information element	80
8.3.3.10.1	Description	80
8.3.3.10.2	Attributes	80
8.3.3.11	NsLinkPort information element	81
8.3.3.11.1	Description	81
8.3.3.11.2	Attributes	81
8.3.3.12	SapInfo information element	81
8.3.3.12.1	Description	81
8.3.3.12.2	Attributes	81
8.3.3.13	VnffgInfo information element	82
8.3.3.13.1	Description	82
8.3.3.13.2	Attributes	82
8.3.3.14	PnfExtCpInfo information element	82
8.3.3.14.1	Description	82
8.3.3.14.2	Attributes	82
8.3.3.15	Nfp information element	83
8.3.3.15.1	Description	83
8.3.3.15.2	Attributes	83
8.3.3.16	NsScaleInfo information element	83
8.3.3.16.1	Description	83
8.3.3.16.2	Attributes	83
8.3.3.17	VnfExtCpInfo information element	83
8.3.3.17.1	Description	83

8.3.4.17.2	Attributes	84
8.3.3.18	ExtVirtualLinkInfo information element	84
8.3.3.18.1	Description	84
8.3.3.18.2	Attributes	84
8.3.3.19	ExtManagedVirtualLinkInfo information element	84
8.3.3.19.1	Description	84
8.3.3.19.2	Attributes	84
8.3.3.20	VnfLinkPort information element	85
8.3.3.20.1	Description	85
8.3.3.20.2	Attributes	85
8.3.3.21	ScaleInfo information element	85
8.3.3.21.1	Description	85
8.3.3.21.2	Attributes	85
8.3.3.22	ExtLinkPort information element	85
8.3.3.22.1	Description	85
8.3.3.22.2	Attributes	86
8.3.3.23	VnfcCpInfo information element	86
8.3.3.23.1	Description	86
8.3.3.23.2	Attributes	86
8.3.4	Information elements related to NS Lifecycle Management operations	86
8.3.4.1	Introduction	86
8.3.4.2	SapData information element	86
8.3.4.2.1	Description	86
8.3.4.2.2	Attributes	86
8.3.4.3	VnfInstanceData information element	87
8.3.4.3.1	Description	87
8.3.4.3.2	Attributes	87
8.3.4.4	VnfLocationConstraint information element	87
8.3.4.4.1	Description	87
8.3.4.4.2	Attributes	87
8.3.4.5	ParamsForVnf information element	88
8.3.4.5.1	Description	88
8.3.4.5.2	Attributes	88
8.3.4.6	ScaleNsData information element	88
8.3.4.6.1	Description	88
8.3.4.6.2	Attributes	88
8.3.4.7	ScaleNsByStepsData information element	89
8.3.4.7.1	Description	89
8.3.4.7.2	Attributes	89
8.3.4.8	ScaleNsToLevelData information element	89
8.3.4.8.1	Description	89
8.3.4.8.2	Attributes	90
8.3.4.9	ScaleVnfData information element	90
8.3.4.9.1	Description	90
8.3.4.9.2	Attributes	90
8.3.4.10	ScaleToLevelData information element	90
8.3.4.10.1	Description	90
8.3.4.10.2	Attributes	91
8.3.4.11	ScaleByStepData information element	91
8.3.4.11.1	Description	91
8.3.4.11.2	Attributes	91
8.3.4.12	InstantiateVnfData information element	92
8.3.4.12.1	Description	92
8.3.4.12.2	Attributes	92
8.3.4.13	ExtVirtualLinkData information element	93
8.3.4.13.1	Description	93
8.3.4.13.2	Attributes	93
8.3.4.14	VnfExtCpData information element	93
8.3.4.14.1	Description	93
8.3.4.14.2	Attributes	93
8.3.4.15	ChangeVnfFlavourData information element	94
8.3.4.15.1	Description	94

8.3.4.15.2	Attributes	94
8.3.4.16	OperateVnfData information element	94
8.3.4.16.1	Description	94
8.3.4.16.2	Attributes	94
8.3.4.17	ModifyVnfInfoData information element	95
8.3.4.17.1	Description	95
8.3.4.17.2	Attributes	95
8.3.4.18	Void.....	96
8.3.4.19	AssocNewNsdVersionData information element.....	96
8.3.4.19.1	Description	96
8.3.4.19.2	Attributes	96
8.3.4.20	MoveVnfInstanceData information element	96
8.3.4.20.1	Description	96
8.3.4.20.2	Attributes	97
8.3.4.21	AddVnffgData information element	97
8.3.4.21.1	Description	97
8.3.4.21.2	Attributes	97
8.3.4.22	UpdateVnffgData information element.....	97
8.3.4.22.1	Description	97
8.3.4.22.2	Attributes	97
8.3.4.23	NfpData information element.....	98
8.3.4.23.1	Description	98
8.3.4.23.2	Attributes	98
8.3.4.24	HealNsData information element.....	98
8.3.4.24.1	Description	98
8.3.4.24.2	Attributes	98
8.3.4.25	HealVnfData information element.....	99
8.3.4.25.1	Description	99
8.3.4.25.2	Attributes	99
8.3.4.26	AffinityOrAntiAffinityRule information element.....	99
8.3.4.26.1	Description	99
8.3.4.26.2	Attributes	100
8.3.4.27	ChangeNsFlavourData information element.....	100
8.3.4.27.1	Description	100
8.3.4.27.2	Attributes	100
8.3.4.28	ExtManagedVirtualLinkData information element.....	100
8.3.4.28.1	Description	100
8.3.4.28.2	Attributes	100
8.3.4.29	ChangeExtVnfConnectivityData information element	101
8.3.4.29.1	Description	101
8.3.4.29.2	Attributes	101
8.4	Information elements and notifications related to NS Performance Management	102
8.4.1	Introduction.....	102
8.4.2	ObjectSelection information element.....	102
8.4.2.1	Description	102
8.4.2.2	Attributes.....	102
8.4.3	PmJob information element	102
8.4.3.1	Description	102
8.4.3.2	Attributes.....	103
8.4.4	Threshold information element.....	103
8.4.4.1	Description	103
8.4.4.2	Attributes.....	103
8.4.5	PerformanceReport information element.....	104
8.4.5.1	Description	104
8.4.5.2	Attributes.....	104
8.4.6	PerformanceReportEntry information element.....	104
8.4.6.1	Description	104
8.4.6.2	Attributes.....	104
8.4.7	PerformanceValueEntry information element	105
8.4.7.1	Description	105
8.4.7.2	Attributes.....	105
8.4.8	PerformanceInformationAvailableNotification	105

8.4.8.1	Description	105
8.4.8.2	Trigger Conditions	105
8.4.8.3	Attributes	105
8.4.9	ThresholdCrossedNotification	106
8.4.9.1	Description	106
8.4.9.2	Trigger Conditions	106
8.4.9.3	Attributes	106
8.5	Information elements and notifications NS Fault management	106
8.5.1	Introduction	106
8.5.2	AlarmNotification	107
8.5.2.1	Description	107
8.5.2.2	Trigger conditions	107
8.5.2.3	Attributes	107
8.5.3	AlarmClearedNotification	107
8.5.3.1	Description	107
8.5.3.2	Trigger conditions	107
8.5.3.3	Attributes	107
8.5.4	Alarm information element	108
8.5.4.1	Description	108
8.5.4.2	Attributes	108
8.6	Information elements and notifications related to VNF Package	108
8.6.1	Introduction	108
8.6.2	OnboardedVnfPkgInfo information element	109
8.6.2.1	Description	109
8.6.2.2	Attributes	109
8.6.3	Vnfd information element	110
8.6.3.1	Description	110
8.6.3.2	Attributes	110
8.6.4	VnfPackageSoftwareImageInformation information element	110
8.6.4.1	Description	110
8.6.4.2	Attributes	110
8.6.5	SoftwareImageInformation information element	110
8.6.5.1	Description	110
8.6.5.2	Attributes	110
8.6.6	VnfPackageArtifactInformation information element	111
8.6.6.1	Description	111
8.6.6.2	Attributes	111
8.6.7	Void	111
8.6.8	VnfPackageOnBoardingNotification	111
8.6.8.1	Description	111
8.6.8.2	Trigger Conditions	112
8.6.8.3	Attributes	112
8.6.9	VnfPackageChangeNotification	112
8.6.9.1	Description	112
8.6.9.2	Trigger Conditions	112
8.6.9.3	Attributes	112

Annex A (informative):	Principles related to VNF lifecycle management and NS lifecycle management	114
-------------------------------	---	------------

Annex B (informative):	Use cases for VNF reuse and referencing in NSs	115
-------------------------------	---	------------

B.1	Re-use of VNFs from a terminated NS	115
B.2	Creation of VNF instances in anticipation of future NS demand	115
B.3	Bottom-up NS instantiation	116
B.4	Shared VNF instances	116

Annex C (informative):	Message flows for supporting use cases with fine grained NS lifecycle management	117
-------------------------------	---	------------

C.1	Introduction	117
-----	--------------------	-----

C.2	New NS with VNF pools.....	117
C.3	New NS utilizing VNF instances from a VNF pool NS.....	119
C.4	Terminating NS instance with retained VNF instances	120
Annex D (informative): State models.....		122
D.1	VNF Package state model	122
D.1.1	Introduction	122
D.1.2	State model.....	122
D.2	NSD state model.....	122
D.2.1	Introduction	122
D.2.2	State model.....	123
D.3	NS state model	123
D.3.1	Introduction	123
D.3.2	State model.....	123
Annex E (informative): NS scaling		125
E.1	Forms of NS scaling.....	125
E.2	NS scaling triggers	125
E.2.1	NS auto-scale.....	125
E.2.2	NS scale triggered by OSS/BSS	125
E.3	Relation to NS DF	126
E.4	Input and tools for NS auto-scaling.....	126
E.4.1	Monitoring parameter.....	126
E.4.2	VNF indicator.....	126
E.4.3	Auto-scale policies/rules	126
Annex F (informative): Authors & contributors.....		127
Annex G (informative): Change History		129
History		130

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<https://ipr.etsi.org/>).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

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

Modal verbs terminology

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

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

1 Scope

The present document defines the interfaces supported over the Os-Ma-nfvo reference point of the NFV-MANO architectural framework [i.2] as well as the information elements exchanged over those interfaces.

Applications and end-to-end services on top of network services are out of scope of the present document and are addressed in Application and Service Management Interface and Information Model Specification [i.7].

2 References

2.1 Normative references

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

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

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

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

- [1] ETSI GS NFV-IFA 010: "Network Functions Virtualisation (NFV); Management and Orchestration; Functional Requirements Specification".
- [2] ETSI GS NFV-IFA 011: "Network Functions Virtualisation (NFV); Management and Orchestration; VNF Packaging Specification".
- [3] ETSI GS NFV-IFA 014: "Network Functions Virtualisation (NFV); Management and Orchestration; Network Service Templates Specification".

2.2 Informative references

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

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

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

- [i.1] ETSI GS NFV 003: "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".
- [i.2] ETSI GS NFV-MAN 001: "Network Functions Virtualisation (NFV); Management and Orchestration".
- [i.3] ISO/IEC 9646-7: "Information technology - Open Systems Interconnection - Conformance testing methodology and framework - Part 7: Implementation Conformance Statements".
- [i.4] ETSI GS NFV-IFA 005: "Network Functions Virtualisation (NFV); Management and Orchestration; Or-Vi reference point - Interface and Information Model Specification".
- [i.5] ETSI GS NFV-IFA 007: "Network Functions Virtualisation (NFV); Management and Orchestration; Or-Vnfm reference point - Interface and Information Model Specification".

- [i.6] ETSI GS NFV-IFA 009: "Network Functions Virtualisation (NFV); Management and Orchestration; Report on Architectural Options".
- [i.7] ETSI GS NFV-IFA 012: " Network Functions Virtualization (NFV) Release3; Management and Orchestration; Os-Ma-Nfvo reference point - Application and Service Management Interface and Information Model Specification".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in ETSI GS NFV 003 [i.1].

NOTE: A term defined in the present document takes precedence over the definition of the same term, if any, in ETSI GS NFV 003 [i.1].

3.2 Abbreviations

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

NOTE: An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in ETSI GS NFV 003 [i.1].

CP	Connection Point
CPD	Connection Point Descriptor
DF	Deployment Flavour
FB	Functional Block
NFP	Network Forwarding Path
NSD	Network Service Descriptor
PNFD	Physical Network Function Descriptor
SAP	Service Access Point
SAPD	Service Access Point Descriptor
VL	Virtual Link
VLD	Virtual Link Descriptor
VNFFG	VNF Forwarding Graph
VNFFGD	VNF Forwarding Graph Descriptor

4 Overview of interfaces and information elements associated to the Os-Ma-nfvo reference point

4.1 Introduction

This clause provides an overview of interfaces and information models associated to the Os-Ma-nfvo reference point.

The Os-Ma-nfvo reference point is used for exchanges between the OSS/BSS and the NFV Orchestrator (NFVO), and supports the following interfaces:

- Network Service Descriptor (NSD) Management
- Network Service (NS) Lifecycle Management
- NS Performance Management
- NS Fault Management
- VNF Package Management

All the interfaces above are produced by the NFVO and consumed by the OSS/BSS.

The present document does not specify any interface produced by the OSS/BSS.

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

4.2 Relation to other NFV group specifications

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

- Management and Orchestration - Report on Architectural Options ETSI GS NFV-IFA 009 [i.6]:
 - ETSI GS NFV-IFA 009 [i.6] provides architectural options that can influence the way some of the Os-Ma-nfvo interfaces are used or might even suggest the need for extension.
- Management and Orchestration - Functional requirements specification ETSI GS NFV-IFA 010 [1]:
 - Interfaces associated with the Os-Ma-nfvo reference point are based on the functional requirements specified in ETSI GS NFV-IFA 010 [1] for the NFVO FB.
- Management and Orchestration - Os-Ma-Nfvo reference point - Application and Service Management Interface and Information Model Specification ETSI GS NFV-IFA 012 [i.7]:
 - ETSI GS NFV-IFA 012 [i.7] covers the Os-Ma-nfvo reference point, handling the application and end-to-end services on top of network services. As such, ETSI GS NFV-IFA 012 [i.7] might directly impact NS interfaces described in the present document.
- Management and Orchestration - Or-Vnfm reference point - Interface and Information Model Specification ETSI GS NFV-IFA 007 [i.5].

4.3 Conventions

The following notations, defined in ISO/IEC 9646-7 [i.3], are used for the qualifier column of interface information elements:

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

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

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

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

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

EXAMPLE 2: "Identifier (Reference to Vnfc)" or "Identifier (Reference to Vnfc, Virtual Link (VL) or VirtualStorage)".

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

5 Reference point and interface requirements

5.1 Introduction

This clause defines requirements applicable to interfaces in the specific context of the Os-Ma-nfvo reference point.

5.2 Os-Ma-nfvo reference point requirements

Table 5.2-1 specifies requirements applicable to the Os-Ma-nfvo reference point.

Table 5.2-1: Os-Ma-nfvo reference point requirements

Numbering	Functional requirement description
Os-Ma-nfvo.001	The Os-Ma-nfvo reference point shall support the NSD Management interface produced by the NFVO.
Os-Ma-nfvo.002	The Os-Ma-nfvo reference point shall support the NS Lifecycle Management interface produced by the NFVO.
Os-Ma-nfvo.003	The Os-Ma-nfvo reference point shall support the NS Lifecycle Change Notifications interface produced by the NFVO.
Os-Ma-nfvo.004	The Os-Ma-nfvo reference point shall support the NS Performance Management interface produced by the NFVO.
Os-Ma-nfvo.005	The Os-Ma-nfvo reference point shall support the NS Fault Management interface produced by the NFVO.
Os-Ma-nfvo.006	The Os-Ma-nfvo reference point shall support the VNF Package Management interface produced by the NFVO.
Os-Ma-nfvo.007	Any interaction on the Os-Ma-Nfvo reference point concerning a VNF shall be associated with at least one NS instance.

5.3 Interface requirements

5.3.1 NSD Management interface requirements

Table 5.3.1-1 specifies requirements applicable to the NSD management interface produced by the NFVO on the Os-Ma-nfvo reference point.

Table 5.3.1-1: NSD management interface requirements

Numbering	Functional requirement description
Os-Ma-nfvo.Nsd.001	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support on-boarding NSD.
Os-Ma-nfvo.Nsd.002	The NSDManagement interface produced by the NFVO on the Os-Ma-nfvo reference point shall support disabling an NSD.
Os-Ma-nfvo.Nsd.003	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support enabling an NSD.
Os-Ma-nfvo.Nsd.004	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support updating an NSD.
Os-Ma-nfvo.Nsd.005	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying NSDs.
Os-Ma-nfvo.Nsd.006	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting an NSD.
Os-Ma-nfvo.Nsd.007	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications about the on-boarding of NSDs.

Numbering	Functional requirement description
Os-Ma-nfvo.Nsd.008	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications as a result of changes on NSD states.
Os-Ma-nfvo.Nsd.009	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support on-boarding Physical Network Function Descriptor (PNFD).
Os-Ma-nfvo.Nsd.010	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support updating a PNFD.
Os-Ma-nfvo.Nsd.011	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting a PNFD.
Os-Ma-nfvo.Nsd.012	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying PNFDs.
Os-Ma-nfvo.Nsd.013	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support subscribing to notifications related to NSD management changes.

5.3.2 NS Lifecycle Management interface requirements

Table 5.3.2-1 specifies requirements applicable to the network service lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point.

Table 5.3.2-1: Network service lifecycle management interface requirements

Numbering	Functional requirement description
Os-Ma-nfvo.NsLcm.001	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support instantiating an NS.
Os-Ma-nfvo.NsLcm.002	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support terminating an NS.
Os-Ma-nfvo.NsLcm.003	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying NSs.
Os-Ma-nfvo.NsLcm.004	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support scaling an NS.
Os-Ma-nfvo.NsLcm.005	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support updating an NS.
Os-Ma-nfvo.NsLcm.006	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support creating a classification and selection rule for the existing Network Forwarding Path (NFP) instance.
Os-Ma-nfvo.NsLcm.007	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support updating the classification and selection rule for the existing NFP instance.
Os-Ma-nfvo.NsLcm.008	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support instantiating an NS which includes existing VNF instance(s). See notes 1 and 3.
Os-Ma-nfvo.NsLcm.009	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support updating an NS which includes existing VNF instance(s). See notes 2 and 3.
Os-Ma-nfvo.NsLcm.010	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support healing an NS.
Os-Ma-nfvo.NsLcm.011	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support sharing a VNF instance or a nested NS instance between multiple NS instances. See note 3.
Os-Ma-nfvo.NsLcm.012	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support instantiating a VNF instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.013	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support adding/removing an existing VNF instance to/from an NS instance as part of the update of an NS. See note 4.
Os-Ma-nfvo.NsLcm.014	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support scaling a VNF instance explicitly as part of the scaling of an NS.
Os-Ma-nfvo.NsLcm.015	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying information about a VNF instance as part of the query of an NS.
Os-Ma-nfvo.NsLcm.016	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support healing a VNF instance explicitly as part of the healing of an NS.
Os-Ma-nfvo.NsLcm.017	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support changing the state of a VNF instance explicitly as part of the update of an NS. See note 5.
Os-Ma-nfvo.NsLcm.018	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support changing the deployment flavour (DF) of a VNF instance explicitly as part of the update of an NS.

Numbering	Functional requirement description
Os-Ma-nfvo.NsLcm.019	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support modifying information and/or the configuration parameters of a VNF instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.036	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support changing the external connectivity of a VNF instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.021	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing additional affinity or anti-affinity rules when instantiating an NS.
Os-Ma-nfvo.NsLcm.035	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support creating a NS instance identifier and the associated instance of an NS information element.
Os-Ma-nfvo.NsLcm.022	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting a NS instance identifier and the associated instance of an NS information element.
Os-Ma-nfvo.NsLcm.023	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support adding/removing an existing nested NS instance to/from an NS instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.024	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support adding new Service Access Point (SAP) to an NS and removing existing SAP from a NS explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.025	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support associating a new NSD version to an existing NS instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.026	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support moving existing VNF instance(s) from one NS instance (source) to another NS instance (destination) explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.027	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support adding a new VNFFG to an NS instance, remove existing VNF Forwarding Graph (VNFFG) and updating a VNFFG from an NS instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.028	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying the status of an NS lifecycle management operation.
Os-Ma-nfvo.NsLcm.029	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing to the OSS/BSS notifications about changes of an NS instance that are related to NS lifecycle management operations.
Os-Ma-nfvo.NsLcm.030	Notifications provided on the NS lifecycle interface produced by the NFVO on the Os-Ma-nfvo reference point shall contain information about the type of the NS lifecycle change, the addition/deletion/modification of VNFs and/or Physical Network Functions (PNFs), about change in the connectivity between elements of the NS. See note 7.
Os-Ma-nfvo.NsLcm.031	Notifications provided on the NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall contain information about the VLs and VNFFGs that are added/modified/deleted as part of the NS lifecycle operation. See note 6.
Os-Ma-nfvo.NsLcm.032	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support notifying the result (successful or failed) of NS instantiation with indicating the NS instance Id.
Or-Ma-nfvo.NsLcm.033	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing to the OSS/BSS notifications about creation and deletion of an NS instance identifier and the associated instance of an NS information element, further referred to as NS identifier creation/deletion notifications.
Or-Ma-nfvo.NsLcm.034	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support subscribing to NS lifecycle change notifications and to NS identifier creation/deletion notifications.
Or-Ma-nfvo.NsLcm.037	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the capability to invoke NS error handling operation(s) after the NS life cycle operation occurrence fails. See note 8 and note 9.

Numbering	Functional requirement description
NOTE 1:	The existing VNF instance(s) may need to be modified as part of NS instantiation.
NOTE 2:	The existing VNF instance(s) may need to be modified as part of NS update.
NOTE 3:	A VNF instance or a nested NS instance can be shared between NS instances managed by the same NFVO.
NOTE 4:	If the VNF instance being removed is no longer part of any NS instance, it will be terminated.
NOTE 5:	Changing the state of a VNF instance refers to starting or stopping a VNF instance. These operations are complementary to instantiating or terminating a VNF.
NOTE 6:	This provides information about VLs and VNFFGs points used by the NS and whose creation was triggered by the NFVO.
NOTE 7:	The modification of VNFs includes the operations of VNF scaling, change of VNF flavours, VNF healing, change of VNF operational state, modification of VNF information data, and/or VNF configuration parameters and the change of VNF external connectivity.
NOTE 8:	It is up to the protocol design stage to design the detail error handling operation(s).
NOTE 9:	It depends on the NS capabilities whether and how the operation(s) are supported by a particular NS.

5.3.3 Void

5.3.4 NS Performance Management interface requirements

Table 5.3.4-1 specifies requirements applicable to the network service performance management interface produced by the NFVO on the Os-Ma-nfvo reference point.

Table 5.3.4-1: Network service performance management interface requirements

Numbering	Requirements description
Os-Ma-Nfvo.NsPm.001	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to control the collection and reporting of performance information for NSs.
Os-Ma-Nfvo.NsPm.002	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the capability to notify the availability of performance information. See note 1.
Os-Ma-Nfvo.NsPm.003	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall expose the type of performance information that the NFVO can collect from the NSs.
Os-Ma-Nfvo.NsPm.004	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to create a PM job specifying the type of resource(s) and performance information that the OSS/BSS requires.
Os-Ma-Nfvo.NsPm.005	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to create a PM job specifying the granularity for collection and reporting of performance information on NSs.
Os-Ma-Nfvo.NsPm.006	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to delete one or more explicitly identified PM job(s).
Os-Ma-Nfvo.NsPm.007	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support periodic collection of performance information (bounded or unbounded).
Os-Ma-Nfvo.NsPm.008	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference shall support the grouping of measurements. See note 2.
Os-Ma-Nfvo.NsPm.009	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to manage the thresholds on the performance information collected by the NFVO for NSs. See note 3.
Os-Ma-Nfvo.NsPm.010	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the capability to notify about a threshold defined for a specified metric of NSs being crossed.
Os-Ma-Nfvo.NsPm.011	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to receive notifications related to threshold crossing.
Os-Ma-Nfvo.NsPm.012	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point should support querying the list of active PM jobs and defined threshold conditions by the consumer entity that created them.

Numbering	Requirements description
Os-Ma-Nfvo.NsPm.013	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the deletion of threshold conditions on the performance information collected by the NFVO for NSs.
NOTE 1:	Performance information on a given NS results from either collected performance information of the virtualised resources impacting the connectivity of this NS instance or VNF performance information issued by the VNFM for the VNFs that is part of this NS instance. The latter performance information also results from collected performance information of the virtualised resources.
NOTE 2:	The group does not imply any modification/aggregation of performance measurements data and may be viewed as an alias for a pre-defined list of measurements. The group can be created by VNF, by NS, by virtual machine, etc.
NOTE 3:	Management of thresholds includes creation, deletion and query of the thresholds on the performance information collected.

5.3.5 NS Fault Management interface requirements

Table 5.3.5-1 specifies requirements applicable to the network service fault management interface produced by the NFVO on the Os-Ma-nfvo reference point.

Table 5.3.5-1: Network service fault management interface requirements

Numbering	Requirements description
Os-Ma-nfvo.NsFm.001	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support collecting NSs fault information. See note.
Os-Ma-nfvo.NsFm.002	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing alarm notifications related to faults on NSs to the OSS/BSS.
Os-Ma-nfvo.NsFm.003	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing a notification when there is a change in the alarm information on NS.
Os-Ma-nfvo.NsFm.004	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the sending of notification to the OSS/BSS when an alarm on an NS has been created.
Os-Ma-nfvo.NsFm.005	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the sending of notification to the OSS/BSS when an alarm on an NS has been cleared.
Os-Ma-nfvo.NsFm.006	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall allow unambiguous identification of the alarm on an NS sent to the OSS/BSS.
Os-Ma-nfvo.NsFm.007	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall allow unambiguous identification of the NS causing the alarm.
Os-Ma-nfvo.NsFm.008	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall allow unambiguous identification of the alarm cause.
NOTE:	Fault information on a given NS instance can include the information related to the alarm (e.g. alarm created, alarm cleared, etc.), alarm cause(s) and identification of this NS instance and fault information concerning the virtualised resources supporting the constituent VNFs for this NS instance and the virtualised resources supporting the connectivity of this NS instance.

5.3.6 VNF Package Management interface requirements

Table 5.3.6-1 specifies requirements applicable to the VNF Package management interface produced by the NFVO on the Os-Ma-nfvo reference point.

Table 5.3.6-1: VNF Package management interface requirements

Numbering	Functional requirement description
Os-Ma-nfvo.VnfPkgm.001	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support on-boarding a VNF Package.
Os-Ma-nfvo.VnfPkgm.002	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support disabling a VNF Package.
Os-Ma-nfvo.VnfPkgm.003	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support enabling a VNF Package.

Numbering	Functional requirement description
Os-Ma-nfvo.VnfPkgm.004	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying VNF Package information. See note.
Os-Ma-nfvo.VnfPkgm.005	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting a VNF Package.
Os-Ma-nfvo.VnfPkgm.006	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications about the on-boarding of VNF Packages.
Os-Ma-nfvo.VnfPkgm.007	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications as a result of changes on VNF Package states.
Os-Ma-nfvo.VnfPkgm.008	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support fetching a VNF Package, or selected artifacts contained in a package.
Os-Ma-nfvo.VnfPkgm.009	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support aborting the pending deletion of a VNF Package as long as a VNF instance exists that is based on this package.
NOTE:	VNF Package information can include information such as release date, vendor info, manifest, VNFD, SW image meta-data, files contained in the VNF Package, etc.

6 OSS exposed interfaces

No interface is exposed by the OSS/BSS FB to NFVO over the Os-Ma-nfvo reference point.

7 NFVO exposed interfaces

7.1 Introduction

This clause defines the interfaces exposed by the NFVO towards the OSS/BSS over the Os-Ma-nfvo reference point.

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

7.2 NSD Management interface

7.2.1 Description

This interface allows the management of NSDs and associated PNFDs. Virtual Link Descriptors (VLDs) and VNF Forwarding Graph Descriptors (VNFFGDs) are considered as part of the NSD and handled along with it.

The following operations are defined for this interface:

- On-board NSD;
- Enable NSD;
- Disable NSD;
- Update NSD;
- Delete NSD;
- Query NSD;
- On-board PNFD;

- Update PNFD;
- Delete PNFD;
- Query PNFD;
- Subscribe, for subscribing to notifications related to NSD management changes;
- Notify, for delivering notifications related to NSD management changes.

7.2.2 On-board NSD operation

7.2.2.1 Description

This operation will on-board an NSD in the NFVO.

Associated descriptors (VLD and VNFFGD), that are part of the NSD, are on-boarded at the same time.

All descriptors needed by the NSD: VNFD, PNFD and NSD for nested NSs shall be on-boarded before being able to successfully on-board the NSD.

Table 7.2.2.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.2.1-1: On-board NSD operation

Message	Requirement	Direction
OnboardNsdRequest	Mandatory	OSS/BSS → NFVO
OnboardNsdResponse	Mandatory	NFVO → OSS/BSS

7.2.2.2 Input parameters

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

Table 7.2.2.2-1: On-board NSD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsd	M	1	Nsd	NSD to be on-boarded.
userDefinedData	O	0..N	KeyValuePair	User defined data for the NSD.

7.2.2.3 Output parameters

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

Table 7.2.2.3-1: On-board NSD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier	Identifier of the on-boarded instance of the NSD.

7.2.2.4 Operation results

The result of the operation indicates whether the on-boarding of the NSD has been successful or not with a standard success/error result.

The nsdInfoId is only returned when the operations has been successful.

Once on-boarded, the NSD and all referenced descriptors are known to and validated by the NFVO. It is in "Enabled, Not in use" state, allowing its use for NS lifecycle management. See the NSD state model in clause D.2.

7.2.3 Enable NSD operation

7.2.3.1 Description

This operation will enable a previously disabled NSD instance, allowing again its use for instantiation of new network service with this descriptor. The "In use/Not in use" sub-state shall not change as a result of the operation.

Table 7.2.3.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.3.1-1: Enable NSD operation

Message	Requirement	Direction
EnableNsdRequest	Mandatory	OSS/BSS → NFVO
EnableNsdResponse	Mandatory	NFVO → OSS/BSS

7.2.3.2 Input parameters

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

Table 7.2.3.2-1: Enable NSD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfold	M	1	Identifier	Identifier of the on-boarded instance of the NSD.

7.2.3.3 Output parameters

No output parameter.

7.2.3.4 Operation results

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

If the NSD was already enabled or is in deletion pending state, this operation returns an error.

7.2.4 Disable NSD operation

7.2.4.1 Description

This operation will disable a previously enabled NSD instance, preventing any further use for instantiation of new network service with this descriptor. The "In use/Not in use" sub-state shall not change as a result of the operation.

Table 7.2.4.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.4.1-1: Disable NSD operation

Message	Requirement	Direction
DisableNsdRequest	Mandatory	OSS/BSS → NFVO
DisableNsdResponse	Mandatory	NFVO → OSS/BSS

7.2.4.2 Input parameters

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

Table 7.2.4.2-1: Disable NSD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfold	M	1	Identifier	Identifier of the on-boarded instance of the NSD.

7.2.4.3 Output parameters

No output parameter.

7.2.4.4 Operation results

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

If the NSD was already disabled or is in deletion pending state, this operation returns an error.

7.2.5 Update NSD operation

7.2.5.1 Description

This operation will update an already on-boarded NSD, creating a new version of the NSD. The operation can also be used to update the userDefinedData of an existing NsdInfo information element without creating a new version of the NSD.

The previous versions of the NSDs are not modified.

It is possible to add (remove) constituent descriptors (i.e. VNFDs, PNFDs, nested NSDs, VLDs, VNFFGDs and Service Access Point Descriptors (SAPDs)) to (from) an NSD via the Update NSD operation. This is done by changing the various descriptor references in the new NSD. For example, to add VNFDs to an NSD, the OSS/BSS adds corresponding VNFD identifiers to the list of vnfIds in the new NSD. To remove VNFDs, the OSS/BSS simply does not include the vnfIds (of the VNFDs to be removed) in the new NSD.

Table 7.2.5.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.5.1-1: Update NSD operation

Message	Requirement	Direction
UpdateNsdRequest	Mandatory	OSS/BSS → NFVO
UpdateNsdResponse	Mandatory	NFVO → OSS/BSS

7.2.5.2 Input parameters

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

Table 7.2.5.2-1: Update NSD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfold	M	1	Identifier	Identifier of the on-boarded version of the NSD.
nsd	M	0..1	Nsd	New NSD to be created. Only present if the NSD itself is updated. See note.
userDefinedData	O	0..N	KeyValuePair	User defined data to be updated. For existing keys, the value is replaced. See note.
NOTE:	At least one of the two parameters shall be present. If nsd is not present, the operation is used only to update existing or add additional user defined data using the userDefinedData parameter.			

7.2.5.3 Output parameters

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

Table 7.2.5.3-1: Update NSD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfold	M	1	Identifier	Identifier of the updated NSD.

7.2.5.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result. The nsdInfold is only returned when the operations has been successful.

If the NSD was not already on-boarded or is in deletion pending state, this operation returns an error.

7.2.6 Delete NSD operation

7.2.6.1 Description

This operation will delete one or more NSD(s).

It is possible to delete only a single version of an NSD or all versions.

An NSD can only be deleted when there is no instantiated NS using it.

An NSD in the deletion pending state can no longer be enabled, disabled or updated. It is not possible to instantiate NS(s) using an NSD in the deletion pending state.

Table 7.2.6.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.6.1-1: Delete NSD operation

Message	Requirement	Direction
DeleteNsdRequest	Mandatory	OSS/BSS → NFVO
DeleteNsdResponse	Mandatory	NFVO → OSS/BSS

7.2.6.2 Input parameters

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

Table 7.2.6.2-1: Delete NSD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfold	M	1..N	Identifier	Identifier of the on-boarded instance(s) of the NSD to be deleted.
applyOnAllVersions	O	0..1	Boolean	Indicates if the delete operation is to be applied on all versions of this NSD. By default, if not present, it applies only on the current version.

7.2.6.3 Output parameters

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

Table 7.2.6.3-1: Delete NSD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deletedNsdInfoId	M	1..N	Identifier	Identifier of the deleted NSD version(s).

7.2.6.4 Operation results

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

If one of the NSDs is still in use and is marked as delete pending, this is indicated as part of the operation result and it will be deleted once all associated and instantiated NSs are terminated.

7.2.7 Query NSD operation

7.2.7.1 Description

This operation will enable the OSS/BSS to query the NFVO concerning details of one or more NSDs.

Table 7.2.7.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.7.1-1: Query NSD operation

Message	Requirement	Direction
QueryNsdRequest	Mandatory	OSS/BSS → NFVO
QueryNsdResponse	Mandatory	NFVO → OSS/BSS

7.2.7.2 Input parameters

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

Table 7.2.7.2-1: Query NSD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filter defining the NSDs on which the query applies, based on attributes of the NSDs. It can also be used to specify one or more NSDs to be queried by providing their identifiers.
attributeSelector	M	0..N	String	Provides a list of attribute names of the NSD. If present, only these attributes are returned for the instances of NSD(s) matching the filter. If absent, the complete NSD instances are returned.

7.2.7.3 Output parameters

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

Table 7.2.7.3-1: Query NSD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	NsdInfo	Details of the on-boarded NSD(s) matching the input filter.

7.2.7.4 Operation results

After success operation, the NFVO has queried the internal NSD information objects. The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular query, information about the NSDs that the consumer has access to and that match the filter is returned.

7.2.8 On-board PNFD operation

7.2.8.1 Description

This operation will on-board a PNF in the NFVO, making it available to be used by NSDs.

Table 7.2.8.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.8.1-1: On-board PNFD operation

Message	Requirement	Direction
OnboardPnfdRequest	Mandatory	OSS/BSS → NFVO
OnboardPnfdResponse	Mandatory	NFVO → OSS/BSS

7.2.8.2 Input parameters

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

Table 7.2.8.2-1: On-board PNFD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfd	M	1	Pnfd	PNFD to be on-boarded.
userDefinedData	O	0..N	KeyValuePair	User defined data for the PNFD.

7.2.8.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.8.3-1.

Table 7.2.8.3-1: On-board PNFD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier	Identifier of the on-boarded instance of the PNFD.

7.2.8.4 Operation results

The result of the operation indicates if the on-boarding of the PNFD has been successful or not with a standard success/error result.

The pnfdInfoId is only returned when the operation has been successful.

Once on-boarded, the PNFD is known to and validated by the NFVO.

7.2.9 Update PNFD operation

7.2.9.1 Description

This operation will update a PNFD, creating a new version of already on-boarded PNFD. The operation can also be used to update the userDefinedData of an existing PnfInfo information element without creating a new version of the PNFD.

The previous versions of the PNFD are not modified.

Table 7.2.9.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.9.1-1: Update PNFD operation

Message	Requirement	Direction
UpdatePnfdRequest	Mandatory	OSS/BSS → NFVO
UpdatePnfdResponse	Mandatory	NFVO → OSS/BSS

7.2.9.2 Input parameters

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

Table 7.2.9.2-1: Update PNFD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pfnfdInfold	M	1	Identifier	Identifier of the on-boarded version of the PNFD.
pfnfd	M	0..1	Pnfd	New PNFD to be created. See note.
userDefinedData	O	0..N	KeyValuePair	User defined data to be updated. For existing Keys, the value is replaced. See note.
NOTE: At least one of the two parameters shall be present. If pfnfd is not present, the operation is used only to update existing or add additional user defined data using the userDefinedData parameter.				

7.2.9.3 Output parameters

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

Table 7.2.9.3-1: Update PNFD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
pfnfdInfold	M	1	Identifier	Identifier of the updated PNFD.

7.2.9.4 Operation results

The result of the operation indicates if it has been successful or not with a standard success/error result. The pfnfdInfoId is only returned when the operations has been successful.

If the PNFD was not already on-boarded, this operation returns an error.

7.2.10 Delete PNFD operation

7.2.10.1 Description

This operation will delete one or more PNFDs.

A PNFD can only be deleted when there is no NS (in the active or NOT_INSTANTIATED state) using it.

It is not possible to instantiate NSs that include a PNFD in deletion pending state.

Table 7.2.10.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.10.1-1: Delete PNFD operation

Message	Requirement	Direction
DeletePnfdRequest	Mandatory	OSS/BSS → NFVO
DeletePnfdResponse	Mandatory	NFVO → OSS/BSS

7.2.10.2 Input parameters

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

Table 7.2.10.2-1: Delete PNFD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfdInfold	M	1..N	Identifier	Identifier of the on-boarded instance(s) of the PNFD(s) to be deleted.
applyOnAllVersions	O	0..1	Boolean	Indicates if the delete operation is to be applied on all versions of these PNFD instances. By default, if not present, it applies only on the current version.

7.2.10.3 Output parameters

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

Table 7.2.10.3-1: Delete PNFD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deletedPnfdInfold	M	1..N	Identifier	Identifier of the deleted PNFD(s).

7.2.10.4 Operation results

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

If one of the PNFDs is still in use and is marked as deletion pending, this is indicated as part of the operation result.

If it is marked in deletion pending, it will be deleted once all instantiated NSs using this PNFD are terminated.

7.2.11 Query PNFD operation

7.2.11.1 Description

This operation will enable the OSS/BSS to query the NFVO concerning details of one or more PNFDs.

Table 7.2.11.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.11.1-1: Query PNFD operation

Message	Requirement	Direction
QueryPnfdRequest	Mandatory	OSS/BSS → NFVO
QueryPnfdResponse	Mandatory	NFVO → OSS/BSS

7.2.11.2 Input parameters

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

Table 7.2.11.2-1: Query PNFD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filter defining the PNFDs on which the query applies, based on attributes of the PNFDs It can also be used to specify one or more PNFDs to be queried by providing their identifiers.
attributeSelector	M	0..N	String	Provides a list of attribute names of the PNFD. If present, only these attributes are returned for the instances of PNFD matching the filter. If absent, the complete PNFD instances are returned.

7.2.11.3 Output parameters

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

Table 7.2.11.3-1: Query PNFD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	PnfdInfo	Details of the on-boarded PNFD matching the input filter.

7.2.11.4 Operation results

After success operation, the NFVO has queried the internal PNFD information objects. The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular query, information about the PNFDs that the consumer has access to and that are matching the filter is returned.

7.2.12 Subscribe operation

7.2.12.1 Description

This operation enables the OSS/BSS to subscribe with a filter for the notifications related to changes of NSD sent by the NFVO.

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

Table 7.2.12.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.12.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	OSS/BSS → NFVO
SubscribeResponse	Mandatory	NFVO → OSS/BSS

7.2.12.2 Input parameters

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

Table 7.2.12.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting the NSD(s) and the related change notifications to subscribe to. This filter can contain information about specific types of changes to subscribe to, or attributes of the NsdInfo.

7.2.12.3 Output parameters

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

Table 7.2.12.3-1: Subscribe operation output parameters

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

7.2.12.4 Operation results

After successful subscription, the consumer (OSS/BSS) is registered to receive notifications about changes of NSD.

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

7.2.13 Notify operation

7.2.13.1 Description

This operation distributes notifications to subscribers related to NSD Management changes. It is a one-way operation issued by the NFVO that cannot be invoked as an operation by the consumer (OSS/BSS).

In order to receive notifications, the OSS/BSS shall have a subscription.

Table 7.2.13.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.13.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

The following notification is sent by this operation:

- NsdOnBoardingNotification. See clause 8.2.6.
- NsdChangeNotification. See clause 8.2.7.

7.3 NS Lifecycle Management interface

7.3.1 Description

This interface allows the OSS/BSS to invoke NS lifecycle management operations towards the NFVO.

The following operations are defined for this interface:

- Create NS Identifier;
- Instantiate NS;
- Scale NS;
- Update NS;
- Query NS;
- Terminate NS;
- Delete NS Identifier;
- Heal NS;
- Get Operation Status.

An identifier (i.e. lifecycleOperationOccurrenceId) is generated for each NS lifecycle operation occurrence, except for Query NS, Create NS, Delete NS and Get operation status.

Furthermore, this interface allows the OSS/BSS to manage subscriptions to notifications sent by the NFVO which inform about changes of a NS instance that are related to NS lifecycle management operation occurrences, related to updates of NS information attributes as well as related to the creation/deletion of a NS instance identifier. It further allows the NFVO to provide such notifications to the subscriber.

7.3.2 Create NS Identifier operation

7.3.2.1 Description

This operation creates an NS instance identifier, and an associated instance of an NsInfo information element, identified by that identifier, in the NOT_INSTANTIATED state without instantiating the NS or doing any additional lifecycle operation(s). It allows the immediate return of an NS instance identifier that can be used in subsequent lifecycle operations, such as the Instantiate NS operation. The NS state model is provided in clause D.3.

Table 7.3.2.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.2.1-1: Create NS Identifier operation

Message	Requirement	Direction
CreateNsIdentifierRequest	Mandatory	OSS/BSS → NFVO
CreateNsIdentifierResponse	Mandatory	NFVO → OSS/BSS

7.3.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.2.2-1.

Table 7.3.2.2-1: Create NS Identifier operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdId	M	1	Identifier (Reference to Nsd)	Reference to the NSD used to create this NS instance.
nsName	M	1	String	Human readable name of the NS instance.
nsDescription	M	1	String	Human readable description of the NS instance.

7.3.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.3.2.3-1.

Table 7.3.2.3-1: Create NS Identifier operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the instance of a NS that has been created.

7.3.2.4 Operation results

In case of success, an NS instance identifier and the associated instance of an NsInfo information element has been created in the NOT_INSTANTIATED state and can be used in subsequent lifecycle operations. In case of failure, appropriate error information is returned.

7.3.3 Instantiate NS operation

7.3.3.1 Description

This operation will instantiate an NS. This operation can only be used with an NS instance in the NOT_INSTANTIATED state.

The operation allows for references to existing VNF instances and NS instances that are to be used in the new NS (i.e. the NS being instantiated) and additional parameterization for new VNFs and NSs. The hierarchy of nested NS and VNFs below the NS being instantiated shall be acyclic (i.e. no loops).

NOTE: The vnfProfile information element in the NSD allows the OSS/BSS to specify the number of VNFs to be created at NS instantiation time. It is possible for this number to be zero.

An NSD instance, which can be reused among different NS instantiations, shall have been indicated using the Create NS operation (see clause 7.3.2) previous to executing the Instantiate NS operation.

Table 7.3.3.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.3.1-1: Instantiate NS operation

Message	Requirement	Direction
InstantiateNsRequest	Mandatory	OSS/BSS → NFVO
InstantiateNsResponse	Mandatory	NFVO → OSS/BSS

7.3.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.3.2-1.

Table 7.3.3.2-1: Instantiate NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the instance of the NS.
flavourId	M	1	Identifier (Reference to NsDf)	Flavour of the NSD used to instantiate this NS. See note 1.
sapData	M	0..N	SapData	Create data concerning the SAPs of this NS.
pnfInfo	M	0..N	PnfInfo	Information on the PNF(s) that are part of this NS.
vnfInstanceData	M	0..N	VnfInstanceData	Specify an existing VNF instance to be used in the NS. If needed, the VNF Profile to be used for this VNF instance is also provided. See note 2.
nestedNsInstanceId	M	0..N	Identifier	Specify an existing NS instance to be used as a nested NS within the NS. See notes 3 and 4.
locationConstraints	M	0..N	VnfLocationConstraint	Defines the location constraints for the VNF to be instantiated as part of the NS instantiation. An example can be a constraint for the VNF to be in a specific geographic location.
additionalParamForNs	M	0..N	KeyValuePair	Allows the OSS/BSS to provide additional parameter(s) at the NS level (as opposed to the VNF level, which is covered in additionalParamForVnf).
additionalParamForVnf	M	0..N	ParamsForVnf	Allows the OSS/BSS to provide additional parameter(s) per VNF instance (as opposed to the NS level, which is covered in additionalParamForNs). This is for VNFs that are to be created by the NFVO as part of the NS instantiation and not for existing VNF that are referenced for reuse.
startTime	M	0..1	DateTime	Timestamp indicating the earliest time to instantiate the NS. Cardinality "0" indicates the NS instantiation takes place immediately.
nsInstantiationLevelId	M	0..1	Identifier	Identifies one of the NS instantiation levels declared in the DF applicable to this NS instance. If not present, the default NS instantiation level as declared in the NSD shall be used.

Parameter	Qualifier	Cardinality	Content	Description
additionalAffinityOrAntiAffinityRule	M	0..N	AffinityOrAntiAffinityRule	Specifies additional affinity or anti-affinity constraint for the VNF instances to be instantiated as part of the NS instantiation. Shall not conflict with rules already specified in the NSD.
NOTE 1: The NsDf information element is defined in ETSI GS NFV-IFA 014 [3], clause 6.3.2.				
NOTE 2: The DF of the VNF instance shall match the VNF DF present in the associated VNF Profile.				
NOTE 3: The NS DF of each nested NS shall be one of the allowed flavours in the associated NSD (as referenced in the nestedNsd attribute of the NSD of the NS to be instantiated).				
NOTE 4: The NSD of each referenced NSs (i.e. each nestedInstanceD) shall match the one of the nested NSD in the composite NSD.				

7.3.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.3.3.3-1.

Table 7.3.3.3-1: Instantiate NS operation output parameters

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

7.3.3.4 Operation results

In case of success, the NS has been instantiated. In case of failure, appropriate error information is provided in the "result" Lifecycle Change Notification.

The NFVO shall first return the lifecycleOperationOccurrenceId and second send the "start" Lifecycle Change Notification (see NsLifecycleChangeNotification in clause 8.3.2.2) before additional notifications or messages as part of this operation are issued, or operations towards the VNFM or VIM are invoked.

On the successful as well as the unsuccessful completion of the operation, the NFVO shall send the "result" Lifecycle Change Notification.

If the NS instance was already in the INSTANTIATED state, this operation fails.

7.3.4 Scale NS operation

7.3.4.1 Description

This operation will scale an NS instance. Scaling an NS instance can be performed by explicitly adding/removing existing VNF instances to/from the NS instance, by leveraging on the abstraction mechanism provided by the NS scaling aspects and NS levels information elements declared in the NSD or by scaling individual VNF instances that are part of the NS itself. When adding VNFs and nested NSs - already existing or not - to the NS to be scaled, the NFVO shall follow the indications provided by the dependencies attribute, as specified in the corresponding NSD.

NOTE: In case the NS is a composite NS, it is also possible to scale directly its nested NS, as they are also NS and thus indirectly effectively scale the composite NS.

Table 7.3.4.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.4.1-1: Scale NS operation

Message	Requirement	Direction
ScaleNsRequest	Mandatory	OSS/BSS → NFVO
ScaleNsResponse	Mandatory	NFVO → OSS/BSS

7.3.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.4.2-1.

Table 7.3.4.2-1: Scale NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceid	M	1	Identifier	Identifier of the instance of the NS.
scaleType	M	1	Enum	Indicates the type of scaling to be performed. Possible values: SCALE_NS, SCALE_VNF.
scaleNsData	M	0..1	ScaleNsData	Provides the necessary information to scale the referenced NS instance. It shall be present when scaleType = SCALE_NS. See note.
scaleVnfData	M	0..N	ScaleVnfData	Provides the information to scale a given VNF instance that is part of the referenced NS instance. Shall be present when scaleType = SCALE_VNF. See note.
scaleTime	M	0..1	DateTime	Timestamp indicating the scale time of the NS, i.e. the NS will be scaled at this timestamp. Cardinality "0" indicates the NS scaling takes place immediately.

NOTE: Either scaleNsData or scaleVnfData, but not both, shall be present.

7.3.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.3.4.3-1.

Table 7.3.4.3-1: Scale NS operation output parameters

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

7.3.4.4 Operation results

In case of success, the NS instance has been scaled according to the request. In case of failure, appropriate error information is provided in the "result" Lifecycle Change Notification.

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

In case of scaling in an NS, if some VNF instances are removed from this NS instance, these VNF instances are terminated unless they are still part of another NS instance.

On the successful as well as the unsuccessful completion of the operation, the NFVO shall send the "result" Lifecycle Change Notification.

7.3.5 Update NS operation

7.3.5.1 Introduction

This operation updates an NS instance. This operation is also used to embed VNF LCM operations in support of fine grained NS LCM approach. See the informative message flows in annex C. Actions that can be performed with an update include:

- Adding existing VNF instances to the NS instance.
- Removing VNF instances from the NS instance.
- Instantiating new VNF instances and adding them to the NS instance.
- Changing the DF of VNF instances belonging to the NS instance.
- Changing the operational state of a VNF instance belonging to the NS instance.
- Modifying information data and/or the configurable properties of a VNF instance belonging to the NS instance.
- Changing the external connectivity of a VNF instance belonging to the NS instance.
- Adding SAPs to the NS instance.
- Removing SAPs from the NS instance.
- Adding existing NS instances to the NS instance.
- Removing nested NS instances from the NS instance.
- Associate a new NSD version to the NS instance.
- Moving VNF instances from one NS instance to another NS instance.
- Adding VNFFGs to the NS instance.
- Removing VNFFGs from the NS instance.
- Update VNFFGs of the NS instance.
- Changing the DF of the NS instance.

Only one type of update shall be allowed per operation.

Table 7.3.5.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO. It is possible, however, to request several updates of a given type in one Update NS operation (as indicated in the cardinalities in table 7.3.5.2-1).

Table 7.3.5.1-1: Update NS operation

Message	Requirement	Direction
UpdateNsRequest	Mandatory	OSS/BSS → NFVO
UpdateNsResponse	Mandatory	NFVO → OSS/BSS

7.3.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.5.2-1.

Table 7.3.5.2-1: Update NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstancelId	M	1	Identifier	Identifier of the NS instance being updated.
updateType	M	1	Enum	Specifies the type of update. This parameter determines also which one of the following parameter is present in the operation. Possible values are: <ul style="list-style-type: none"> • AddVnf (adding existing VNF instance(s)), • RemoveVnf (removing VNF instance(s)), • InstantiateVnf (instantiating new VNF(s)), • ChangeVnfDf (Changing VNF DF), • OperateVnf (changing VNF state), • ModifyVnfInformation (modifying VNF information and/or the configurable properties of VNF instance(s)), • ChangeExtVnfConnectivity (changing the external connectivity of VNF instance(s)). • AddSap (adding SAP(s)), • RemoveSap (removing SAP(s)), • AddNestedNs (adding existing NS instance(s) as nested NS(s)), • RemoveNestedNs (removing existing nested NS instance(s)), • AssocNewNsdVersion (associating a new NSD version to the NS instance), • MoveVnf (moving VNF instance(s) from one origin NS instance to a another target NS instance), • AddVnffg (adding VNFFG(s)), • RemoveVnffg (removing VNFFG(s)), • UpdateVnffg (updating VNFFG(s)), • ChangeNsDf (changing NS DF).
addVnfInstance	M	0..N	VnfInstanceData	Specify an existing VNF instance to be added to the NS instance. This parameter shall be present only if updateType=AddVnf.
removeVnfInstancelId	M	0..N	Identifier	Specify an existing VNF instance to be removed from the NS instance. The parameter contains the identifier(s) of the VNF instances to be removed. This parameter shall be present only if updateType=RemoveVnf. See note.

Parameter	Qualifier	Cardinality	Content	Description
instantiateVnfData	M	0..N	InstantiateVnfData	Specify the new VNF to be instantiated. This parameter can be used e.g. for the bottom-up NS creation. This parameter shall be present only if updateType=InstantiateVnf.
changeVnfFlavourData	M	0..N	ChangeVnfFlavourData	Specify the new DF of the VNF instance to be changed to. This parameter shall be present only if updateType=ChangeVnfDf.
operateVnfData	M	0..N	OperateVnfData	Specify the state of the VNF instance to be changed. This parameter shall be present only if updateType=OperateVnf.
modifyVnfInfoData	M	0..N	ModifyVnfInfoData	Specify the VNF Information parameters and/or the configurable properties of VNF instance to be modified. This parameter shall be present only if updateType=ModifyVnfInformation.
changeExtVnfConnectivityData	M	0..N	ChangeExtVnfConnectivityData	Specify the new external connectivity data of the VNF instance to be changed. This parameter shall be present only if updateType= ChangeExtVnfConnectivity.
addSap	M	0..N	SapData	Specify a new SAP to be added to the NS instance. This parameter shall be present only if updateType=AddSap.
removeSapId	M	0..N	Identifier	Specify an existing SAP to be removed from the NS instance. The parameter shall be present only if updateType=RemoveSap.
addNestedNsId	M	0..N	Identifier	Specify an existing nested NS instance to be added to (nested within) the NS instance. This parameter shall be present only if updateType=AddNestedNs.
removeNestedNsId	M	0..N	Identifier	Specify an existing nested NS instance to be removed from the NS instance. The parameter shall be present only if updateType=RemoveVnfNestedNs.
assocNewNsdVersionData	M	0..1	AssocNewNsdVersionData	Specify the new NSD to be used for the NS instance. This parameter shall be present only if updateType=AssocNewNsdVersion.
moveVnfInstanceData	M	0..N	MoveVnfInstanceData	Specify existing VNF instance to be moved from one NS instance to another NS instance. This parameter shall be present only if updateType=MoveVnf.
addVnffg	M	0..N	AddVnffgData	Specify the new VNFFG to be created to the NS Instance. This parameter shall be present only if updateType=AddVnffg.
removeVnffgId	M	0..N	Identifier	Identifier of an existing VNFFG to be removed from the NS Instance. This parameter shall be present only if updateType=RemoveVnffg.
updateVnffg	M	0..N	UpdateVnffgData	Specify the new VNFFG Information data to be updated for a VNFFG of the NS Instance. This parameter shall be present only if updateType=UpdateVnffg.
changeNsFlavourData	M	0..1	ChangeNsFlavourData	Specifies the new DF to be applied to the NS instance. It shall be present only if updateType=ChangeNsDf.
updateTime	M	0..1	DateTime	Timestamp indicating the update time of the NS, i.e. the NS will be updated at this timestamp. Cardinality "0" indicates the NS update takes place immediately.
NOTE: If a VNF instance is removed from an NS and this NS was the last one for which this VNF instance was a part, the VNF instance is terminated by the NFVO.				

7.3.5.3 Output parameters

The output parameter returned by the operation shall follow the indications provided in table 7.3.5.3-1.

Table 7.3.5.3-1: Update NS operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	0..N	Identifier	Identifier of the instance of the instantiated VNF. This information shall be returned as the result of the operation if successful.
vnffgId	M	0..N	Identifier	Identifier of the instance of the created VNFFG. It shall be present only if updateType = AddVnffg. This information shall be returned as the result of the operation if successful.
sapId	M	0..N	Identifier	Identifier of the instance of the created SAP. It shall be present only if updateType = addSap. This information shall be returned as the result of the operation if successful.
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the NS lifecycle operation occurrence. This information shall be returned immediately before any notification, message or operation is done.

7.3.5.4 Operation results

In case of success, the NS has been updated according to the request. In case of failure, appropriate error information is provided in the "result" Lifecycle Change Notification.

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

On successful as well as unsuccessful completion of the operation, the NFVO shall send the "result" Lifecycle Change Notification.

7.3.6 Query NS operation

7.3.6.1 Description

This operation will enable the OSS/BSS to query from the NFVO information on one or more NS(s). The operation also supports querying information about VNF instance(s) that is (are) part of an NS.

Table 7.3.6.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.6.1-1: Query NS operation

Message	Requirement	Direction
QueryNsRequest	Mandatory	OSS/BSS → NFVO
QueryNsResponse	Mandatory	NFVO → OSS/BSS

7.3.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.6.2-1.

Table 7.3.6.2-1: Query NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filter defining the NSs on which the query applies, based on attributes of the Network Service. It can also be used to specify one or more NS(s) to be queried by providing their identifiers. It can also be used to specify one or more VNF instances(s) that are part of an NS by providing their identifiers.
attributeSelector	M	0..N	String	Provides a list of attribute names of NS. If present, only these attributes are returned for the instances of NS matching the filter. If absent, the complete instances of NS(s) are returned. In the case of query information about VNF instance(s) that are part of an NS, it provides a list of attribute names. And only the attributes are returned for the VNF instance(s) matching the filter. And if absent, the complete information are returned for the VNF instance(s) matching the filter.

7.3.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.3.6.3-1.

Table 7.3.6.3-1: Query NS operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryNsResult	M	0..N	NsInfo	Information on the NS and VNF instances part of the NS matching the input filter. If attributeSelector is present, only the attributes listed in attributeSelector are returned for the selected NSs and VNF instances. See note.
NOTE: The lower cardinality is 0 since there may be no matches to the provided filter.				

7.3.6.4 Operation results

After success operation, the NFVO has queried the internal NS information objects including retrieved requested VNF instance information. The result of the operation indicates whether it has been successful or not with a standard success/error result. For a particular query, information about the NSs including VNF instance(s) part of the NS that the consumer has access to and that are matching the filter shall be returned.

7.3.7 Terminate NS operation

7.3.7.1 Description

This operation will terminate an NS.

This operation can only be used with an NS instance in the INSTANTIATED state.

Terminating an NS instance does not delete the NS instance identifier, and the associated instance of the NsInfo information element, but rather transitions the NS into the NOT_INSTANTIATED state.

Table 7.3.7.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.7.1-1: Terminate NS operation

Message	Requirement	Direction
TerminateNsRequest	Mandatory	OSS/BSS → NFVO
TerminateNsResponse	Mandatory	NFVO → OSS/BSS

7.3.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.7.2-1.

Table 7.3.7.2-1: Terminate NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the NS instance to terminate.
terminateTime	M	0..1	DateTime	Timestamp indicating the end time of the NS, i.e. the NS will be terminated automatically at this timestamp. Cardinality "0" indicates the NS termination takes place immediately.

7.3.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.3.7.3-1.

Table 7.3.7.3-1: Terminate NS output parameters

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

7.3.7.4 Operation results

In case of success, the NS has been terminated (i.e. put into NOT_INSTANTIATED state), and resources used by the NS have been released. As part of the NS termination, the following actions take place:

- All the VNF instances part of the terminated NS are terminated, unless they are still part of any other NS instance(s);
- All VLs, VNF FGs and information on PNF Connection Points (CPs) created at NS instantiation are deleted;
- Nested NS instances are just released and not terminated.

NOTE: It is possible to avoid termination of the constituent VNFs by first moving the VNFs to another NS (by requesting "Update NS/Move Vnf" before the Terminate NS request).

In case of failure, appropriate error information is provided in the "result" Lifecycle Change Notification.

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

On successful as well as unsuccessful completion of the operation, the NFVO shall send the "result" Lifecycle Change Notification.

If the NS instance was already in the NOT_INSTANTIATED state, this operation fails.

7.3.8 Delete NS Identifier operation

7.3.8.1 Description

This operation deletes an NS instance identifier and the associated NsInfo information element which is in the NOT_INSTANTIATED state.

Table 7.3.8.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.8.1-1: Delete NS Identifier operation

Message	Requirement	Direction
DeleteNsRequest	Mandatory	OSS/BSS → NFVO
DeleteNsResponse	Mandatory	NFVO → OSS/BSS

7.3.8.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.8.2-1.

Table 7.3.8.2-1: Delete NS Identifier operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstancelid	M	1	Identifier	NS instance identifier to be deleted.

7.3.8.3 Output parameters

No output parameter.

7.3.8.4 Operation results

In case of success, the NS instance identifier and the associated instance of the NsInfo information element has been deleted and can no longer be used. If the NS instance was not in the NOT_INSTANTIATED state (i.e. terminated or not instantiated), the operation is rejected.

In case of failure, appropriate error information is returned.

7.3.9 Heal NS operation

7.3.9.1 Description

This operation supports the healing of an NS instance, either by healing the complete NS instance or by healing one of more of the VNF instances that are part of this NS.

NOTE: In case the NS is a composite NS, it is also possible to execute individual heal requests on one or more of the NSs that are nested within this NS.

Table 7.3.9.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.9.1-1: Heal NS operation

Message	Requirement	Direction
HealNsRequest	Mandatory	OSS/BSS → NFVO
HealNsResponse	Mandatory	NFVO → OSS/BSS

7.3.9.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.9.2-1.

Table 7.3.9.2-1: Heal NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	The parameter identifies the NS instance which shall be healed.
healNsData	M	0..1	HealNsData	Provides the information needed to heal an NS. See note.
healVnfData	M	0..N	HealVnfData	Provides the information needed to heal a VNF. See note.
NOTE: Either the parameter healNsData or the parameter healVnfData, but not both shall be provided.				

7.3.9.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.3.9.3-1.

Table 7.3.9.3-1: Heal NS output parameters

Parameter	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of the NS lifecycle operation occurrence. This information shall be returned immediately before any notification, message or operation is done.

7.3.9.4 Operation results

In case of success, the NS has been healed, that means complete or partial healing as requested. In case of failure, appropriate error information is provided in the "result" Lifecycle Change Notification.

NOTE: Testing procedures could be used e. g. to find the root cause of a failure situation.
In addition testing procedures could also be applied during or after the healing process to check whether the healing actions were successful, etc.

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

On the successful as well as the unsuccessful completion of the operation, the NFVO shall send the "result" Lifecycle Change Notification.

7.3.10 Get Operation Status operation

7.3.10.1 Description

This operation provides the status of an NS lifecycle management operation.

Table 7.3.10.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.10.1-1: Get Operation Status operation

Message	Requirement	Direction
GetOperationStatusRequest	Mandatory	OSS/BSS → NFVO
GetOperationStatusResponse	Mandatory	NFVO → OSS/BSS

7.3.10.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.10.2-1.

Table 7.3.10.2-1: Get Operation Status operation input parameters

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

7.3.10.3 Output Parameters

The output parameters returned by the operation shall follow the indications provided in table 7.3.10.3-1.

Table 7.3.10.3-1: Get Operation Status operation output parameters

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

7.3.10.4 Operation results

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

7.3.11 Subscribe operation

7.3.11.1 Description

This operation enables the OSS/BSS to subscribe with a filter for the notifications sent by the NFVO which are related to NS lifecycle changes, as well as to the creation/deletion of NS instance identifiers and the associated NsInfo information element instances.

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

Table 7.3.11.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.11.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	OSS/BSS → NFVO
SubscribeResponse	Mandatory	NFVO → OSS/BSS

7.3.11.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.11.2-1.

Table 7.3.11.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting the notifications. It can be on the NS instances of interest or other attributes of the notification.

7.3.11.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.3.11.3-1.

Table 7.3.11.3-1: Subscribe operation output parameters

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

7.3.11.4 Operation results

After successful subscription, the consumer (OSS/BSS) is registered to receive notifications about NS lifecycle changes, as well as the creation/deletion of NS instance identifiers and the associated NsInfo information element instances.

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

7.3.12 Notify operation

7.3.12.1 Description

This operation distributes to subscribers notifications about NS lifecycle changes, as well as the creation/deletion of NS instance identifiers and the associated NsInfo information element instances. It is a one-way operation issued by the producer (NFVO) that cannot be invoked as an operation by the consumer (OSS/BSS). In order to receive notifications, the consumer (OSS/BSS) has to perform an explicit Subscribe operation beforehand.

Table 7.3.12.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.12.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

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

- NsLifecycleChangeNotification. See clause 8.3.2.2.
- NsIdentifierCreationNotification. See clause 8.3.2.9.
- NsIdentifierObjectDeletionNotification. See clause 8.3.2.10.

7.3.13 Terminate Subscription operation

7.3.13.1 Description

This operation enables the OSS/BSS to terminate a particular subscription.

Table 7.3.13.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.13.1-1: Terminate Subscription operation

Message	Requirement	Direction
TerminateSubscriptionRequest	Mandatory	OSS/BSS → NFVO
TerminateSubscriptionResponse	Mandatory	NFVO → OSS/BSS

7.3.13.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.13.2-1.

Table 7.3.13.2-1: Terminate Subscription operation input parameters

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

7.3.13.3 Output parameters

None.

7.3.13.4 Operation results

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

7.3.14 Query Subscription Info operation

7.3.14.1 Description

This operation enables the OSS/BSS to query information about subscriptions.

Table 7.3.14.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.3.14.1-1: Query Subscription Info operation

Message	Requirement	Direction
QuerySubscriptionInfoRequest	Mandatory	OSS/BSS → NFVO
QuerySubscriptionInfoResponse	Mandatory	NFVO → OSS/BSS

7.3.14.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.3.14.2-1.

Table 7.3.14.2-1: Query Subscription Info operation input parameters

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

7.3.14.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.3.14.3-1.

Table 7.3.14.3-1: Query Subscription Info operation output parameters

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

7.3.14.4 Operation results

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

7.4 Void

7.5 NS Performance Management interface

7.5.1 Description

This interface allows providing of performance information (measurement results collection and notifications) related to network services.

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

Performance information on a given NS results from either collected performance information of the virtualised resources impacting the connectivity of this NS instance or VNF performance information, resulting from virtualised resource performance information, issued by the VNFM for the VNFs that is part of this NS instance.

When new performance information is available, the consumer is notified using the notification PerformanceInformationAvailableNotification (see clause 8.4.8). The details of the performance measurements are provided using the PerformanceReport information element (see clause 8.4.5). Delivery mechanism for the performance reports is left for later specification.

The following operations are defined for this interface which will be consumed by the OSS/BSS:

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

7.5.2 Create PM Job operation

7.5.2.1 Description

This operation will create a PM job, enabling an OSS/BSS to specify an NS or set of NSs, that the NFVO is managing, for which it wants to receive performance information. This will allow the requesting OSS/BSS to specify its performance information requirements with the NFVO.

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

Table 7.5.2.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.5.2.1-1: Create PM Job operation

Message	Requirement	Direction
CreatePmJobRequest	Mandatory	OSS/BSS → NFVO
CreatePmJobResponse	Mandatory	NFVO → OSS/BSS

7.5.2.2 Input parameters

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

Table 7.5.2.2-1: Create PM Job operation input parameters

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

7.5.2.3 Output parameters

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

Table 7.5.2.3-1: Create PM Job operation output parameters

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

7.5.2.4 Operation results

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

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

7.5.3 Delete PM Jobs operation

7.5.3.1 Description

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

Table 7.5.3.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.5.3.1-1: Delete PM Jobs operation

Message	Requirement	Direction
DeletePmJobsRequest	Mandatory	OSS/BSS → NFVO
DeletePmJobsResponse	Mandatory	NFVO → OSS/BSS

7.5.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.3.2-1.

Table 7.5.3.2-1: Delete PM Jobs operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pmJobId	M	1..N	Identifier	Identifiers of the PM jobs to be deleted.

7.5.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.5.3.3-1.

Table 7.5.3.3-1: Delete PM Jobs operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deletedPmJobId	M	1..N	Identifier	Identifiers of the PM Jobs that have been deleted successfully.

7.5.3.4 Operation results

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

7.5.4 Subscribe operation

7.5.4.1 Description

This operation enables the OSS/BSSs to subscribe with a filter for the notifications related to performance information with the NFVO.

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

Table 7.5.4.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.5.4.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	OSS/BSS → NFVO
SubscribeResponse	Mandatory	NFVO → OSS/BSS

7.5.4.2 Input parameters

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

Table 7.5.4.2-1: Subscribe operation input parameters

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

7.5.4.3 Output parameters

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

Table 7.5.4.3-1: Subscribe operation output parameters

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

7.5.4.4 Operation results

As a result of this operation, the NFVO shall indicate to the OSS/BSS in the subscribeResponse message whether the subscription was successful or not.

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

7.5.5 Notify operation

7.5.5.1 Description

This operation distributes notifications to subscribers. It is a one-way operation issued by the NFVO that cannot be invoked as an operation by the consumer (OSS/BSS).

In order to receive notifications, the OSS/BSS shall have a subscription.

Table 7.5.5.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.5.5.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

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

- PerformanceInformationAvailableNotification. See clause 8.4.8.
- ThresholdCrossedNotification. See clause 8.4.9.

7.5.6 Query PM Job operation

7.5.6.1 Description

This operation will enable the OSS/BSS to solicit from the NFVO the details of one or more PM job(s).

This operation does not return performance reports.

Table 7.5.6.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.5.6.1-1: Query PM Job operation

Message	Requirement	Direction
QueryPmJobRequest	Mandatory	OSS/BSS → NFVO
QueryPmJobResponse	Mandatory	NFVO → OSS/BSS

7.5.6.2 Input parameters

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

Table 7.5.6.2-1: Query PM Job operation input parameters

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

7.5.6.3 Output parameters

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

Table 7.5.6.3-1: Query PM Job operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
pmJobDetails	M	1..N	PmJob	Details of PM jobs matching the input filter.

7.5.6.4 Operation results

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

7.5.7 Create Threshold operation

7.5.7.1 Description

This operation will allow the OSS/BSS to create a threshold and specify threshold levels on specified performance metric (for NS(s)) for which notifications will be generated when crossed.

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

Table 7.5.7.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.5.7.1-1: Create Threshold operation

Message	Requirement	Direction
CreateThresholdRequest	Mandatory	OSS/BSS → NFVO
CreateThresholdResponse	Mandatory	NFVO → OSS/BSS

7.5.7.2 Input parameters

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

Table 7.5.7.2-1: Create Threshold operation input parameters

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

7.5.7.3 Output parameters

The parameters returned by the operation shall follow the indications provided in table 7.5.7.3-1.

Table 7.5.7.3-1: Create Threshold operation output parameters

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

7.5.7.4 Operation results

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

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

7.5.8 Delete Thresholds operation

7.5.8.1 Description

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

Table 7.5.8.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.5.8.1-1: Delete Thresholds operation

Message	Requirement	Direction
DeleteThresholdsRequest	Mandatory	OSS/BSS → NFVO
DeleteThresholdsResponse	Mandatory	NFVO → OSS/BSS

7.5.8.2 Input parameters

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

Table 7.5.8.2-1: Delete Thresholds operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
thresholdId	M	1..N	Identifier	Identifiers of the thresholds to be deleted.

7.5.8.3 Output parameters

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

Table 7.5.8.3-1: Delete Thresholds operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deletedThresholdId	M	1..N	Identifier	Identifiers of the thresholds that have been deleted successfully.

7.5.8.4 Operation results

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

7.5.9 Query Threshold operation

7.5.9.1 Description

This operation will allow the OSS/BSS to query the details of an existing threshold.

Table 7.5.9.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.5.9.1-1: QueryThreshold operation

Message	Requirement	Direction
QueryThresholdRequest	Mandatory	OSS/BSS → NFVO
QueryThresholdResponse	Mandatory	NFVO → OSS/BSS

7.5.9.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.9.2-1.

Table 7.5.9.2-1: QueryThreshold operation input parameters

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

7.4.9.3 Output parameters

The parameters returned by the operation shall follow the indications provided in table 7.5.9.3-1.

Table 7.5.9.3-1: QueryThreshold operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
thresholdDetails	M	1..N	Threshold	List of threshold details matching the input filter.

7.5.9.4 Operation results

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

7.6 NS Fault Management interface

7.6.1 Description

This interface shall allow the NFVO to provide alarms related to the NSs visible to the consumer.

An alarm on a given NS results from either a collected virtualised resource fault impacting the connectivity of the NS instance or a VNF alarm, resulting from a virtualised resource alarm, issued by the VNFM for a VNF that is part of this NS instance.

The fault management interface shall support the following operations:

- Subscribe operation: Subscription of OSS/BSSs with the NFVO for the notifications related to the alarms.
- Notify operation: Notifications of alarms or alarm state change from NFVO to OSS/BSS.
- Get alarm list operation: Accessing active alarms from the NFVO.

7.6.2 Subscribe operation

7.6.2.1 Description

This operation enables the OSS/BSS to subscribe with a filter for the notifications related to NS alarms sent by the NFVO.

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

Table 7.6.2.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.6.2.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	OSS/BSS → NFVO
SubscribeResponse	Mandatory	NFVO → OSS/BSS

7.6.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.6.2.2-1.

Table 7.6.2.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting NSs and related alarms. This can contain the NS information, severity and cause of the alarm.

7.6.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.6.2.3-1.

Table 7.6.2.3-1: Subscribe operation output parameters

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

7.6.2.4 Operation results

As a result of this operation, the NFVO shall indicate to the OSS/BSS in the subscribeResponse message whether the subscription was successful or not.

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

7.6.3 Notify operation

7.6.3.1 Description

This operation distributes notifications to subscribers. It is a one-way operation issued by the NFVO and cannot be invoked as an operation by the consumer (OSS/BSS).

In order to receive notifications, the OSS/BSS shall have a subscription.

Table 7.6.3.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.6.3.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

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

- AlarmNotification. See clause 8.5.2.
- AlarmClearedNotification. See clause 8.5.3.

7.6.4 Get Alarm List operation

7.6.4.1 Description

This operation enables the OSS/BSSs to query the active alarms from the NFVO.

Table 7.6.4.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.6.4.1-1: Get Alarm List operation

Message	Requirement	Direction
GetAlarmListRequest	Mandatory	OSS/BSS → NFVO
GetAlarmListResponse	Mandatory	NFVO → OSS/BSS

7.6.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.6.4.2-1.

Table 7.6.4.2-1: Get Alarm List operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting alarms. This can contain the list of the NS identifiers, severity and cause.

7.6.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.6.4.3-1.

Table 7.6.4.3-1: Get Alarm List operation output parameters

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

7.6.4.4 Operation results

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

For a particular request, only alarms matching the filter will be delivered to the OSS/BSS.

7.7 VNF Package management interface

7.7.1 Description

This interface allows for the management of VNF Packages.

The following operations are defined for this interface:

- On-board VNF Package.
- Enable VNF Package.
- Disable VNF Package.
- Delete VNF Package.
- Query VNF Package.
- Fetch VNF Package.
- Subscribe to new notifications.
- Notify of on-boarding of new VNF Package or of changes of VNF Packages.

7.7.2 On-board VNF Package operation

7.7.2.1 Description

This operation will on-board a VNF Package in the NFVO.

Table 7.7.2.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.2.1-1: On-board VNF Package operation

Message	Requirement	Direction
OnboardVnfPackageRequest	Mandatory	OSS/BSS → NFVO
OnboardVnfPackageResponse	Mandatory	NFVO → OSS/BSS

7.7.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.2.2-1.

Table 7.7.2.2-1: On-board VNF Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
name	M	1	String	Name of the VNF Package to be on-boarded.
version	M	1	Version	Version of the VNF Package to be on-boarded.
provider	M	1	String	Provider of the VNF Package to be on-boarded.
checksum	M	1	Not specified.	Checksum of the on-boarded VNF Package.
userDefinedData	O	0..N	KeyValuePair	User defined data for the VNF Package.
vnfPackagePath	M	1	URL	Address information based on which the VNF Package can be obtained. See note.

NOTE: This Structure can be the address information related to an FTP server when the VNF Package is stored, or be a URL where the NFVO can download the VNF Package.

7.7.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.7.2.3-1.

Table 7.7.2.3-1: On-board VNF Package operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfold	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF Package. This identifier was allocated by the NFVO.
vnfdId	M	1	Identifier	Identifier that identifies the VNF Package in a globally unique way. See note.
NOTE: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2.				

7.7.2.4 Operation results

The result of the operation indicates if the on-boarding of the VNF Package has been successful or not with a standard success/error result.

The onboardedVnfPkgInfold and vnfdId are only returned when the operation has been successful.

Once on-boarded, the VNF Package is known to and validated by the NFVO. It is in Enabled state, allowing its use for VNF lifecycle management.

7.7.3 Enable VNF Package operation

7.7.3.1 Description

This operation will enable a previously disabled VNF Package instance, allowing again its use for the instantiation of new VNFs and the creation of new VNF identifiers based on this package.

Table 7.7.3.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.3.1-1: Enable VNF Package operation

Message	Requirement	Direction
EnableVnfPackageRequest	Mandatory	OSS/BSS → NFVO
EnableVnfPackageResponse	Mandatory	NFVO → OSS/BSS

7.7.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.3.2-1.

Table 7.7.3.2-1: Enable VNF Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfold	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF Package. This identifier was allocated by the NFVO.

7.7.3.3 Output parameters

No output parameter.

7.7.3.4 Operation results

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

If the VNF Package was already enabled, this operation returns an error.

This operation is rejected if the VNF Package is in deletion pending state.

7.7.4 Disable VNF Package operation

7.7.4.1 Description

This operation will disable a previously enabled VNF Package instance, preventing further use for instantiation of new VNFs with this package (unless and until the VNF Package is re-enabled).

Table 7.7.4.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.4.1-1: Disable VNF Package operation

Message	Requirement	Direction
DisableVnfPackageRequest	Mandatory	OSS/BSS → NFVO
DisableVnfPackageResponse	Mandatory	NFVO → OSS/BSS

7.7.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.4.2-1.

Table 7.7.4.2-1: Disable VNF Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfold	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF Package. This identifier was allocated by the NFVO.

7.7.4.3 Output parameters

No output parameter.

7.7.4.4 Operation results

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

If the VNF Package was already disabled, this operation returns an error.

This operation is rejected if the VNF Package is in deletion pending state.

7.7.5 Delete VNF Package operation

7.7.5.1 Description

This operation will delete a VNF Package.

Note that a VNF Package can only be deleted once there are no VNFs using it.

A deletion pending VNF Package can no longer be enabled, disabled or updated. It is not possible to instantiate VNFs or create VNF identifiers using a VNF Package in the "deletion pending" state.

Table 7.7.5.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.5.1-1: Delete VNF Package operation

Message	Requirement	Direction
DeleteVnfPackageRequest	Mandatory	OSS/BSS → NFVO
DeleteVnfPackageResponse	Mandatory	NFVO → OSS/BSS

7.7.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.5.2-1.

Table 7.7.5.2-1: Delete VNF Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfoId	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF Package, which is to be deleted. This identifier was allocated by the NFVO.

7.7.5.3 Output parameters

No output parameter

7.7.5.4 Operation results

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

If the VNF Package is still in use and is marked as delete pending, this is indicated as part of the operation result.

If it is marked in deletion pending, it will be deleted once all instantiated VNFs are terminated.

7.7.6 Query On-boarded VNF Package Information operation

7.7.6.1 Description

When a VNF Package is on-boarded by the NFVO, the NFVO creates and stores information associated with this VNF Package. It maintains during the VNF Package's operational lifecycle. This operation will enable the OSS/BSS to query from the NFVO for information it has stored about one or more VNF Packages.

The operation allows querying specific components of the information stored in the NFVO about a VNF Package, for instance, retrieving the VNFD.

NOTE: The VNFD is an attribute of the OnboardedVnfPkgInfo.

Table 7.7.6.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.6.1-1: Query On-boarded VNF Package Information operation

Message	Requirement	Direction
QueryOnboardedVnfPkgInfoRequest	Mandatory	OSS/BSS → NFVO
QueryOnboardedVnfPkgInfoResponse	Mandatory	NFVO → OSS/BSS

7.7.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.6.2-1.

Table 7.7.6.2-1: Query On-boarded VNF Package Information operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filter defining the VNF Packages on which the query applies, based on attributes of OnboardedVnfPkgInfo. It can also be used to specify one or more VNF Packages to be queried by providing their vnfdId or onboardedVnfPkgId. See note.
attributeSelector	M	0..N	String	It provides a list of attribute names of onboardedVnfPkgInfo. If present, only these attributes are returned for OnboardedVnfPkgInfo matching the filter. If absent, the complete OnboardedVnfPkgInfo are returned.
NOTE: The vnfdId, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2. The onboardedVnfPkgInfoId identifies the information related to the onboarding of a VNF package into the NFVO, which implies that it also identifies an onboarded VNF package.				

7.7.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.7.6.3-1.

Table 7.7.6.3-1: Query On-boarded VNF Package Information operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	OnboardedVnfPkgInfo	Details of the on-boarded VNF Packages matching the input filter. If attributeSelector is present, only the attributes listed in attributeSelector are returned for the selected entities.

7.7.6.4 Operation results

After success operation, the NFVO has queried the internal VNF Package information objects. The result of the operation indicates if it has been successful or not with a standard success/error result. For a particular query, information about the VNF Package that the consumer has access to and that are matching the filter shall be returned.

7.7.7 Subscribe operation

7.7.7.1 Description

This operation enables the OSS/BSS to subscribe with a filter for the notifications related to on-boarding of VNF Packages and changes of VNF Packages sent by the NFVO.

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

Table 7.7.7.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.7.1-1: Subscribe operation

Message	Requirement	Direction
SubscribeRequest	Mandatory	OSS/BSS → NFVO
SubscribeResponse	Mandatory	NFVO → OSS/BSS

7.7.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.7.2-1.

Table 7.7.7.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
inputFilter	M	1	Filter	Input filter for selecting the VNF Package(s) and the related change notifications to subscribe to. This filter can contain information about specific types of changes to subscribe to, or attributes of the VNF Package.

7.7.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.7.7.3-1.

Table 7.7.7.3-1: Subscribe operation output parameters

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

7.7.7.4 Operation results

After successful subscription, the OSS/BSS is registered to receive notifications related to changes of VNF Packages sent by the NFVO. The result of the operation shall indicate if the subscription has been successful or not with a standard success/error result. For a particular subscription, only notifications matching the filter will be delivered to the OSS/BSS.

7.7.8 Notify operation

7.7.8.1 Description

This operation distributes notifications to subscribers. It is a one-way operation issued by the NFVO that cannot be invoked as an operation by the consumer (OSS/BSS).

In order to receive notifications, the OSS/BSS shall have a subscription.

Table 7.7.8.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.8.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

The following notification is sent by this operation:

- VnfPackageOnBoardingNotification. See clause 8.6.8.
- VnfPackageChangeNotification. See clause 8.6.9.

7.7.9 Void

7.7.10 Fetch On-boarded VNF Package operation

7.7.10.1 Description

This operation enables the OSS to fetch a whole on-boarded VNF Package. The package is addressed using an identifier of information held by the NFVO about the specific on-boarded VNF Package. This identifier is contained within the VnfPackageOnBoardingNotification or is returned as a result of the On-board NSD operation.

Table 7.7.10.1-1 lists the information flow exchanged between the NFVO and the OSS.

Table 7.7.10.1-1: Fetch On-boarded VNF Package operation

Message	Requirement	Direction
FetchOnboardedVnfPackageRequest	Mandatory	OSS → NFVO
FetchOnboardedVnfPackageResponse	Mandatory	NFVO → OSS

7.7.10.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.10.2-1.

Table 7.7.10.2-1: Fetch On-boarded VNF Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfolId	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF Package. This identifier was allocated by the NFVO.

7.7.10.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.7.10.3-1.

Table 7.7.10.3-1: Fetch On-boarded VNF Package operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfPackage	M	1	Binary	The VNF Package.

7.7.10.4 Operation results

After successful operation, the NFVO has provided to the OSS a copy of the requested VNF package.

7.7.11 Fetch On-boarded VNF Package Artifacts operation

7.7.11.1 Description

This operation enables the OSS/BSS to fetch selected artifacts contained in an on-boarded VNF package. Artifacts are addressed using selector information that can be obtained using the QueryOnboardedVnfPkgInfo operation.

Table 7.7.11.1-1 lists the information flow exchanged between the OSS and the NFVO.

Table 7.7.11.1-1: Fetch On-boarded VNF Artifacts operation

Message	Requirement	Direction
FetchOnboardedVnfPackageArtifactsRequest	Mandatory	OSS → NFVO
FetchOnboardedVnfPackageArtifactsResponse	Mandatory	NFVO → OSS

7.7.11.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.11.2.-1.

Table 7.7.11.2-1: Fetch On-boarded VNF Artifacts operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfold	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF Package. This identifier was allocated by the NFVO.
artifactSelector	M	1..N		Selector to address an individual VNF package artifact, or list of selectors to address multiple of those. See note.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to obtain multiple artifacts in one go, or as a series of operations that obtain one artifact at a time.				

7.7.11.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.7.11.3-1.

Table 7.7.11.3-1: Fetch On-boarded VNF Artifacts operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfPackageArtifact	M	1..N	Not specified	A VNF package artifact (e.g. file) or multiple thereof. See note.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to obtain multiple artifacts in one go, or as a series of operations that obtain one artifact at a time.				

7.7.11.4 Operation results

After successful operation, the NFVO has provided to the OSS a copy/copies of the requested artifact(s) contained in the on-boarded VNF package.

7.7.12 Abort VNF Package deletion operation

7.7.12.1 Description

This operation enables the OSS to abort the deletion of a VNF Package that is in deletion pending state.

Table 7.7.12.1-1 lists the information flow exchanged between the OSS and the NFVO.

Table 7.7.12.1-1: Abort VNF Package deletion operation

Message	Requirement	Direction
AbortPackageDeletionRequest	Mandatory	OSS → NFVO
AbortPackageDeletionResponse	Mandatory	NFVO → OSS

7.7.12.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.12.2-1.

Table 7.7.12.2-1: Abort VNF Package deletion operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfold	M	1	Identifier (Reference to OnboardedVnfPkgInfo)	Identifier of the onboarded VNF Package of which the deletion is requested to be aborted.

7.7.12.3 Output parameters

None.

7.7.12.4 Operation results

After successful operation, an onboarded VNF Package that has been in deletion pending state is in disabled state.

It is an error to invoke the operation for an onboarded VNF Package that is not in deletion pending state.

8 Information elements exchanged

8.1 Introduction

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

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

- String
- Integer
- Identifier
- Filter
- DateTime
- Value
- Rule
- KeyValuePair
- Version
- Binary

8.2 Information elements related to NSD Management

8.2.1 Introduction

The clauses below define information elements related to NSD management.

8.2.2 NsdInfo information element

8.2.2.1 Description

This information element provides the details of an on-boarded NSD.

8.2.2.2 Attributes

The attributes of the NsdInfo information element shall follow the indications provided in table 8.2.2.2-1.

Table 8.2.2.2-1: Attributes of the NsdInfo information element

Attribute	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier	Identifier of the on-boarded instance of the NSD.
nsdId	M	1	Identifier (Reference to Nsd)	Identifier of (reference to) the NSD that is on-boarded.
name	M	1	String	Name of the on-boarded NSD.
version	M	1	Version	Version of the on-boarded NSD.
designer	M	1	String	Designer of the on-boarded NSD .
nsd	M	1	Nsd	NSD details.
onboardedVnfPkgInfoId	M	1..N	Identifier (Reference to OnboardedVnfPkgInfo)	Identifies the OnboardedVnfPkgInfo element for the VNFD referenced by the NSD.
pnfdInfoId	M	0..N	Identifier (Reference to PnfdInfo)	Identifies the PnfdInfo for the PNFD referenced by the NSD.
previousNsdVersionId	M	0..1	Identifier (Reference to NsdInfo)	Reference to the previous version if any of this NSD. This can be recursive if more than one previous version.
operationalState	M	1	OperationalState: Enum {Enabled, Disabled}	Operational state of the on-boarded instance of the NSD.
usageState	M	1	UsageState: Enum {InUse, NotInUse}	Usage state of the on-boarded instance of the NSD.
deletionPending	M	1	Boolean	Indicates if deletion of this instance of the NS descriptor has been requested but the NSD is still being used by instantiated NSs. This instance of the NSD will be deleted once all NSs instantiated from this descriptor are terminated.
userDefinedData	O	0..N	KeyValuePair	User defined data for the NSD.

8.2.3 Pnfd information element

8.2.3.1 Description

This information element provides the details of the PNFD.

8.2.3.2 Attributes

The structure of the Pnfd information element shall comply with the provisions for the Pnfd information element as defined in ETSI GS NFV-IFA 014 [3], clause 6.6.

8.2.4 PnfdInfo information element

8.2.4.1 Description

This information element provides the details of an on-boarded PNFD.

8.2.4.2 Attributes

The PnfdInfo information element shall follow the indications provided in table 8.2.4.2-1.

Table 8.2.4.2-1: Attributes of the PnfdInfo information element

Attribute	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier	Identifier of the on-boarded instance of the PNFD.
pnfdId	M	1	Identifier (Reference to Pnfd)	Identifier of (reference to) the PNFD that is on-boarded.
name	M	1	String	Name of the on-boarded PNFD.
version	M	1	Version	Version of the on-boarded PNFD.
provider	M	1	String	Provider of the on-boarded PNFD.
pnfd	M	1	Pnfd	PNFD details.
previousPnfdVersionId	M	0..1	Identifier (Reference to PnfdInfo)	Reference to the previous version if any of this PNFD. This can be recursive if more than one previous version.
usageState	M	1	UsageState: Enum {InUse, NotInUse}	Usage state of the on-boarded instance of the PNFD.
deletionPending	M	1	Boolean	Indicates if deletion of this instance of the PNFD has been requested but the PNFD is still being used by instantiated NSs. This instance of the PNFD will be deleted once all instantiated NSs using this descriptor are terminated.
userDefinedData	O	0..N	KeyValuePair	User defined data for the PNFD.

8.2.5 Nsd information element

8.2.5.1 Description

This information element provides the details of the NSD.

8.2.5.2 Attributes

The structure of the Nsd information element shall comply with the provisions for the Nsd information element as defined in ETSI GS NFV-IFA 014 [3], clause 6.2.

8.2.6 NsdOnBoardingNotification

8.2.6.1 Description

This notification indicates the on-boarding of an NSD. Support of this notification is mandatory.

8.2.6.2 Trigger Conditions

The notification is produced when:

- New NSD is on-boarded.

8.2.6.3 Attributes

The attributes of the NsdOnBoardingNotification shall follow the indications provided in table 8.2.6.3-1.

Table 8.2.6.3-1: Attributes of the NsdOnBoardingNotification

Attribute	Qualifier	Cardinality	Content	Description
nsdInfold	M	1	Identifier (Reference to NsdInfo)	Identifier of the on-boarded instance of the NSD.
nsdId	M	1	Identifier (Reference to Nsd)	Identifies the NSD being on-boarded.

8.2.7 NsdChangeNotification

8.2.7.1 Description

This notification indicates a change of status in an NSD. Only changes in operational state and deletion pending attribute will be reported, change in usage state is not reported.

Support of this notification is mandatory.

8.2.7.2 Trigger Conditions

The notification is produced when:

- Change of the status (operational state and deletion pending) of an on-boarded NSD
- Deletion of an NSD

8.2.7.3 Attributes

The attributes of the NsdChangeNotification shall follow the indications provided in table 8.2.7.3-1.

Table 8.2.7.3-1: Attributes of the NsdChangeNotification

Attribute	Qualifier	Cardinality	Content	Description
nsdInfold	M	1	Identifier (Reference to NsdInfo)	Identifier of the on-boarded instance of the NSD.
changeType	M	1	Enum	It categorizes the type of change. Possible values can be "change of operational state of an on-boarded NSD", "NSD in deletion pending", and "deletion of an NSD".
operationalState	M	0..1	OperationalState: Enum { Enabled, Disabled}	New operational state of the NSD. Only present when changeType is "change of operational state".
deletionPending	M	0..1	Boolean	Indicates if the deletion instance of the NSD has been requested but the NSD still being used by instantiated NSs. Only present when changeType is "NSD in deletion pending".

8.3 Information elements and notifications related to NS Lifecycle Management

8.3.1 Introduction

The clauses below define information elements and notifications related to network service lifecycle management.

8.3.2 Information elements and notifications related to NS Lifecycle Change Notification

8.3.2.1 Introduction

The clauses below define information elements and notifications related to NS lifecycle change notification.

8.3.2.2 NsLifecycleChangeNotification

8.3.2.2.1 Description

This notification informs the receiver of changes in the NS lifecycle. The support of the notification is mandatory.

8.3.2.2.2 Trigger conditions

This notification is produced when there is a change in the NS lifecycle, including:

- Instantiation of the NS (start and result).
- Scaling of the NS (start and result).
- Update of the NS (start and result).
- Termination of the NS (start and result).
- Healing of the NS (start and result).

8.3.2.2.3 Attributes

The attributes of the NsLifecycleChangeNotification notification shall follow the indications provided in table 8.3.2.2.3-1.

Table 8.3.2.2.3-1: Attributes of the NsLifecycleChangeNotification notification

Attribute	Qualifier	Cardinality	Content	Description
nsInstancelId	M	1	Identifier	Identifier of the NS instance affected.
lifecycleOperationOccurrenceId	M	1	Identifier	Identifier of the NS lifecycle operation occurrence associated to the notification.
operation	M	1	String	The lifecycle operation.
status	M	1	Enum	Indicates whether this notification reports about the start of a lifecycle operation occurrence or the result of a lifecycle operation occurrence.
affectedVnf	M	0..N	AffectedVnf	Information about the VNF instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
affectedPnf	M	0..N	AffectedPnf	Information about the PNF instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.

Attribute	Qualifier	Cardinality	Content	Description
affectedVI	M	0..N	AffectedVirtualLink	Information about the VL instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
affectedVnffg	M	0..N	AffectedVnffg	Information about the VNFFG instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
affectedNs	M	0..N	AffectedNs	Information about the nested NS instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
affectedSap	M	0..N	AffectedSap	Information about the SAP instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
NOTE 1: If this notification represents the result of a lifecycle operation result that was not successful, the notification shall contain appropriate error information.				
NOTE 2: If the notification represents the successful result of a lifecycle operation, at least an affectedVnf, or affectedPnf, or affectedVI, or affectedVnffg or affectedNs, or affectedSap shall be present.				

8.3.2.3 AffectedVnf information element

8.3.2.3.1 Description

This information element provides information about added, deleted and modified VNF instances.

8.3.2.3.2 Attributes

The AffectedVnf information element shall follow the indications provided in table 8.3.2.3.2-1.

Table 8.3.2.3.2-1: Attributes of the AffectedVnf information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.
vnfdId	M	1	Identifier (Reference to Vnfd)	Identifier of the VNFD of the VNF instance.
vnfProfileId	M	1	Identifier (Reference to VnfProfile)	Identifier of the VNF profile of the NSD.
vnfName	M	1	String	Name of the VNF instance.
changeType	M	1	Enum	Signals the type of lifecycle change (instantiated, terminated, scaled, flavour changed, healed, operated, information/configuration modified).

8.3.2.4 AffectedPnf information element

8.3.2.4.1 Description

This information element provides information about added and removed PNFs from an NS.

8.3.2.4.2 Attributes

The AffectedPnf information element shall follow the indications provided in table 8.3.2.4.2-1.

Table 8.3.2.4.2-1: Attributes of the AffectedPnf information element

Attribute	Qualifier	Cardinality	Content	Description
pnfName	M	1	String	Name of the PNF instance.
pnfdId	M	1	Identifier (Reference to Pnfd)	Identifier of the PNFD of the PNF instance.
pnfProfileId	M	1	Identifier (Reference to PnfProfile)	Identifier of the PNF profile of the NSD.
changeType	M	1	Enum	Signals the type of lifecycle change. Permitted values include: added, removed.

8.3.2.5 AffectedVirtualLink information element

8.3.2.5.1 Description

This information element provides information about added, deleted or modified VLs of an NS.

8.3.2.5.2 Attributes

The AffectedVirtualLink information element shall follow the indications provided in table 8.3.2.5.2-1.

Table 8.3.2.5.2-1: Attributes of the AffectedVirtualLink information element

Attribute	Qualifier	Cardinality	Content	Description
nsVirtualLinkId	M	1	Identifier	Identifier of the VL instance.
nsVirtualLinkDescId	M	1	Identifier (Reference to NsVirtualLinkDesc)	Identifier of the VLD in the NSD for this VL.
vlProfileId	M	1	Identifier (Reference to vlProfile)	Identifier of the VL profile of the NSD.
changeType	M	1	Enum	Signals the type of lifecycle change. Permitted values include: added, delete, modified, link port added, link port removed.

8.3.2.6 AffectedVnffg information element

8.3.2.6.1 Description

This information element provides information about added, deleted or modified VNFFG instances.

8.3.2.6.2 Attributes

The AffectedVnffg information element shall follow the indications provided in table 8.3.2.6.2-1.

Table 8.3.2.6.2-1: Attributes of the AffectedVnffg information element

Attribute	Qualifier	Cardinality	Content	Description
vnffgId	M	1	Identifier	Identifier of the VNFFG instance.
vnffgdId	M	1	Identifier (Reference to Vnffgd)	Identifier of the VNFFGD of the VNFFG instance.
changeType	M	1	Enum	Signals the type of lifecycle change (added, removed, modified). See note.

NOTE: CP or NFP information might be modified for the VNFFG.

8.3.2.7 AffectedNs information element

8.3.2.7.1 Description

This information element provides information about added, deleted or modified nested NSs.

8.3.2.7.2 Attributes

The AffectedNs information element shall follow the indications provided in table 8.3.2.7.2-1.

Table 8.3.2.7.2-1: Attributes of the AffectedNs information element

Attribute	Qualifier	Cardinality	Content	Description
nsInstanceld	M	1	Identifier	Identifier of the nested NS instance.
nsdld	M	1	Identifier (reference to Nsd)	Identifier of the NSD of the nested NS instance.
changeType	M	1	Enum	Signals the type of lifecycle change (instantiated, scaled, updated, healed, terminated).

8.3.2.8 AffectedSap information element

8.3.2.8.1 Description

This information element provides information about added, removed or modified SAP of an NS.

8.3.2.8.2 Attributes

The AffectedVirtualLink information element shall follow the indications provided in table 8.3.2.8.2-1.

Table 8.3.2.8.2-1: Attributes of the AffectedSap information element

Attribute	Qualifier	Cardinality	Content	Description
sapInstanceld	M	1	Identifier	Identifier of this SapInfo information element, identifying the SAP instance.
sapdld	M	1	Identifier (Reference to Sapd)	Reference to the SAPD for this SAP.
sapName	M	1	String	Human readable name for the SAP.
changeType	M	1	Enum	Signals the type of lifecycle change. Permitted values include: added, removed, modified.

8.3.2.9 NsIdentifierCreationNotification

8.3.2.9.1 Description

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

8.3.2.9.2 Trigger conditions

- Creation of an NS instance identifier and of the associated instance of an NsInfo information element.

8.3.2.9.3 Attributes

The NsIdentifierCreationNotification shall follow the indications provided in table 8.3.2.9.3-1.

Table 8.3.2.9.3-1: Attributes of the NsIdentifierCreationNotification

Attribute	Qualifier	Cardinality	Content	Description
nsInstancelId	M	1	Identifier	The newly created NS instance identifier.

8.3.2.10 NsIdentifierDeletionNotification

8.3.2.10.1 Description

This notification informs the receiver of the deletion of an NS instance identifier and of the associated instance of an NsInfo information element identified by that identifier. The support of the notification is mandatory.

8.3.2.10.2 Trigger conditions

- Deletion of an NS instance identifier and of the associated instance of an information element.

8.3.2.10.3 Attributes

The NsIdentifierDeletionNotification shall follow the indications provided in table 8.3.2.10.3-1.

Table 8.3.2.10.3-1: Attributes of the NsIdentifierDeletionNotification

Attribute	Qualifier	Cardinality	Content	Description
nsInstancelId	M	1	Identifier	The NS instance identifier to be deleted.

8.3.3 Information elements related to NsInfo

8.3.3.1 Introduction

The clauses below define information elements related to NsInfo.

8.3.3.2 NsInfo information element

8.3.3.2.1 Description

This information element provides run-time information about an NS instance.

8.3.3.2.2 Attributes

The attributes of the NsInfo information element shall follow the indications provided in table 8.3.3.2.2-1.

Table 8.3.3.2.2-1: Attributes of the NsInfo information element

Attribute	Qualifier	Cardinality	Content	Description
nsInstancelId	M	1	Identifier	Identifier of this NsInfo information element, identifying the NS instance.
nsName	M	1	String	Human readable name of the NS instance.
description	M	1	String	Human readable description of the NS instance.
nsdId	M	1	Identifier (Reference to Nsd)	Reference to the NSD used to instantiate this NS.
flavourId	M	0..1	Identifier (Reference to NsDf)	Reference to the flavour of the NSD used to instantiate this NS. See notes 1 and 2.
vnfInfo	M	0..N	VnfInfo	Information on constituent VNFs of this NS. See note 2.

Attribute	Qualifier	Cardinality	Content	Description
pnfInfo	M	0..N	PnfInfo	Information on the PNF(s) that are part of this NS.
virtualLinkInfo	M	0..N	NsVirtualLinkInfo	Information on the VLs of this NS. See note 2.
vnffglInfo	M	0..N	VnffglInfo	Information on the VNFFGs of this NS.
sapInfo	M	0..N	SapInfo	Information on the SAPs of this NS.
nestedNsInfold	M	0..N	Identifier (Reference to NsInfo)	Reference to information on nested NSs of this NS.
nsState	M	1	Enum	The state of the NS. Possible values: NOT_INSTANTIATED (i.e. the NS instance is not instantiated or terminated), INSTANTIATED (i.e. the NS instance is instantiated).
nsScaleStatus	M	0..N	NsScaleInfo	Represents for each NS scaling aspect declared in the applicable DF, how "big" the NS instance has been scaled w.r.t. that aspect.
additionalAffinityOrAntiAffinityRule	M	0..N	AffinityOrAntiAffinityRule	Information on the additional affinity or anti-affinity rule from NS instantiation operation. Shall not conflict with rules already specified in the NSD. See clause 8.3.4.26.
NOTE 1: The NsDf information element is defined in ETSI GS NFV-IFA 014 [3], clause 6.3.2.				
NOTE 2: Cardinality of zero is only valid for a non-instantiated NS.				

8.3.3.3 VnfInfo information element

8.3.3.3.1 Description

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

NOTE: In ETSI GS NFV-MAN 001 [i.2], the concept of the VNF record (VNFR) was introduced which is a model for the totality of information managed by the VNFM regarding a running VNF instance. VNFR is not used in the present document.

8.3.3.3.2 Attributes

The VnfInfo information element shall follow the indications provided in table 8.3.3.3.2-1.

Table 8.3.3.3.2-1: Attributes of the VnfInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance that is represented by this VnfInfo information element.
vnfInstanceName	M	0..1	String	VNF instance name. See note 1.
vnfInstanceDescription	M	0..1	String	Human-readable description of the VNF instance. See note 1.
vnfdId	M	1	Identifier	Identifier of the VNFD on which the VNF instance is based. See notes 2 and 3.
vnfProvider	M	1	String	See note 3.
vnfProductName	M	1	String	See note 3.
vnfSoftwareVersion	M	1	Version	See note 3.

Attribute	Qualifier	Cardinality	Content	Description
vnfdVersion	M	1	Version	See note 3.
onboardedVnfPkgInfold	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF package on which the VNF is based. This identifier was allocated by the NFVO. See notes 1 and 4.
vnfConfigurableProperty	M	0..N	KeyValuePair	Current values of the configurable properties of the VNF instance. Configurable properties referred in this attribute are declared in the VNFD (see clause 7.1.12 in ETSI GS NFV-IFA 011 [2]). They include those set as initial configuration, and/or those that modify a running configuration. See notes 1 and 5.
instantiationState	M	1	Enum	The instantiation state of the VNF. Possible values: NOT_INSTANTIATED (VNF instance is terminated or not instantiated, and the identifier of the VNF instance exists), INSTANTIATED (VNF is instantiated).
instantiatedVnfInfo	M	0..1	InstantiatedVnfInfo	Information specific to an instantiated VNF instance. Shall be present if the VNF is in INSTANTIATED instantiation state.
metadata	M	0..1	KeyValuePair	Additional VNF-specific metadata describing the VNF instance. Metadata that are writeable are declared in the VNFD (see clause 7.1.14.2 in ETSI GS NFV-IFA 011 [2]). See note 1.
extension	M	0..N	KeyValuePair	VNF-specific attributes that affect the lifecycle management of this VNF instance by the VNFM, or the lifecycle management scripts. Extensions that are writeable are declared in the VNFD (see clause 7.1.14.2 in ETSI GS NFV-IFA 011 [2]). See note 1.
<p>NOTE 1: This attribute in the VnfInfo shall be writable through the modifyVnfInfoData attribute of the Update NS operation (refer to clause 7.3.5.2).</p> <p>NOTE 2: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way.</p> <p>NOTE 3: See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package which was used to instantiate the VNF instance.</p> <p>NOTE 4: Modifying the value of this attribute can be performed when no conflicts exist between the previous and the newly referred VNF Package, e.g. when the new VNFD is not changed with respect to the previous VNFD apart from referencing to other VNF software image(s). In order to avoid misalignment of the VnfInfo with the current VNF's on-boarded VNF Package, the values copied from the VNFD of the on-boarded VNF Package (see note 3) need to be kept in sync.</p>				

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

8.3.3.4 InstantiatedVnfInfo information element

8.3.3.4.1 Description

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

Annex A of ETSI GS NFV-IFA 007 [i.5] provides examples illustrating the relationship among the different run-time information elements (CP, VL and link ports) used to represent the connectivity of a VNF.

8.3.3.4.2 Attributes

The InstantiatedVnfInfo information element shall follow the indications provided in table 8.3.3.4.2-1.

Table 8.3.3.4.2-1: Attributes of the InstantiatedVnfInfo information element

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

Attribute	Qualifier	Cardinality	Content	Description
vnfVirtualLinkResourceInfo	M	0..N	VnfVirtualLinkResourceInfo	Information on the virtualised network resource(s) used by the VLs of the VNF instance.
virtualStorageResourceInfo	M	0..N	VirtualStorageResourceInfo	Information on the virtualised storage resource(s) used as storage for the VNF instance.
NOTE 1: The VnfDf information element is defined in ETSI GS NFV-IFA 011 [2], clause 7.1.8.2.				
NOTE 2: For every scaling aspect, together with the information provided by the "maxScaleLevel" attribute of the "ScalingAspect" information element in the VNFD, this allows an external entity to derive how many scaling steps are possible for scaling in or scaling out a VNF instance. Per aspect, the number of steps possible to scale in corresponds to the "scaleLevel" attribute for that aspect in the "scaleStatus" information element, and the possible number of steps to scale out corresponds to the difference between "maxScaleLevel" for that aspect, and the "scaleLevel" attribute for that aspect in the "scaleStatus" information element.				
NOTE 3: The monitoring parameters to be tracked by VNFM are identified by VNF provider in the VNFD. The VNFM collects the values of identified performance metrics using one or more locally initiated PM Jobs.				

8.3.3.5 VnfcResourceInfo information element

8.3.3.5.1 Description

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

8.3.3.5.2 Attributes

The VnfcResourceInfo information element shall follow the indications provided in table 8.3.3.5.2-1.

Table 8.3.3.5.2-1: Attributes of the VnfcResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfcInstanceId	M	1	Identifier	Identifier of this VNFC instance.
vduId	M	1	Identifier (Reference to Vdu)	Reference to the applicable Vdu information element in the VNFD.
computeResource	M	1	ResourceHandle	Reference to the VirtualCompute resource.
storageResource	M	0..N	Identifier (Reference to VirtualStorageResourceInfo)	Reference(s) to the VirtualStorage resource(s).
reservationId	M	0..1	Identifier	The reservation identifier applicable to the resource. It shall be present when an applicable reservation exists.
vnfcCpInfo	M	0..N	VnfcCpInfo	CP(s) of the VNFC instance. Shall be present when that particular CP of the VNFC instance is associated to an external CP of the VNF instance. May be present otherwise.
metadata	M	0..N	KeyValuePair	Metadata about this resource.

8.3.3.6 VnfVirtualLinkResourceInfo information element

8.3.3.6.1 Description

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

8.3.3.6.2 Attributes

The VnfVirtualLinkResourceInfo information element shall follow the indications provided in table 8.3.3.6.2-1.

Table 8.3.3.6.2-1: Attributes of the VnfVirtualLinkResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
virtualLinkInstanceld	M	1	Identifier	Identifier of this VL instance.
vnfVirtualLinkDescId	M	1	Identifier (Reference to VnfVirtualLinkDesc)	Identifier of the VNF VLD in the VNFD.
networkResource	M	1	ResourceHandle	Reference to the VirtualNetwork resource.
reservationId	M	0..1	Identifier	The reservation identifier applicable to the resource. It shall be present when an applicable reservation exists.
vnfLinkPort	M	0..N	VnfLinkPort	Links ports of this VL. Shall be present when the linkPort is used for external connectivity by the VNF (refer to VnfLinkPort in clause 8.3.3.20). May be present otherwise.
metadata	M	0..N	KeyValuePair	Metadata about this resource.

8.3.3.7 VirtualStorageResourceInfo information element

8.3.3.7.1 Description

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

8.3.3.7.2 Attributes

The VirtualStorageResourceInfo information element shall follow the indications provided in table 8.3.3.7.2-1.

Table 8.3.3.7.2-1: Attributes of the VirtualStorageResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
virtualStorageInstanceld	M	1	Identifier	Identifier of this virtual storage resource instance.
virtualStorageDescId	M	1	Identifier (Reference to VirtualStorageDesc)	Identifier of the VirtualStorageDesc in the VNFD.
storageResource	M	1	ResourceHandle	Reference to the VirtualStorage resource(s).
reservationId	M	0..1	Identifier	The reservation identifier applicable to the resource. It shall be present when an applicable reservation exists.
metadata	M	0..N	KeyValuePair	Metadata about this resource.

8.3.3.8 ResourceHandle information element

8.3.3.8.1 Description

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

8.3.3.8.2 Attributes

The ResourceHandle information element shall follow the indications provided in table 8.3.3.8.2-1.

Table 8.3.3.8.2-1: Attributes of the ResourceHandle information element

Attribute	Qualifier	Cardinality	Content	Description
vimId	CM	0..1	Identifier	Identifier of the VIM under whose control this resource is placed. This attribute shall be supported when VNF-related Resource Management in direct mode is applicable.
resourceProviderId	CM	0..1	Identifier	Identifies the entity responsible for the management of the virtualised resource. This attribute shall be supported when VNF-related Resource Management in indirect mode is applicable.
resourceId	M	1	Identifier	Identifier of the resource in the scope of the VIM or the resource provider.
vimLevelResourceType	M	0..1	Not specified	Type of the resource in the scope of the VIM or the resource provider. See note.
NOTE: The value set of the "vimLevelResourceType" attribute is within the scope of the VIM or the resource provider and can be used as information that complements the ResourceHandle.				

8.3.3.9 PnfInfo information element

8.3.3.9.1 Description

This information element provides information about a PNF that is part of an NS instance.

8.3.3.9.2 Attributes

The attributes of the PnfInfo information element shall follow the indications provided in table 8.3.3.9.2-1.

Table 8.3.3.9.2-1: Attributes of the PnfInfo information element

Attribute	Qualifier	Cardinality	Content	Description
pnfName	M	1	String	Name of the PNF.
pnfdInfold	M	1	Identifier (Reference to PnfInfo)	Identifier of (reference to) the PNFD information related to this PNF.
cpInfo	M	1..N	PnfExtCpInfo	Information on the external CP of the PNF.

8.3.3.10 NsVirtualLinkInfo information element

8.3.3.10.1 Description

This information element provides run-time information about an NS VL instance.

As an NS can include NFs deployed in NFVI PoPs under the control of several different VIMs, deploying an NS VL can involve several VIMs each allocating different virtualised network resources.

NOTE: The connectivity between virtualised network resources allocated in different VIMs and part of the same VL is not addressed in the present document.

8.3.3.10.2 Attributes

The attributes of the NsVirtualLinkInfo information element shall follow the indications provided in table 8.3.3.10.2-1.

Table 8.3.3.10.2-1: Attributes of the NsVirtualLink information element

Attribute	Qualifier	Cardinality	Content	Description
nsVirtualLinkInstanceId	M	1	Identifier	Identifier of this NsVirtualLinkInfo information element, identifying the NS VL instance.
nsVirtualLinkDescId	M	1	Identifier (Reference to NsVirtualLinkDesc)	Identifier of the VLD in the NSD for this VL.
resourceHandle	M	1..N	ResourceHandle	Identifier(s) of the virtualised network resource(s) realizing this VL.
linkPort	M	0..N	NsLinkPort	Link ports of this VL. Cardinality of zero indicates that no port have yet been created for this VL.

8.3.3.11 NsLinkPort information element

8.3.3.11.1 Description

This information element provides information about a port of an NS VL.

8.3.3.11.2 Attributes

The attributes of the NsLinkPort information element shall follow the indications provided in table 8.3.3.11.2-1.

Table 8.3.3.11.2-1: Attributes of the NsLinkPort information element

Attribute	Qualifier	Cardinality	Content	Description
resourceHandle	M	1	ResourceHandle	Identifier(s) of the virtualised network resource(s) realizing this link port.
cpId	M	0..1	Identifier (Reference to VnfExtCplInfo or PnfExtCplInfo or SapInfo)	CP connected to this link port.

8.3.3.12 SapInfo information element

8.3.3.12.1 Description

This information element provides information about an SAP of an NS instance.

8.3.3.12.2 Attributes

The attributes of the SapInfo information element shall follow the indications provided in table 8.3.3.12.2-1.

Table 8.3.3.12.2-1: Attributes of the SapInfo information element

Attribute	Qualifier	Cardinality	Content	Description
sapInstanceld	M	1	Identifier	Identifier of this SapInfo information element, identifying the SAP instance.
sapdld	M	1	Identifier (Reference to Sapd)	Reference to the SAPD for this SAP.
sapName	M	1	String	Human readable name for the SAP.
description	M	1	String	Human readable description for the SAP.
address	M	1	Not specified.	Address for this SAP. In some cases, the NFVO provides the address. See note.
NOTE: The address information shall be compatible with the type and sub-type attribute of the SAPD. In case of an IP address, a port number may be included.				

8.3.3.13 VnffgInfo information element

8.3.3.13.1 Description

This information element contains information about a VNFFFG instance.

8.3.3.13.2 Attributes

The attributes of the VnffgInfo information element shall follow the indications provided in table 8.3.3.13.2-1.

Table 8.3.3.13.2-1: Attributes of the VnffgInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnffgld	M	1	Identifier	Identifier of the Vnffg information element.
vnffgdld	M	1	Identifier (Reference to Vnffgd)	Identifier of the VNFFGD used to instantiate this VNFFFG.
vnfld	M	1..N	Identifier (Reference to VnflInfo)	Identifier(s) of the constituent VNF instance(s) of the VNFFFG.
pnfld	M	0..N	Identifier (Reference to PnflInfo)	Identifier(s) of the constituent PNF instance(s) of the VNFFFG.
virtualLinkld	M	1..N	Identifier (Reference to NsVirtualLink)	Identifier(s) of the constituent VL instance(s) of the VNFFFG.
cpld	M	1..N	Identifier (Reference to VnfExtCpInfo or PnfExtCpInfo or SapInfo)	Identifiers of the CP instances attached to the constituent VNFs and PNFs or the sap instances of the VNFFFG (see note).
nfp	M	1..N	Nfp	Information on the NFPs of this VNFFFG.
NOTE: It indicates an exhaustive list of all the CP instances and SAP instances of the VNFFFG.				

8.3.3.14 PnfExtCpInfo information element

8.3.3.14.1 Description

This information element provides information about the external CP of the PNF.

8.3.3.14.2 Attributes

The attributes of the PnfExtCpInfo information element shall follow the indications provided in table 8.3.3.14.2-1.

Table 8.3.3.14.2-1: Attributes of the PnfExtCplInfo information element

Attribute	Qualifier	Cardinality	Content	Description
cpId	M	1	Identifier (Reference to Cpd)	Identifier of (reference to) the Connection Point Descriptor (CPD) for this CP.
address	M	1	Not specified.	Address for this CP.
NOTE: The address information shall be compatible with the type and sub-type attribute of the CPD. In case of an IP address, a port number may be included.				

8.3.3.15 Nfp information element

8.3.3.15.1 Description

The Nfp information element defines the information related to the NFP.

8.3.3.15.2 Attributes

The attributes of the Nfp information element shall follow the indications provided in table 8.3.3.15.2-1.

Table 8.3.3.15.2-1: Attributes of the Nfp information element

Attribute	Qualifier	Cardinality	Content	Description
nfpId	M	1	Identifier	Identifier of this Nfp information element.
cpId	M	1..N	Identifier (Reference to VnfExtCplInfo or PnfExtCplInfo or SapInfo)	Identifier(s) of the CPs and/or SAPs which the NFP passes by. See note.
totalCp	O	0..1	Integer	Total number of CPs in this NFP.
nfpRule	M	1	Rule	NFP classification and selection rule.
nfpState	M	1	Enum	An indication of whether the NFP is enabled or disabled.
NOTE: When multiple identifiers are included, the position of the identifier in the information element value specifies the position of the CP or the SAP in the path.				

8.3.3.16 NsScaleInfo information element

8.3.3.16.1 Description

8.3.3.16.2 Attributes

The attributes of the NsScaleInfo information element shall follow the indications provided in table 8.3.3.16.2-1.

Table 8.3.3.16.2-1: Attributes of the NsScaleInfo information element

Attribute	Qualifier	Cardinality	Content	Description
nsScalingAspectId	M	1	Identifier (Reference to NsScalingAspect)	Identifier of the NS scaling aspect.
nsScaleLevelId	M	1	Identifier (Reference to NsLevel)	Identifier of the NS scale level.

8.3.3.17 VnfExtCplInfo information element

8.3.3.17.1 Description

This information element provides information related to an external CP.

8.3.4.17.2 Attributes

The VnfExtCpInfo information element shall follow the indications provided in table 8.3.3.17.2-1.

Table 8.3.3.17.2-1: Attributes of the VnfExtCpInfo information element

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

8.3.3.18 ExtVirtualLinkInfo information element

8.3.3.18.1 Description

This information element provides a reference to an external VL.

8.3.3.18.2 Attributes

The ExtVirtualLinkInfo information element shall follow the indications provided in table 8.3.3.18.2-1.

Table 8.3.3.18.2-1: Attributes of the ExtVirtualLinkInfo information element

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

8.3.3.19 ExtManagedVirtualLinkInfo information element

8.3.3.19.1 Description

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

8.3.3.19.2 Attributes

The ExtManagedVirtualLinkInfo information element shall follow the indications provided in table 8.3.3.19.2-1.

Table 8.3.3.19.2-1: Attributes of the ExtManagedVirtualLinkInfo information element

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

8.3.3.20 VnfLinkPort information element

8.3.3.20.1 Description

This information element provides information about a port of a VNF's internal VL. See also VnfVirtualLinkResourceInfo in clause 8.3.3.6.

8.3.3.20.2 Attributes

The attributes of the VnfLinkPort information element shall follow the indications provided in table 8.3.3.20.2-1.

Table 8.3.3.20.2-1: Attributes of the VnfLinkPort information element

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

8.3.3.21 ScaleInfo information element

8.3.3.21.1 Description

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

8.3.3.21.2 Attributes

The ScaleInfo information element shall follow the indications provided in table 8.3.3.21.2-1.

Table 8.3.3.21.2-1: Attributes of the ScaleInfo information element

Attribute	Qualifier	Cardinality	Content	Description
aspectId	M	1	Identifier (Reference to ScalingAspect)	Reference to the scaling aspect.
scaleLevel	M	1	Integer	The scale level for that aspect. Minimum value 0, maximum value maxScaleLevel as declared in the VNFD (see ETSI GS NFV-IFA 011 [2], clause 7.1.10.2.2).

8.3.3.22 ExtLinkPort information element

8.3.3.22.1 Description

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

8.3.3.22.2 Attributes

The attributes of the ExtLinkPort information element shall follow the indications provided in table 8.3.3.22.2-1.

Table 8.3.3.22.2-1: Attributes of the ExtLinkPort information element

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

8.3.3.23 VnfcCplInfo information element

8.3.3.23.1 Description

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

8.3.3.23.2 Attributes

The VnfcCplInfo information element shall follow the indications provided in table 8.3.3.23.2-1.

Table 8.3.3.23.2-1: Attributes of the VnfcCplInfo information element

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

8.3.4 Information elements related to NS Lifecycle Management operations

8.3.4.1 Introduction

The clauses below define information elements related to network service lifecycle management operations.

8.3.4.2 SapData information element

8.3.4.2.1 Description

The SapData information element defines information related to a SAP of an NS.

8.3.4.2.2 Attributes

The attributes of the SapData information element shall follow the indications provided in table 8.3.4.2.2-1.

Table 8.3.4.2.2-1: Attributes of the SapData information element

Attribute	Qualifier	Cardinality	Content	Description
sapId	M	1	Identifier (Reference to Sapd)	Reference to the SAPD for this SAP.
sapName	M	1	String	Human readable name for the SAP.
description	M	1	String	Human readable description for the SAP.
address	M	0..1	Not specified.	Address for this SAP. In some cases, the NFVO provides the address (refer to attribute sapAddressAssignment of Sapd information element in ETSI GS NFV-IFA 014 [3], clause 6.2.3.2). See note.
NOTE: The address information shall be compatible with the type and sub-type attribute of the SAPD. In case of an IP address, a port number may be included.				

8.3.4.3 VnfInstanceData information element

8.3.4.3.1 Description

The VnfInstanceData specifies an existing VNF instances to be used in the NS instance and if needed, the VNF Profile to use for this VNF instance.

8.3.4.3.2 Attributes

The attributes of the VnfInstanceData information element shall follow the indications provided in table 8.3.4.3.2-1.

Table 8.3.4.3.2-1: Attributes of the VnfInstanceData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceid	M	1	Identifier	Identifier of the existing VNF instance to be used in the NS.
vnfProfileid	M	0..1	Identifier (Reference to VnfProfile)	Identifier of (Reference to) a vnfProfile defined in the NSD which the existing VNF instance shall be matched with. If not presents, the NFVO will select the VnfProfile matching the information in the VNF instance.

8.3.4.4 VnfLocationConstraint information element

8.3.4.4.1 Description

The VnfLocationConstraint information element defines the location constraints for the VNF to be instantiated.

8.3.4.4.2 Attributes

The attributes of the VnfLocationConstraint information element shall follow the indications provided in table 8.3.4.4.2-1.

Table 8.3.4.4.2-1: Attributes of the VnfLocationConstraint information element

Attribute	Qualifier	Cardinality	Content	Description
vnfProfileid	M	1	Identifier (Reference to VnfProfile)	Identifier (reference to) of a VnfProfile in the NSD used to manage the lifecycle of the VNF instance.
locationConstraints	M	1	Not specified.	Defines the location constraints for the VNF instance to be created.

8.3.4.5 ParamsForVnf information element

8.3.4.5.1 Description

The ParamsForVnf specifies additional parameters for an NS instance on a per VNF instance basis.

8.3.4.5.2 Attributes

The attributes of the ParamsForVnf information element shall follow the indications provided in table 8.3.4.5.2-1.

Table 8.3.4.5.2-1: Attributes of the ParamsForVnf information element

Attribute	Qualifier	Cardinality	Content	Description
vnfProfileId	M	1	Identifier (Reference to VnfProfile)	Identifier of (reference to) a vnfProfile to which the additional parameters apply.
additionalParam	M	0..N	KeyValuePair	Additional parameters that are to be applied per VNF instance.

8.3.4.6 ScaleNsData information element

8.3.4.6.1 Description

The ScaleNsData information element describes the information needed to scale an NS instance either by explicitly adding/removing existing VNF instances or by leveraging on the abstraction mechanism provided by the NS scaling aspects and NS levels information elements declared in the NSD.

8.3.4.6.2 Attributes

The attributes of the ScaleNsData information element shall follow the indications provided in table 8.3.4.6.2-1.

Table 8.3.4.6.2-1: Attributes of the ScaleNsData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceToBeAdded	M	0..N	VnfInstanceData	Specifies an existing VNF instance to be added to the NS instance as part of the scaling operation. If needed, the VNF Profile to be used for this VNF instance is also provided. See notes 1, 2 and 3.
vnfInstanceToBeRemoved	M	0..N	Identifier	Specifies a VNF instance to be removed from the NS instance as part of the scaling operation. See notes 1 and 4.
scaleNsByStepsData	M	0..1	ScaleNsByStepsData	Specifies the information needed to scale an NS instance by one or more scaling steps. See note 1.
scaleNsToLevelData	M	0..1	ScaleNsToLevelData	Specifies the information needed to scale an NS instance to a target size. See note 1.
additionalParamForNs	M	0..N	KeyValuePair	Allows the OSS/BSS to provide additional parameter(s) at the NS level necessary for the NS scaling (as opposed to the VNF level, which is covered in additionalParamForVnf).

Attribute	Qualifier	Cardinality	Content	Description
additionalParamForVnf	M	0..N	ParamsForVnf	Allows the OSS/BSS to provide additional parameter(s) per VNF instance (as opposed to the NS level, which is covered in additionalParamforNs). This is for VNFs that are to be created by the NFVO as part of the NS scaling and not for existing VNF that are covered by the scaleVnfData.
locationConstraints	M	0..N	VnfLocationConstraint	Defines the location constraints for the VNF to be instantiated as part of the NS scaling. An example can be a constraint for the VNF to be in a specific geographic location.
<p>NOTE 1: No more than two attributes between vnfInstanceToBeAdded, vnfInstanceToBeRemoved, scaleNsByStepsData and scaleNsToLevelData shall be present. In case of two, the attributes shall be vnfInstanceToBeAdded and vnfInstanceToBeRemoved.</p> <p>NOTE 2: The DF of the VNF instance shall match the VNF DF present in the associated VNF Profile of the new NS flavour.</p> <p>NOTE 3: This functionality is the same as the one provided by the Update NS operation when the AddVnf update type is selected (see clause 7.3.5).</p> <p>NOTE 4: This functionality is the same as the one provided by the Update NS operation when the RemoveVnf update type is selected (see clause 7.3.5).</p>				

8.3.4.7 ScaleNsByStepsData information element

8.3.4.7.1 Description

The ScaleNsByStepsData information element describes the information needed to scale an NS instance by one or more scaling steps, with respect to a particular NS scaling aspect. Performing a scaling step means increasing/decreasing the capacity of an NS instance in a discrete manner, i.e. moving from one NS scale level to another. The NS scaling aspects and their corresponding NS scale levels applicable to the NS instance are declared in the NSD.

8.3.4.7.2 Attributes

The attributes of the ScaleNsByStepsData information element shall follow the indications provided in table 8.3.4.7.2-1.

Table 8.3.4.7.2-1: Attributes of the ScaleNsByStepsData information element

Attribute	Qualifier	Cardinality	Content	Description
scalingDirection	M	1	Enum	Specifies the scaling direction. Possible values: SCALE_IN, SCALE_OUT.
aspectId	M	1	Identifier (Reference to NsScalingAspect)	Provides the aspect of the NS that is requested to be scaled, as declared in the NSD.
numberOfSteps	M	0..1	Integer	Specifies the number of scaling steps to be performed. Defaults to 1.

8.3.4.8 ScaleNsToLevelData information element

8.3.4.8.1 Description

The ScaleNsToLevelData information element describes the information needed to scale an NS instance to a target size. The target size is either expressed as an NS instantiation level or as a list of NS scale levels, one per NS scaling aspect, of the current DF. The NS instantiation levels, the NS scaling aspects and their corresponding NS scale levels applicable to the NS instance are declared in the NSD.

8.3.4.8.2 Attributes

The attributes of the ScaleNsToLevelData information element shall follow the indications provided in table 8.3.4.8.2-1.

Table 8.3.4.8.2-1: Attributes of the ScaleNsToLevelData information element

Attribute	Qualifier	Cardinality	Content	Description
nsInstantiationLevel	M	0..1	Identifier (Reference to NsLevel)	Identifier of the target NS instantiation level of the current DF to which the NS instance is requested to be scaled. See note.
nsScaleInfo	M	0..N	NsScaleInfo	For each NS scaling aspect of the current DF, defines the target NS scale level to which the NS instance is to be scaled. See note.
NOTE: Either nsInstantiationLevel or nsScaleInfo, but not both, shall be present.				

8.3.4.9 ScaleVnfData information element

8.3.4.9.1 Description

This information element describes the information needed, either to scale a VNF instance to a given level, or to scale a VNF instance by steps.

8.3.4.9.2 Attributes

The attributes of the ScaleVnfData information element shall follow the indications provided in table 8.3.4.9.2-1.

Table 8.3.4.9.2-1: Attributes of the ScaleVnfData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance being scaled.
type	M	1	Enum	Defines the type of the scale VNF operation requested (scale out, scale in, scale to instantiation level, scale to scale level(s)). The set of types actually supported depends on the capabilities of the VNF being managed. See note 1.
scaleToLevelData	M	0..1	ScaleToLevelData	Provides the information needed when scaling to a given level. See note 2.
scaleByStepData	M	0..1	ScaleByStepData	Provides the information needed when scaling by steps. See note 2.
NOTE 1: ETSI GS NFV-IFA 010 [1] specifies that the lifecycle management operations that expand or contract a VNF instance include scale in, scale out, scale up and scale down. Vertical scaling (scale up, scale down) is not supported in the present document.				
NOTE 2: Either scaleToLevelData or scaleByStepData but not both shall be present. The scaleByStepData is used for scale out/in type of scaling, and the scaleToLevelData is used for scale to instantiation/scale level type of scaling.				

8.3.4.10 ScaleToLevelData information element

8.3.4.10.1 Description

The ScaleToLevelData information element describes the information needed to scale a VNF instance to a target size. The target size is either expressed as an instantiation level of that DF as defined in the VNFD, or given as a list of scale levels, one per scaling aspect of that DF. Instantiation levels and scaling aspects are declared in the VNFD. The NFVO shall then invoke the ScaleVnfToLevel operation towards the appropriate VNF. The specific parameters passed by the NFVO to the VNF are specified in clause 7.2.5.2 of ETSI GS NFV-IFA 007 [i.5].

8.3.4.10.2 Attributes

The attributes of the ScaleToLevelData information element shall follow the indications provided in table 8.3.4.10.2-1.

Table 8.3.4.10.2-1: Attributes of the ScaleToLevelData information element

Attribute	Qualifier	Cardinality	Content	Description
instantiationLevelId	M	0..1	Identifier (Reference to InstantiationLevel)	Identifier of (reference to) the target instantiation level of the current DF to which the VNF instance is requested to be scaled. See note.
scaleInfo	M	0..N	ScaleInfo	For each scaling aspect of the current DF, defines the target scale level to which the VNF instance is to be scaled. The VNF provider defines in the VNFD whether or not a particular VNF supports scaling according to this parameter. Such a property in the VNFD applies for all instances of a particular VNF. See note.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the scaling process, specific to the VNF instance being scaled, as declared in the VNFD (see clause 7.1.5.5 in ETSI GS NFV-IFA 011 [2]).
NOTE: Either instantiationLevelId or scaleInfo but not both shall be present.				

8.3.4.11 ScaleByStepData information element

8.3.4.11.1 Description

The ScaleByStepData information element describes the information needed to scale a VNF instance by steps. The NFVO shall then invoke the ScaleVNF operation towards the appropriate VNFM. The specific parameters passed by the NFVO to the VNFM are specified in clause 7.2.4.2 of ETSI GS NFV-IFA 007 [i.5].

8.3.4.11.2 Attributes

The attributes of the ScaleByStepData information element shall follow the indications provided in table 8.3.4.11.2-1.

Table 8.3.4.11.2-1: Attributes of the ScaleByStepData information element

Attribute	Qualifier	Cardinality	Content	Description
aspectId	M	1	Identifier (Reference to ScalingAspect)	Identifier of (reference to) the aspect of the VNF that is requested to be scaled, as declared in the VNFD.
numberOfSteps	M	0..1	Integer	Number of scaling steps. It shall be a positive number. Defaults to 1. The VNF provider defines in the VNFD whether or not a particular VNF supports performing more than one step at a time. Such a property in the VNFD applies for all instances of a particular VNF. See note.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the scaling process, specific to the VNF instance being scaled, as declared in the VNFD (see clause 7.1.5.4 in ETSI GS NFV-IFA 011 [2]).
NOTE: A scaling step is the smallest unit by which a VNF instance can be scaled w.r.t a particular scaling aspect.				

8.3.4.12 InstantiateVnfData information element

8.3.4.12.1 Description

The InstantiateVnfData information element specifies the parameters that are needed for VNF instantiation. This information element is used for the bottom-up NS creation when the OSS/BSS explicitly requests VNF instantiation for a given NS. When the NFVO invokes the Instantiate VNF operation a set of these parameters are then passed by the NFVO to the VNFM. The specific parameters passed by the NFVO to the VNFM are specified in clause 7.2.3.2 of ETSI GS NFV-IFA 007 [i.5].

8.3.4.12.2 Attributes

The attributes of the InstantiateVnfData information element shall follow the indications provided in table 8.3.4.12.2-1.

Table 8.3.4.12.2-1: Attributes of the InstantiateVnfData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfdId	M	1	Identifier (Reference to Vnfd)	Information sufficient to identify the VNFD which defines the VNF to be instantiated.
flavourId	M	1	Identifier (Reference to VnfDf)	Identifier of the VNF DF to be instantiated. See note 1.
instantiationLevelId	M	0..1	Identifier (Reference to InstantiationLevel)	Identifier of the instantiation level of the DF to be instantiated. If not present, the default instantiation level as declared in the VNFD shall be instantiated.
vnfInstanceName	M	0..1	String	Human-readable name of the VNF instance to be created.
vnfInstanceDescription	M	0..1	String	Human-readable description of the VNF instance to be created.
extVirtualLink	M	0..N	ExtVirtualLinkData	Information about external VLS to connect the VNF to.
extManagedVirtualLink	M	0..N	ExtManagedVirtualLinkData	Information about internal VLS that are managed by other entities than the VNFM. See note 2.
localizationLanguage	M	0..1	Not specified.	Localization language of the VNF to be instantiated. The localization languages supported by a VNF can be declared in the VNFD. If this parameter is not provided and the "defaultLocalizationLanguage" attribute is declared in the VNFD, the "defaultLocalizationLanguage" shall be used to determine the localization language VNF to be instantiated.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the instantiation process, specific to the VNF being instantiated, as declared in the VNFD (see clause 7.1.5.3 in ETSI GS NFV-IFA 011 [2]).
locationConstraint	M	0..1	VnfLocationConstraint	Defines the location constraints for the VNF to be instantiated as part of the NS instantiation. An example can be a constraint for the VNF to be in a specific geographic location.
NOTE 1: The VnfDf information element is defined in ETSI GS NFV-IFA 011 [2], clause 7.1.8.2.				
NOTE 2: The indication of externally-managed internal VLS is needed in case networks have been pre-configured for use with certain VNFs, for instance to ensure that these networks have certain properties such as security or acceleration features, or to address particular network topologies.				

8.3.4.13 ExtVirtualLinkData information element

8.3.4.13.1 Description

This information element provides the information of an external VL to be used as a parameter passed to NS lifecycle management interface.

8.3.4.13.2 Attributes

The ExtVirtualLinkData information element shall follow the indications provided in table 8.3.4.13.2-1.

Table 8.3.4.13.2-1: Attributes of the ExtVirtualLinkData information element

Attribute	Qualifier	Cardinality	Content	Description
extVirtualLinkId	M	1	Identifier	Identifier of this external VL instance.
vimId	CM	0..1	Identifier	Identifier of the VIM that manages this resource. This attribute shall be supported and present if VNF-related resource management in direct mode is applicable.
resourceProviderId	CM	0..1	Identifier	Identifies the entity responsible for the management of the resource. This attribute shall be supported and present when VNF-related Resource Management in indirect mode is applicable.
resourceId	M	1	Identifier	Identifier of the resource in the scope of the VIM or the resource provider.
extCp	M	1..N	VnfExtCpData	External CPs of the VNF to be connected to this VL.

8.3.4.14 VnfExtCpData information element

8.3.4.14.1 Description

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

8.3.4.14.2 Attributes

The VnfExtCpData information element shall follow the indications provided in table 8.3.4.14.2-1.

Table 8.3.4.14.2-1: Attributes of the VnfExtCpData information element

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

8.3.4.15 ChangeVnfFlavourData information element

8.3.4.15.1 Description

The ChangeVnfFlavourData specifies existing VNF instance for which the DF needs to be changed. This specifies the new DF, the instantiationLevel of the new DF that may be used and the additional parameters as input for the flavour change.

The change of VNF DF depends on VNF capabilities and its support by the VNF is declared in the VNFD.

8.3.4.15.2 Attributes

The attributes of the ChangeVnfFlavourData information element shall follow the indications provided in table 8.3.4.15.2-1.

Table 8.3.4.15.2-1: Attributes of the ChangeVnfFlavourData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance to be modified.
newFlavourId	M	1	Identifier (Reference to VnfDeployemtnFlavour)	Identifier of the new VNF DF to apply to this VNF instance.
instantiationLevelId	M	0..1	Identifier (Reference to InstantiationLevel)	Identifier of the instantiation level of the DF to be used. If not present, the default instantiation level as declared in the VNFD shall be used.
extVirtualLink	M	0..N	ExtVirtualLinkData	Information about external VLs to connect the VNF to.
extManagedVirtualLink	M	0..N	ExtManagedVirtualLinkData	Information about internal VLs that are managed by other entities than the VNFM. See note.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the flavour change process, specific to the VNF being modified, as declared in the VNFD (see clause 7.1.5.9 in ETSI GS NFV-IFA 011 [2]).
NOTE: The indication of externally-managed internal VLs is needed in case networks have been pre-configured for use with certain VNFs, for instance to ensure that these networks have certain properties such as security or acceleration features, or to address particular network topologies.				

8.3.4.16 OperateVnfData information element

8.3.4.16.1 Description

The OperateVnfData information element specifies the VNF instance for which the operational state needs to be changed and the requested new state.

8.3.4.16.2 Attributes

The attributes of the OperateVnfData information element shall follow the indications provided in table 8.3.4.16.2-1.

Table 8.3.4.16.2-1: Attributes of the OperateVnfData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.
changeStateTo	M	1	Enum	The desired state to change the VNF to. Allowed values are: start, stop.
stopType	M	0..1	Enum	It signals whether forceful or graceful stop is requested. Allowed values are: forceful and graceful. In case of forceful stop, the VNF is stopped immediately. Note that if the VNF is still in service, this may adversely impact network service, and therefore, operator policies apply to determine if forceful stop is allowed in the particular situation. In case of graceful stop, the VNFM first arranges to take the VNF out of service (by means out of scope of the present specification, e.g. involving interaction with EM, if required). Once this is successful, or after a timeout, the VNFM stops the VNF. Only applicable when changing state to stop.
gracefulStopTimeout	M	0..1	TimeDuration	The time interval to wait for the VNF to be taken out of service during graceful stop, before stopping the VNF. If not given, it is expected that the VNFM waits for the successful taking out of service of the VNF, no matter how long it takes, before stopping the VNF. See note. Minimum timeout or timeout range are specified by the VNF vendor (e.g. defined in the VNFD or communicated by other means). The parameter is not relevant in case of forceful stop.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the Operate VNF operation, specific to the VNF being operated, as declared in the VNFD (see clause 7.1.5.8 in ETSI GS NFV-IFA 011 [2]).
NOTE: This implies that no VNF stop will be attempted if taking the VNF out of service fails or hangs.				

8.3.4.17 ModifyVnfInfoData information element

8.3.4.17.1 Description

The ModifyVnfInfoData information element specifies for a VNF instance the information that is requested to be modified. The information to be modified shall comply with the associated NSD.EXAMPLE: When the onboardedVnfPkgInfoId attribute value of VnfInfo needs to be updated, the value would need to match the identifier's value of an OnboardedVnfPkgInfo whose vnfId is present in the associated VNF type and profile of the NSD.

8.3.4.17.2 Attributes

The attributes of the ModifyVnfInfoData information element shall follow the indications provided in table 8.3.4.17.2-1.

Table 8.3.4.17.2-1: Attributes of the ModifyVnfInfoData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance for which the writable attributes of VnfInfo are requested to be modified.
newValues	M	1..N	KeyValuePair	Contains the set of attributes to update. The key in the KeyValuePair indicates the name of an attribute that is writable through the interface whose value is to be updated. The value in the KeyValuePair indicates the new attribute value.

8.3.4.18 Void

8.3.4.19 AssocNewNsdVersionData information element

8.3.4.19.1 Description

The AssocNewNsdVersionData information element specifies a new NSD version that is associated to the NS instance. After issuing the Update NS operation with updateType=AssocNewNsdVersion, the NFVO shall use the referred NSD as a basis for the given NS instance.

Different versions of the same NSD have same nsdInvariantId, but different nsdId attributes, therefore if the nsdInvariantId of the NSD version that is to be associated to this NS instance is different from the one used before, the NFVO shall reject the request. Only new versions of the same NSD can be associated to an existing NS instance.

8.3.4.19.2 Attributes

The attributes of the AssocNewNsdVersionData information element shall follow the indications provided in table 8.3.4.19.2-1.

Table 8.3.4.19.2-1: Attributes of the AssocNewNsdVersionData information element

Attribute	Qualifier	Cardinality	Content	Description
newNsdId	M	1	Identifier (reference to Nsd)	Identifier of the new NSD version that is to be associated to the NS instance.
sync	M	0..1	Boolean	Specify whether the NS instance should be automatically synchronized to the new NSD by the NFVO (in case of true value) or the NFVO should not do any action (in case of a false value) and wait for further guidance from OSS/BSS (i.e. waiting for OSS/BSS to issue NS lifecycle management operation to explicitly add/remove VNFs and modify information of VNF instances according to the new NSD). The synchronization to the new NSD means e.g. instantiating/adding those VNFs whose VNFD is referenced by the new NSD version but not referenced by the old one, terminating/removing those VNFs whose VNFD is referenced by the old NSD version but not referenced by the new NSD version, modifying information of VNF instances to the new applicable VNFD provided in the new NSD version.

8.3.4.20 MoveVnfInstanceData information element

8.3.4.20.1 Description

The MoveVnfInstanceData specifies existing VNF instances that needs to be moved from one NS instance (source) to another NS instance (destination). The NS instance defined in the Update NS operation (refer to nsInstanceId in

table 7.3.5.2-1) indicates the source NS instance and the destination NS instance is specified in the present information element (refer to targetNsInstanceId in table 8.3.4.16.2-1).

8.3.4.20.2 Attributes

The attributes of the MoveVnfInstanceData information element shall follow the indications provided in table 8.3.4.20.2-1.

Table 8.3.4.20.2-1: Attributes of the MoveVnfInstanceData information element

Attribute	Qualifier	Cardinality	Content	Description
targetNsInstanceId	M	1	Identifier	Specify the target NS instance where the VNF instances are moved to.
vnfInstanceId	M	1..N	Identifier	Specify the VNF instance that is moved.

8.3.4.21 AddVnffgData information element

8.3.4.21.1 Description

This information element specifies the parameters that are needed for the creation of a new VNFFG instance.

8.3.4.21.2 Attributes

The attributes of the AddVnffgData information element shall follow the indications provided in table 8.3.4.21.2-1.

Table 8.3.4.21.2-1: Attributes of the AddVnffgData information element

Attribute	Qualifier	Cardinality	Content	Description
vnffgdId	M	1	Identifier(Reference to Vnffgd)	Identifier of the VNFFGD which defines the VNFFG to be added.
vnffgName	M	1	String	Human readable name for the VNFFG.
description	M	1	String	Human readable description for the VNFFG.

8.3.4.22 UpdateVnffgData information element

8.3.4.22.1 Description

This information element specifies the parameters needed for the update of an existing VNFFG instance.

8.3.4.22.2 Attributes

The attributes of the UpdateVnffgData information element shall follow the indications provided in table 8.3.4.22.2-1.

Table 8.3.4.22.2-1: Attributes of the UpdateVnffgData information element

Attribute	Qualifier	Cardinality	Content	Description
vnffgdId	M	1	Identifier (Reference to VnffgInfo)	Identifier of an existing VNFFG information element to be updated for the NS Instance.
nfp	M	0..N	NfpData	Indicate the desired new NFP(s) for a given VNFFG after the operations of addition/removal of NS components (e.g. VNFs, VLs, etc.) have been completed, or indicate the updated or newly created NFP classification and selection rule which applied to an existing NFP.
nfpId	M	0..N	Identifier (Reference to Nfp)	Identifier(s) of the NFP to be deleted from a given VNFFG.

8.3.4.23 NfpData information element

8.3.4.23.1 Description

This information element contains information needed to create or modify an NFP instance.

8.3.4.23.2 Attributes

The attributes of the NfpData information element shall follow the indications provided in table 8.3.4.23.2-1.

Table 8.3.4.23.2-1: Attributes of the NfpData information element

Attribute	Qualifier	Cardinality	Content	Description
nfpId	M	0..1	Identifier	Identifier of the NFP to be modified. See note 1.
nfpName	M	0..1	String	Human readable name for the NFP. See note 2.
description	M	0..1	String	Human readable description for the NFP. See note 2.
cp	M	0..N	Identifier (Reference to VnfExtCplInfo or PnfExtCplInfo or SapInfo)	Identifier(s) of the CPs and SAPs which the NFP passes by. Cardinality can be 0 if only updated or newly created NFP classification and selection rule which applied to an existing NFP is provided. See notes 3 and 4.
nfpRule	M	0..1	Rule	NFP classification and selection rule. See note 3.
NOTE 1: It shall be present for modified NFPs and shall be absent for the new NFP. NOTE 2: It shall be present for the new NFP, and it may be present otherwise. NOTE 3: At least a CP or an nfpRule shall be present. NOTE 4: When multiple identifiers are included, the position of the identifier in the information element value specifies the position of the CP or the SAP in the path.				

8.3.4.24 HealNsData information element

8.3.4.24.1 Description

This information element describes the information needed to heal an NS.

8.3.4.24.2 Attributes

The attributes of the HealNsData information element shall follow the indications provided in table 8.3.4.24.2-1.

Table 8.3.4.24.2-1: Attributes of the HealNsData information element

Attribute	Qualifier	Cardinality	Content	Description
degreeHealing	M	1	Enum	Indicates the degree of healing. Allowed values include: "complete healing of the NS restoring the state of the NS before the failure occurred", "complete healing based on the newest QoS values", "complete healing resetting to the state of the NS at the end of instantiation " and "partial healing".
actionsHealing	M	0..N	String	Used to specify dedicated healing actions in a particular order (e.g. as a script). The actionsHealing can be used to provide a specific script whose content and actions might only be possible to be derived during runtime. See note.
healScript	M	0..1	StringIdentifier (Reference to LifeCycleManagementScript)	Provides a rReference to a script from the NSD that shall be used to execute dedicate healing actions in a particular order. The healScript, since it refers to a script in the NSD, can be used to execute healing actions which can be defined in a static way during NS design time. See note.
NOTE: The actionsHealing and healScript are complementary and can be provided jointly in a single request. In such a case, the actionsHealing are used as parameters to the referred healing script.				

8.3.4.25 HealVnfData information element

8.3.4.25.1 Description

The information element describes the information needed to heal a VNF that is part of an NS. The NFVO shall then invoke the HealVNF operation towards the appropriate VNFM. The specific parameters passed by the NFVO to the VNFM are specified in clause 7.2.10.2 of ETSI GS NFV-IFA 007 [i.5].

8.3.4.25.2 Attributes

The attributes of the HealVnfData information element shall follow the indications provided in table 8.3.4.25.2-1.

Table 8.3.4.25.2-1: Attributes of the HealVnfData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifies the VNF instance, part of the NS, requiring a healing action.
cause	M	0..1	String	Indicates the reason why a healing procedure is required.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the healing process, specific to the VNF being healed, as declared in the VNFD (see clause 7.1.5.6 in ETSI GS NFV-IFA 011 [2]) EXAMPLE: Input parameters to VNF-specific healing procedures.

8.3.4.26 AffinityOrAntiAffinityRule information element

8.3.4.26.1 Description

The AffinityOrAntiAffinityRule describes the additional affinity or anti-affinity rule applicable between the VNF instances to be instantiated in the NS instantiation operation request or between the VNF instances to be instantiated in the NS instantiation operation request and the existing VNF instances.

8.3.4.26.2 Attributes

The attributes of the AffinityOrAntiAffinityRule information element shall follow the indications provided in table 8.3.4.26.2-1.

Table 8.3.4.26.2-1: Attributes of the AffinityOrAntiAffinityRule information element

Attribute	Qualifier	Cardinality	Content	Description
descriptorId	M	1..N	Identifier (Reference to VNFD or VnfProfile)	Reference to a VNFD or vnfProfile defined in the NSD. At least one VnfProfile which is used to instantiate VNF for the NS to be instantiated as the subject of the affinity or anti-affinity rule shall be present. When the VNFD or the VnfProfile which is not used to instantiate VNF, it presents all VNF instances of this type as the subjects of the affinity or anti-affinity rule. The VNF instance which the VnfProfile or the VNFD presents is not necessary as a part of the NS to be instantiated.
vnfInstanceId	M	0..N	Identifier (Reference to VNF instance)	Reference to the existing VNF instance as the subject of the affinity or anti-affinity rule. The existing VNF instance is not necessary as a part of the NS to be instantiated.
affinityOrAntiAffiinty	M	1	Boolean	Specifies whether the rule is an affinity rule (TRUE) or an anti-affinity rule (FALSE).
scope	M	1	Enum	Specifies whether the scope of the rule is a node, a PoP, etc.

8.3.4.27 ChangeNsFlavourData information element

8.3.4.27.1 Description

The ChangeNsFlavourData specifies a new DF to be applied to the NS instance.

8.3.4.27.2 Attributes

The attributes of the ChangeNsFlavourData information element shall follow the indications provided in table 8.3.4.27.2-1.

Table 8.3.4.27.2-1: ChangeNsFlavourData operation information element

Attribute	Qualifier	Cardinality	Content	Description
newFlavourId	M	1	Identifier	Identifier of the new NS DF to be applied to this NS instance.
nsInstantiationLevelId	M	0..1	Identifier	Identifier of the NS instantiation level of the DF to be used. If not present, the default NS instantiation level as declared in the NSD shall be used.

8.3.4.28 ExtManagedVirtualLinkData information element

8.3.4.28.1 Description

This information element provides the information of an externally-managed internal VL to be used as a parameter passed to multiple interfaces.

8.3.4.28.2 Attributes

The ExtManagedVirtualLinkData information element shall follow the indications provided in table 8.3.4.28.2-1.

Table 8.3.4.28.2-1: Attributes of the ExtManagedVirtualLinkData information element

Attribute	Qualifier	Cardinality	Content	Description
extManagedVirtualLinId	M	1	Identifier	Identifier of this externally-managed internal VL instance.
virtualLinkDescId	M	1	Identifier (Reference to VnfVirtualLinkDesc)	Identifier of the VLD in the VNFD for this VL.
vimId	CM	0..1	Identifier	Identifier of the VIM that manages this resource. This attribute shall be supported and present if VNF-related resource management in direct mode is applicable.
resourceProviderId	CM	0..1	Identifier	Identifies the entity responsible for the management of the resource. This attribute shall be supported and present when VNF-related Resource Management in indirect mode is applicable.
resourceId	M	1	Identifier	Identifier of the resource in the scope of the VIM or the resource provider.

8.3.4.29 ChangeExtVnfConnectivityData information element

8.3.4.29.1 Description

The ChangeExtVnfConnectivityData information element specifies the external connectivity to change for the VNF. The types of changes that this operation supports are:

- Disconnect the external CPs that are connected to a particular external VL, and connect them to a different external VL.
- Change the connectivity parameters of the existing external CPs, including changing addresses.

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

8.3.4.29.2 Attributes

The attributes of the ChangeExtVnfConnectivityData information element shall follow the indications provided in table 8.3.4.29.2-1.

Table 8.3.4.29.2-1: Attributes of the ChangeExtVnfConnectivityData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.
extVirtualLink	M	1..N	ExtVirtualLinkData	Information about external VLs to change (e.g. connect the VNF to).
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS as input to the external connectivity change process, specific to the VNF being changed, as declared in the VNFD (see clause 7.1.5.10 in ETSI GS NFV-IFA 011 [2]).

8.4 Information elements and notifications related to NS Performance Management

8.4.1 Introduction

The clauses below define information elements and notifications related to network service performance management.

8.4.2 ObjectSelection information element

8.4.2.1 Description

This information element allows specifying network service instances on which performance information will be provided.

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

In the present interface, the ObjectSelection pattern is used to select NS instances.

The pattern proposes 2 exclusive options:

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

In the present interface, the object type will be NS (represented by an NSD), and the filter based on some NS properties.

8.4.2.2 Attributes

The attributes of the ObjectSelection information element shall follow the indications provided in table 8.4.2.2-1.

Table 8.4.2.2-1: Attributes of the ObjectSelection information element

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

8.4.3 PmJob information element

8.4.3.1 Description

This information element provides the details of the PM Job.

The object instances for this information element will be NS instances.

8.4.3.2 Attributes

The attributes of the PmJob information element shall follow the indications provided in table 8.4.3.2-1.

Table 8.4.3.2-1: Attributes of the PmJob information element

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

8.4.4 Threshold information element

8.4.4.1 Description

This information element provides the details of a threshold.

The object instances for this information element will be NS instances.

8.4.4.2 Attributes

The attributes of the Threshold information element shall follow the indications provided in table 8.4.4.2-1.

Table 8.4.4.2-1: Attributes of the Threshold information element

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

8.4.5 PerformanceReport information element

8.4.5.1 Description

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

The object instances for this information element will be NS instances.

8.4.5.2 Attributes

The attributes of the PerformanceReport information element shall follow the indications provided in table 8.4.5.2-1.

Table 8.4.5.2-1: Attributes of the PerformanceReport information element

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

8.4.6 PerformanceReportEntry information element

8.4.6.1 Description

This information element defines a single performance report entry.

The object instances for this information element will be NS instances.

8.4.6.2 Attributes

The attributes of the PerformanceReportEntry information element shall follow the indications provided in table 8.4.6.2-1.

Table 8.4.6.2-1: Attributes of the PerformanceReportEntry information element

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

8.4.7 PerformanceValueEntry information element

8.4.7.1 Description

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

8.4.7.2 Attributes

The attributes of the PerformanceValueEntry information element shall follow the indications provided in table 8.4.7.2-1.

Table 8.4.7.2-1: Attributes of the PerformanceValueEntry information element

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

8.4.8 PerformanceInformationAvailableNotification

8.4.8.1 Description

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

The object instances for this information element will be NS instances.

8.4.8.2 Trigger Conditions

The notification is produced when:

- New performance information is available.

8.4.8.3 Attributes

The attributes of the PerformanceInformationAvailableNotification shall follow the indications provided in table 8.4.8.3-1.

Table 8.4.8.3-1: Attributes of the PerformanceInformationAvailableNotification

Attribute	Qualifier	Cardinality	Content	Description
objectInstanceId	M	1..N	Identifier	Object instances for which performance information is available. The object instances for this information element will be NS instances.

8.4.9 ThresholdCrossedNotification

8.4.9.1 Description

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

The object instances for this information element will be NS instances.

8.4.9.2 Trigger Conditions

The notification is produced when:

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

8.4.9.3 Attributes

The attributes of the ThresholdCrossedNotification shall follow the indications provided in table 8.4.9.3-1.

Table 8.4.9.3-1: Attributes of the ThresholdCrossedNotification

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

8.5 Information elements and notifications NS Fault management

8.5.1 Introduction

The clauses below define information elements and notifications related to network service fault management.

8.5.2 AlarmNotification

8.5.2.1 Description

This notification informs the receiver of alarms related to the network services managed by the NFVO. The notification is mandatory.

8.5.2.2 Trigger conditions

The notification is produced when:

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

8.5.2.3 Attributes

The attributes of the AlarmNotification shall follow the indications provided in table 8.5.2.3-1.

Table 8.5.2.3-1: Attributes of the AlarmNotification

Attribute	Qualifier	Cardinality	Content	Description
alarm	M	1	Alarm	Information about an alarm including AlarmId, affected network service ID, and FaultDetails.

8.5.3 AlarmClearedNotification

8.5.3.1 Description

This notification informs the receiver of the clearing of an alarm related to the network services managed by the NFVO. The alarm's perceived severity shall be set to "cleared" since the corresponding fault has been solved. The notification is mandatory.

8.5.3.2 Trigger conditions

The notification is produced when:

- An alarm has been cleared.

8.5.3.3 Attributes

The attributes of the AlarmClearedNotification shall follow the indications provided in table 8.5.3.3-1.

Table 8.5.3.3-1: Attributes of the AlarmClearedNotification

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

8.5.4 Alarm information element

8.5.4.1 Description

The Alarm information element encapsulates information about an alarm.

8.5.4.2 Attributes

The attributes of the Alarm information element shall follow the indications provided in table 8.5.4.2-1.

Table 8.5.4.2-1: Attributes of the Alarm information element

Attribute	Qualifier	Cardinality	Content	Description
alarmId	M	1	Identifier	Identifier of this Alarm information element.
managedObjectId	M	1	Identifier	Identifier of the affected managed object. The Managed Objects for the current specification will be network services.
alarmRaisedTime	M	1	DateTime	Timestamp indicating when the alarm was raised by the network service.
alarmChangedTime	M	0..1	DateTime	Timestamp indicating when the alarm was last changed.
alarmClearedTime	M	0..1	DateTime	Timestamp indicating when the alarm was cleared.
state	M	1	Enum	State of the alarm, e.g. "fired", "updated", "cleared".
perceivedSeverity	M	1	Enum	Perceived severity of the managed object failure, legal values: <ul style="list-style-type: none"> • Critical. • Major. • Minor. • Warning. • Indeterminate. • Cleared.
eventTime	M	1	DateTime	Timestamp indicating when the fault was observed.
faultType	M	1	String	Type of the fault. The allowed values for this attribute depend on the type of the related managed object.
probableCause	M	1	String	Information about the probable cause of the fault.
isRootCause	M	1	Boolean	Parameter indicating if this fault is the root for other correlated alarms. If TRUE, then the alarms listed in the parameter correlatedAlarmId are caused by this fault.
correlatedAlarmId	M	0..N	Identifier (Reference to Alarm)	List of identifiers of other alarms correlated to this fault.
faultDetails	M	0..N	Not specified.	Provides additional information about the fault.

8.6 Information elements and notifications related to VNF Package

8.6.1 Introduction

The clauses below define information elements and notifications related to VNF Package management.

8.6.2 OnboardedVnfPkgInfo information element

8.6.2.1 Description

This information element provides the details of an on-boarded VNF Package, which the NFVO creates and stores as part of the on-boarding and ongoing operational management process.

8.6.2.2 Attributes

The attributes of the OnboardedVnfPkgInfo information element shall follow the indications provided in table 8.6.2.2-1.

Table 8.6.2.2-1: Attributes of the OnboardedVnfPkgInfo information element

Attribute	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfoId	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF Package. This was allocated by the NFVO.
vnfdId	M	1	Identifier	Identifier that identifies the VNF Package. See note.
vnfProvider	M	1	String	See note.
vnfProductName	M	1	String	See note.
vnfSoftwareVersion	M	1	Version	See note.
vnfdVersion	M	1	Version	See note.
checksum	M	1	Not specified.	Checksum of the on-boarded VNF Package.
vnfd	M	1	Vnfd	VNFD contained in the on-boarded VNF Package.
softwareImage	M	1..N	VnfPackageSoftwareImageInformation	Information about VNF Package artifacts that are software images.
additionalArtifact	M	0..N	VnfPackageArtifactInformation	Information about VNF Package artifacts contained in the VNF Package that are not software images.
operationalState	M	1	OperationalState: Enum { Enabled, Disabled}	Operational state of the on-boarded instance of the VNF Package.
usageState	M	1	UsageState: Enum {InUse, NotInUse}	Usage state of the on-boarded instance of the VNF Package.
deletionPending	M	1	Boolean	Indicates if deletion of this instance of the VNF Package has been requested but the VNF Package is still being used by instantiated VNFs. This instance of the VNF Package will be deleted once all VNFs instantiated from this package are terminated.

Attribute	Qualifier	Cardinality	Content	Description
userDefinedData	O	0..N	KeyValuePair	User defined data for the VNF Package.
NOTE: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package.				

8.6.3 Vnfd information element

8.6.3.1 Description

This information element provides the details of the VNFD.

8.6.3.2 Attributes

The structure of the Vnfd information element shall comply with the provisions for the Vnfd information element as defined in ETSI GS NFV-IFA 011 [2], clause 7.1.2.

8.6.4 VnfPackageSoftwareImageInformation information element

8.6.4.1 Description

This information element represents an artifact contained in a VNF Package which represents a Software Image.

8.6.4.2 Attributes

The VnfPackageSoftwareImageInformation information element shall follow the indications provided in table 8.6.4.2-1.

Table 8.6.4.2-1: Attributes of the VnfPackageSoftwareImageInformation information element

Attribute	Qualifier	Cardinality	Content	Description
softwareImageInformation	M	1	SoftwareImageInformation	Information on the software image(s). Refer to clause 8.6.5.
accessInformation	M	1	Not specified.	Information (such as a URL, a path in the VNF Package, or an identifier) that allows to access a copy of this software image asset.

8.6.5 SoftwareImageInformation information element

8.6.5.1 Description

This information element represents Software Image Information.

NOTE: The definition below is aligned with the definition of the SoftwareImageInformation information element in ETSI GS NFV-IFA 005 [i.4].

8.6.5.2 Attributes

The SoftwareImageInformation information element shall follow the indications provided in table 8.6.5.2-1.

Table 8.6.5.2-1: Attributes of the SoftwareImageInformation information element

Attribute	Qualifier	Cardinality	Content	Description
id	M	1	Identifier	The identifier of this software image.
name	M	1	Not specified.	The name of this software image.
provider	M	1	Not specified.	The provider of this software image.
version	M	1	Not specified.	The version of this software image.
checksum	M	1	Not specified.	The checksum of the software image file.
containerFormat	M	1	Not specified.	The container format indicates whether the software image is in a file format that also contains metadata about the actual software.
diskFormat	M	1	Not specified.	The disk format of a software image is the format of the underlying disk image.
createdAt	M	1	Not specified.	The created time of this software image.
minDisk	M	1	Not specified.	The minimal Disk for this software image.
minRam	M	1	Not specified.	The minimal RAM for this software image.
size	M	1	Not specified.	The size of this software image.
userMetadata	M	0..N	KeyValuePair	User-defined metadata.

8.6.6 VnfPackageArtifactInformation information element

8.6.6.1 Description

This information element provides identification information for an artifact (other than a Software Image) which is contained in the VNF Package.

8.6.6.2 Attributes

The VnfPackageArtifactInformation information element shall follow the indications provided in table 8.6.6.2-1.

Table 8.6.6.2-1: Attributes of the VnfPackageArtifactInformation information element

Attribute	Qualifier	Cardinality	Content	Description
selector	M	1	Not specified.	Information (such as a path) that identifies this artifact in the VNF Package. Definition of the format is left to protocol design.
metadata	M	1	Not specified.	The metadata of the artifact that are available in the VNF Package, such as Content type, size, creation date, etc.

8.6.7 Void

8.6.8 VnfPackageOnBoardingNotification

8.6.8.1 Description

This notification indicates the on-boarding of a VNF Package. Support of this notification is mandatory.

8.6.8.2 Trigger Conditions

- New VNF Package on-boarded.

8.6.8.3 Attributes

The attributes of the VnfPackageOnBoardingNotification shall follow the indications provided in table 8.6.8.3-1.

Table 8.6.8.3-1: Attributes of the VnfPackageOnBoardingNotification

Attribute	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfolId	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF Package. This identifier was allocated by the NFVO.
vnfdId	M	1	Identifier	Identifier that identifies the VNF Package. See note.
NOTE: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package.				

8.6.9 VnfPackageChangeNotification

8.6.9.1 Description

This notification indicates a change of status in a VNF Package. Only changes in operational state and deletion pending attribute will be reported, change in usage state is not reported.

Support of this notification is mandatory.

8.6.9.2 Trigger Conditions

- Change of the status (operational state and deletion pending) of an on-boarded VNF Package.
- Deletion of an on-boarded VNF Package.

8.6.9.3 Attributes

The attributes of the VnfPackageChangeNotification shall follow the indications provided in table 8.6.9.3-1.

Table 8.6.9.3-1: Attributes of the VnfPackageChangeNotification

Attribute	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfold	M	1	Identifier	Identifier of information held by the NFVO about the specific on-boarded VNF Package. This identifier was allocated by the NFVO.
vnfdId	M	1	Identifier	Identifier that identifies the VNF Package. See note.
changeType	M	1	Enum	It categorizes the type of change. Possible values can be change of operational state of an on-boarded VNF Package, VNF Package in deletion pending, and deletion of a VNF Package.
operationalState	M	0..1	OperationalState: Enum { Enabled, Disabled}	New operational state of the VNF Package. Only present when changeType is change of operational state.
deletionPending	M	0..1	Boolean	Indicates if the deletion instance of the VNF Package has been requested but the VNF Package still being used by instantiated VNFs. Only present when changeType is VNF Package in deletion pending.
NOTE: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package.				

Annex A (informative): Principles related to VNF lifecycle management and NS lifecycle management

The following bullets list the main principles related to VNF lifecycle management and NS lifecycle management:

- 1) **Principle #1** The VNFM offers a layer of decoupling for VNFs:
 - VNF lifecycle management is handled by the VNFM corresponding to a given VNF.
 - The VNF can be modified without a call through the VNF lifecycle management interface, for instance in reaction to some alarms/metrics resulting in a scaling. So the FBs using the service of the VNF lifecycle management interface, like NFVO, should expect that some changes are possible for a given VNF. These changes are communicated through notifications.
 - This does not prevent NFVO from being able to send VNF lifecycle management operations to the VNFM.
- 2) **Principle #2** The NFVO offers a layer of decoupling for NSs:
 - NS lifecycle management is handled by the NFVO corresponding to a given NS.
 - The NS can be modified without a request from the OSS/BSS through the NS lifecycle management interface, either because of a VNF change or because the NFVO has reacted to some event and decided to change something in the NS or in a VNF that is part of an NS. So the OSS/BSS should expect that some changes in the NS (and the VNFs that are part of it) are possible. These changes are communicated through notifications.
 - This does not prevent OSS/BSS from being able to send directly NS lifecycle management operations to the NFVO.
- 3) **Principle #3**
 - With respect to the Os-Ma-Nfvo reference point, any interaction concerning a VNF is associated with at least one NS instance.

Annex B (informative): Use cases for VNF reuse and referencing in NSs

B.1 Re-use of VNFs from a terminated NS

In this use case, the consumer (OSS/BSS FB) requests that the provider (NFVO) terminate a given NS instance, and the NFVO retains some of the VNF instances for use in future NS instances (in NS instantiate and/or NS update).

NOTE: The OSS/BSS FB can instruct the NFVO to retain VNF instances by adding the VNF instances in another NS (e.g. a VNF Pool NS).

Some additional points to consider with regard to the reuse of VNF instances:

- At the time the NS instance is terminated, the consumer may not know what specific NS instances are to use the retained VNF instances but only that the VNF instances should be retained. The VNF instances to be retained need to be part of another NS instance (e.g. a VNF Pool NS) before terminating the NS instance to which those VNFs belong.
- The retained VNF instances are of various types, could be a relative large number (consider an NS type such a virtual CDN) and may be reused in multiple NS instances (which may not even be of the same type as the original NS).
- A retained VNF instance may need to be updated before being used in a new NS instance.
- Some reasons for VNF instance retention:
 - The particular type of VNF may take some "cost" (e.g. time) to instantiate and it is easier to retain the VNF instances for future use.
 - There is a known demand for a given type of VNF and it is easier to just retain the VNF instance rather than to terminate and wait for when it needs to be re-used (which might be immediately).
 - As part of some sort of NS instance migration, it may be necessary (or easier) to retain some of the VNF instances from the NS instance that is to be migrated.
 - The VNF instance is part of a leased network segment (allocated to a specific consumer) and whether to retain the VNF instance or not is basically a decision of the consumer.

B.2 Creation of VNF instances in anticipation of future NS demand

Based on (for example, customer demand forecasts) VNF instances are created in advance for use in NS instances at some later point in time. This might be done because instances of the given type of VNF may take some time to instantiate and the OSS/BSS FB wants to ensure quick instantiation of the NS instances that use these VNF instances.

In a variation of this use case, the OSS/BSS FB may create VNF instances in advance for use in pre-ordered NS instances (for some future date by a customer). In this case, the OSS/BSS FB could first request instantiation of the required resources (e.g. VNF instances) in a context of a holding NS (e.g. a VNF Pool NS), and then request the NS instance later when it is needed, with the knowledge that the required VNF instances (for example) are there and ready to be used immediately.

B.3 Bottom-up NS instantiation

From ETSI GS NFV-MAN 001 [i.2], clause 6.8, with regard to the "Reference to an existing VNF instance" information element:

- "List of references to existing VNF instances. This is required, for example, when the NS is being instantiated **bottom-up** wherein the members are already existing and need to be chained together."

It may be that VNF instances are created in advance for a given NS instance, with the knowledge that some of the VNF instances will take longer to create than others and that an immediate NS instantiate request will fail unless all the required VNF instances are instantiated beforehand.

B.4 Shared VNF instances

Some VNF instances can be shared by several NS instances. In some cases, the OSS/BSS FB will indicate to the NFVO that a VNF instance (already being used by at least one NS instance) is be used in another NS instance.

A shared VNF instance may need to be updated before it is reused in another NS instance.

Annex C (informative): Message flows for supporting use cases with fine grained NS lifecycle management

C.1 Introduction

The present section describes how the use cases presented in Annex B can be supported by fine-grain control over VNF instances in NS lifecycle management operations.

The approach uses the basic NS lifecycle management operations and within the NS lifecycle management operations additional attributes are created to support the fine grained VNF lifecycle management tasks.

Table C.1-1 provides the mapping of all identified use cases and proposed solutions.

Table C.1-1

Existing Use Cases in annex B	Solutions in annex C
Clause B.1 Re-use of VNF instances from a terminated NS	"Terminating an NS with retained VNF instance(s)"
Clause B.2 Creation of VNF instances in anticipation of future NS demand	"VNF pool creation"
Clause B.3 Bottom-up NS instantiation	"VNF pool creation" (steps 6 and 7)
Clause B.4 Shared VNF instances	"Terminating an NS with retained VNF instance(s)" (steps 1a, 1b and 2) and "New NS with existing VNF instance(s)"

C.2 New NS with VNF pools

A VNF pool NS can be utilized for the use case of creation of VNF instances in anticipation of future NS demand. (The associated use case is defined in clause B.2.)

The VNF Pool NS is just a normal NS which contains VNF instances and may not have any connectivity between the VNF instances. There are many ways how the VNF Pool NS can be "filled up" with VNF instances. VNFs can be instantiated when an NS is instantiated or OSS/BSS can add VNF instances to the NS instance by the Update NS operation or OSS/BSS can associates existing VNF instances to the NS with the Update NS operation.

Figure C.2-1 shows the flow how a VNF pool NS can be created.

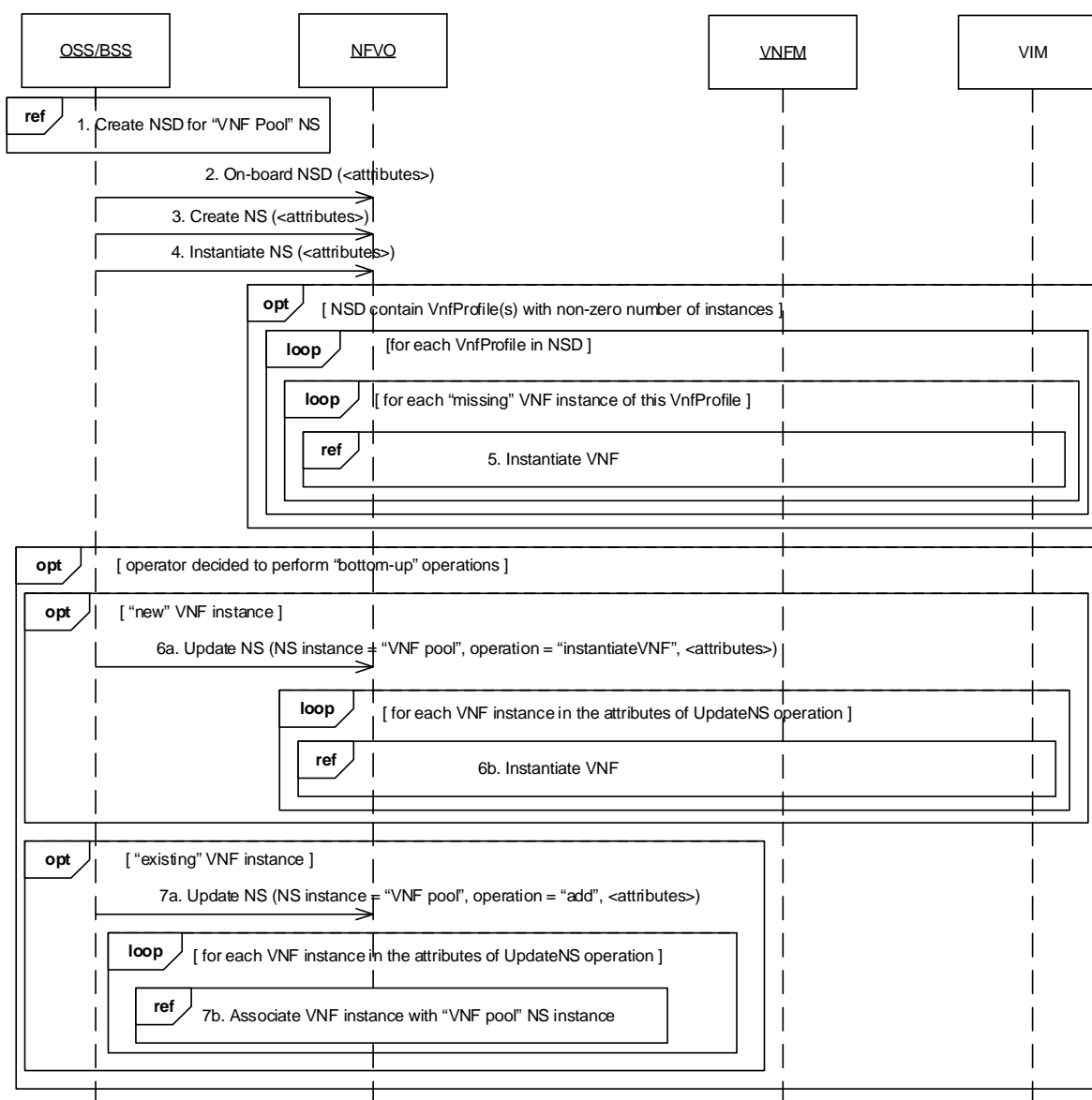


Figure C.2-1: VNF Pool NS creation

- 1) As a pre-requisite for instantiation of a NS, the corresponding NSD has to be created.
- 2) The NSD is on-boarded to the NFVO. It is assumed that all VNF Packages who's VNFDs are referred in the NSD have already been on-boarded to the NFVO.
- 3) The OSS/BSS issues the Create NS operation to the NFVO.
- 4) The OSS/BSS issues the Instantiate NS operation to the NFVO.
- 5) If the NSD contains VNF Profiles with a non-zero number of VNF instances to be instantiated in the NS, then the NFVO performs the instantiation procedure for each VNF that requires instantiation.
- 6) When the NS is instantiated and an NS instance ID is available, the OSS/BSS may request additional VNF instances to be added to the "VNF pool" NS instance either by instantiating them (steps 6a and 6b) or by associating the existing VNF instances with the "VNF pool" NS instance (steps 7a and 7b). This allows the operator to perform a "bottom-up" NS instantiation - complete (if no VNF instances were instantiated in step 5, according to the numbers in the NSD) or partial (in addition to those VNF instances that were instantiated in step 5).
 - 6a) In this operation the OSS/BSS tells what new VNF instances should be added to the "VNF pool" NS instance.
 - 6b) The NFVO performs the VNF instantiation(s) according to the request received from the OSS/BSS.

- 7a) In this operation the OSS/BSS tells what existing VNF instances(e.g. those that are associated with other NS instances) should be added (or possibly removed) to (from) the "VNF pool" NS instance.
- 7b) The NFVO associates the VNF instance to the "VNF Pool" NS instance according to the request received from the OSS/BSS.

C.3 New NS utilizing VNF instances from a VNF pool NS

In this clause the flow is described how an NS can be built up from existing VNF instances.

The sequence diagram of the case is shown in figure C.3-1.

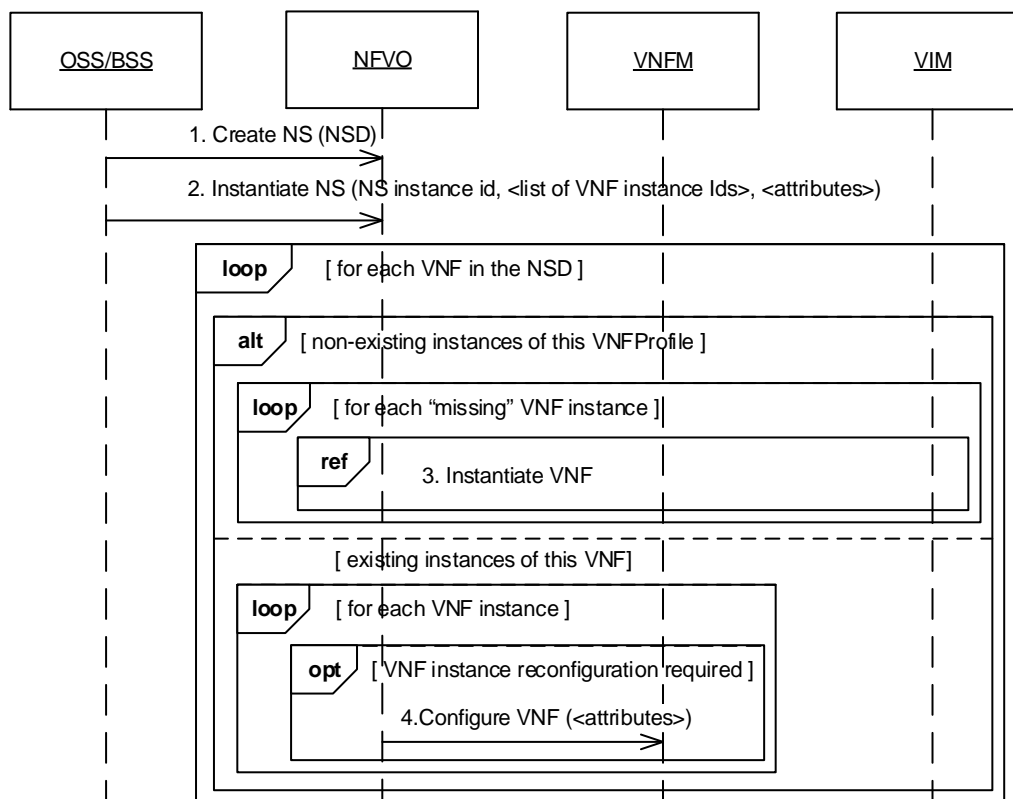


Figure C.3-1: Creating NS with existing VNF instance(s)

- 0) It is assumed, that there is already an existing "VNF pool" NS instance with some VNF instances in it (see "VNF Pool Creation" use case for details). It is also assumed, that the NSD of a "real" NS has been created and on-boarded to the NFVO.
- 1) The OSS/BSS requests the creation of an NS instance for the "real" network service.
- 2) The OSS/BSS requests the instantiation of the NS. As part of the Instantiate NS operation the OSS/BSS can indicate the VNF instances (by listing their ids) to be used in the newly instantiated NS. These VNFs become shared between the "VNF Pool" NS and the "real" NS that is being instantiated.
- 3) The NFVO performs VNF instantiation procedure for each VNFs that is "missing" (does not have an already existing instance) according to the NSD of the NS.
- 4) The NFVO may need to request VNFM to reconfigure the "existing" VNF instances (e.g. those from the "VNF pool") according to the attributes of the NS lifecycle management operation.

C.4 Terminating NS instance with retained VNF instances

In this scenario a NS is terminated, while certain VNF instances needs to be retained. The VNFs to be retained are added to the VNF Pool NS before the "normal" NS is terminated. The flow is described in figure C.4-1.

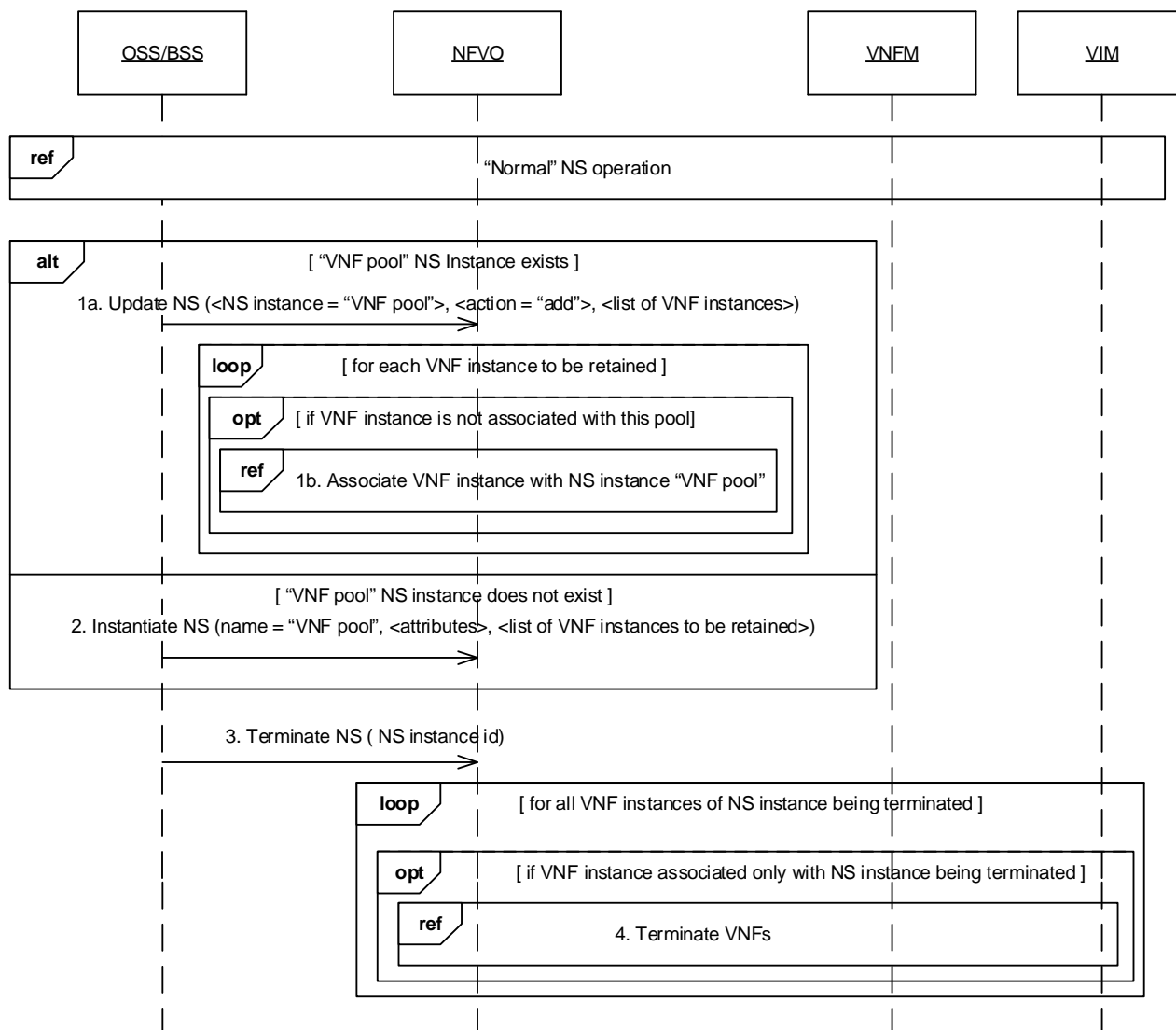


Figure C.4-1: Terminating a NS with retained VNF instance(s)

- 0) It is assumed, that there is an existing instance of an NS.
- 1a) If operator decides to use an existing "VNF pool" NS instance to retain the VNF instances from the NS being terminated, then the OSS/BSS issues an Update NS operation indicating the NS instance Id to be the "VNF Pool" NS and the list of VNF Instance Ids (to be retained) that should be added to the "VNF Pool" NS. Based on operator's decision, there could be multiple "VNF pool" NS instances (e.g. one per VNF type or one per operator's intention) - in such cases, there will be a separate Update NS operation per "VNF pool" NS instance.
- 1b) The NFVO associates each VNF instance being retained to the "VNF Pool" NS instance, according to the Update NS operation.
- 2) If operator decides to use a new "VNF Pool" NS instance, then OSS/BSS needs to create one. In the Instantiate NS operation, OSS/BSS can provide the list of VNF instance Ids to be retained.

- 3) OSS/BSS issues the Terminate NS operation indicating the NS instance Id to be terminated. The VNF instances being retained are already associated with more than one NS (the NS instance being terminated and "VNF pool" NS instance).
- 4) The NFVO terminates all those VNF instances associated only with the NS instance that is being terminated.

Annex D (informative): State models

D.1 VNF Package state model

D.1.1 Introduction

This annex proposes a state model for the use of the VNF Package by the NFVO.

All the steps before the on-boarding of the VNF Package are not part of this state model.

D.1.2 State model

A given version of a VNF Package once on-boarded by the NFVO has 2 elementary states: operational state (Enabled/Disabled) and usage state (In use/Not in use) and a Deletion Pending state.

The state model, shown in figure D.1.2-1, applies to a given version of a VNF Package.

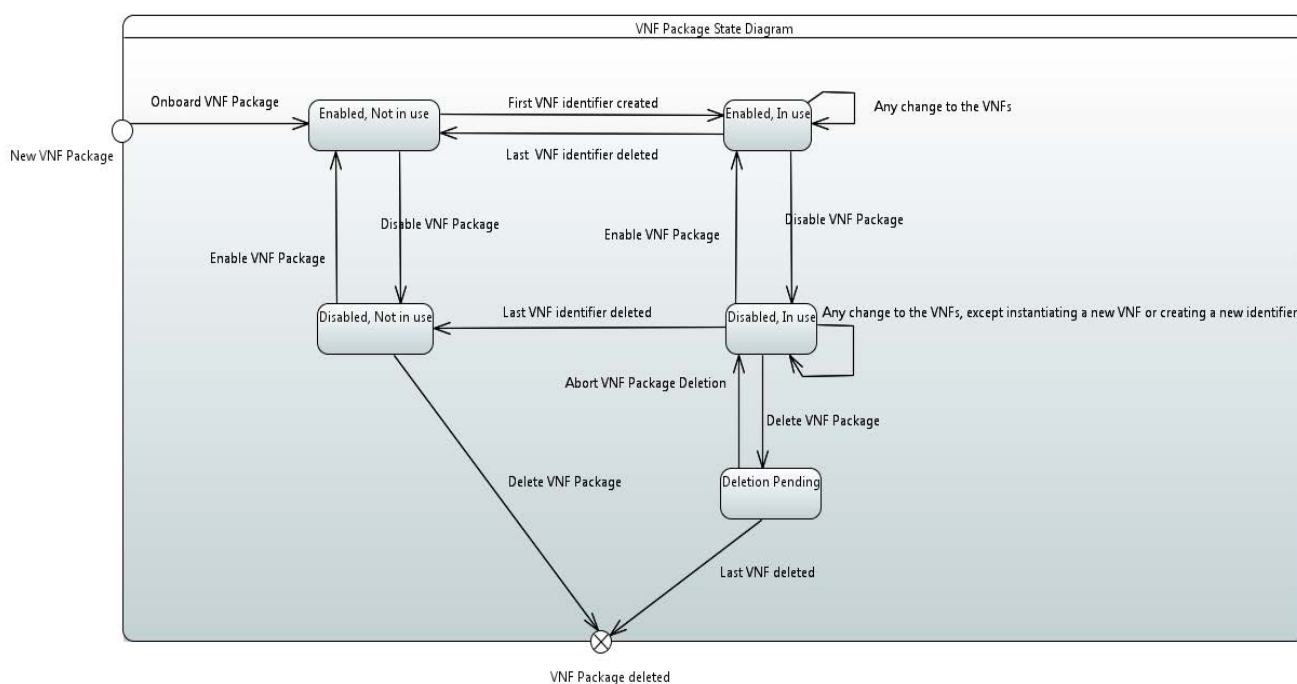


Figure D.1.2-1: VNF Package state model

The query VNF Package operation is considered as available in all the states above.

D.2 NSD state model

D.2.1 Introduction

This annex proposes a state model for the use of the NSD by the NFVO.

All the steps before the on-boarding of the NSD are not part of this state model.

D.2.2 State model

A given version of an NSD once on-boarded by the NFVO has 2 elementary states: operational state (Enabled/Disabled) and usage state (In use/Not in use), and a Deletion Pending state.

The state model, shown in figure D.2.2-1, applies to a given version of an NSD.

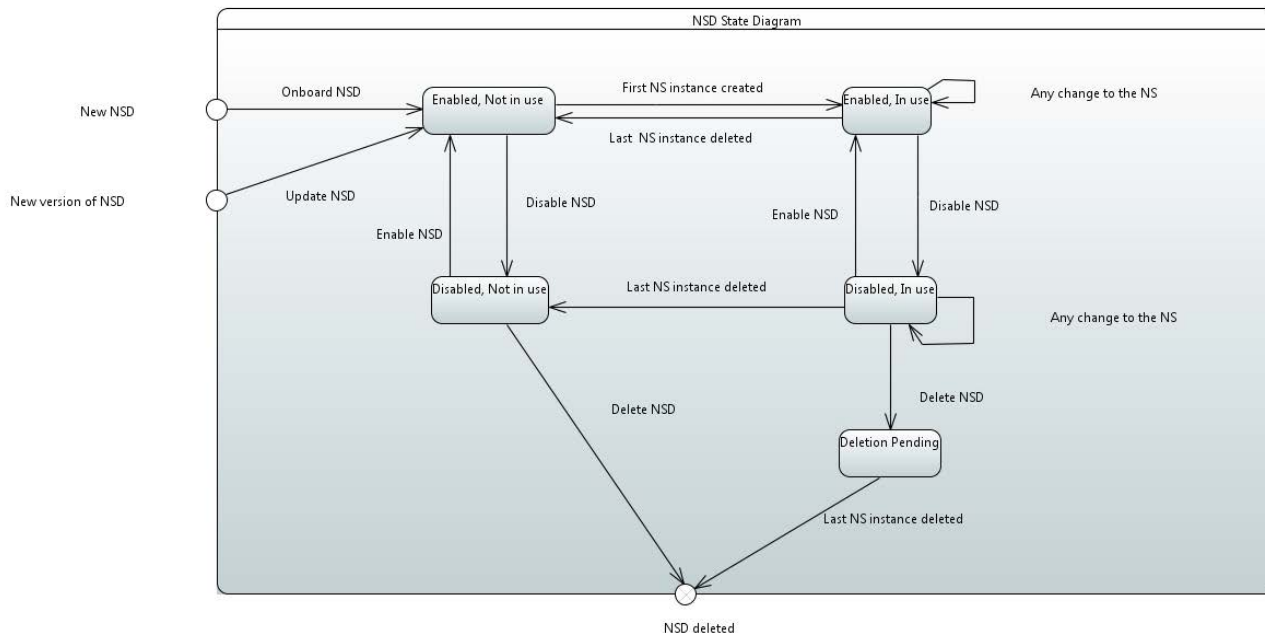


Figure D.2.2-1: NSD state model

The query NSD operation is considered as available in all the states above.

The update NSD will not change the state of the current version of the NSD and will create a new state diagram for the new version, starting in Enabled, Not in use state.

D.3 NS state model

D.3.1 Introduction

This annex proposes a state model for the NS instance in the NFVO.

All the steps before the initial Create NS are not part of this state model.

D.3.2 State model

A given NS instance has 2 elementary state values in the NFVO: INSTANTIATED, NOT_INSTANTIATED, INSTANTIATED.

The state model, shown in figure D.3.2-1, applies to a given NS instance.

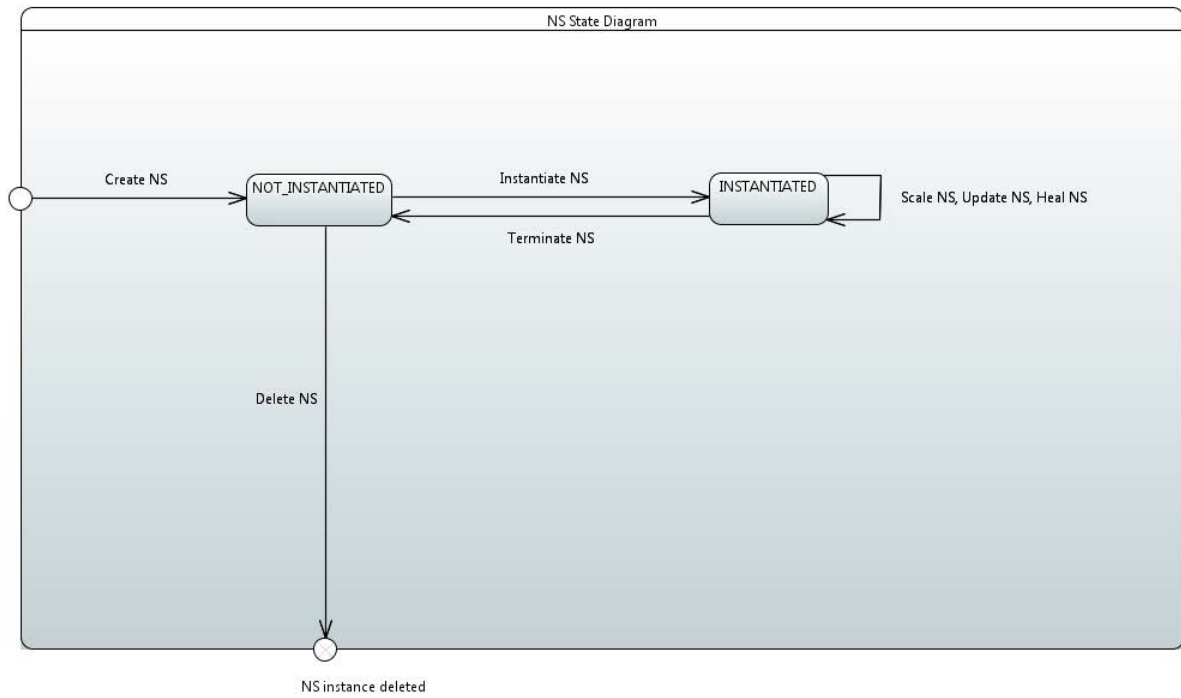


Figure D.3.2-1: Network Service instance state model

The query NS operation is considered as available in all the states above.

Annex E (informative): NS scaling

E.1 Forms of NS scaling

The aim of NS scaling is to increase or decrease the capacity of a network Service instance. This can be achieved in various ways:

- Scaling a VNF instance: if a constituent VNF instance in the NS instance is not yet scaled to its limits. However it is also possible that either the VNF instance is already scaled to its limits or the VNF instance does not support scaling, then VNF scale cannot be used as part of NS scale operation.
- Changing the DF of a VNF instance. If a VNF is already scaled to its limits, then the next step of increasing the capacity of a VNF is to use the DF change (if there is a higher/lower capacity DF of the VNF). If the DF change of the VNF requires topology changes, NFVO can create/modify the required VLs. Also the VNF DF changes may require application level configuration task, which is done by the traditional management system therefore interaction with OSS/BSS or EM may be required.
- Adding/removing VNF instance(s) to/from the NS: In this scenario the capacity of the NS is changed by adding/removing VNF instance(s) to/from the NS instance. In this scenario the NFVO need to take care of creating the necessary links between the VNF instances according to the connectivity requirements defined in the NSD. This type of NS scaling may also require application level configuration task, which needs to be performed by the traditional management system (by OSS/BSS or EM).
- Scaling to a new NS DF: In this scenario the NS DF is changed to a new one, which contains higher/lower capacities. The NFVO may require instantiation/termination of VNF instances according to the NSD of the new NS DF.
- Scaling a nested NS: The capacity of an NS can also be changed by changing the capacity of a nested NS if the network service contains a nested network service.
- Scaling of a VL: It may also be necessary as part of the capacity change need of an NS to change the capacity of a VL in an NS. This may be achieved either by changing the properties of a VL or by adding/removing VL in an NS. The latter may require application level configuration as well, therefore interaction with OSS/BSS may be required.

E.2 NS scaling triggers

E.2.1 NS auto-scale

In this case the NS scale decision is made at NFVO based on the information provided in the NSD. The main attributes to be used for the NS auto-scale functionality is the monitoring parameter and the indicators that are re-exposed in the Or-Vnfm reference point specified in the NSD and the associated auto-scale rule.

E.2.2 NS scale triggered by OSS/BSS

In this case the NS scale operation is requested by OSS/BSS via the Os-Ma-nfvo reference point. OSS/BSS can specify what to scale by providing the scale rules to NFVO (similar to the auto-scale rules defined in the NSD). The OSS/BSS may also provide explicit guidance to the NFVO what to scale and in what way, i.e. the OSS/BSS may tell to NFVO to scale a specific VNF instance to a specific scale level.

E.3 Relation to NS DF

The NS scale operates either within the boundaries of a network service DF as specified in the NSD or by changing the NS DF. The NS DF provides the minimum and maximum number of instances of each VNFs the NS is built upon. Each VNF in the NSD references to a specific VNF DF as specified in the VNFD.

As a consequence of the above an NS scale can operate via a VNF scale within the boundaries of a VNF DF as specified in the VNFD or by adding/removing VNF instances within the boundaries of a NS DF as specified in the NSD.

The capacity of an NS may be changed by changing or moving from one NS DF to another DF.

If an NS contains a nested NS, the change of the capacity can be achieved by scaling the nested NS.

E.4 Input and tools for NS auto-scaling

E.4.1 Monitoring parameter

Monitoring parameters are defined in NSD and may be used to trigger the necessary NS scaling actions at NFVO. Monitoring parameters can specify the values/threshold of a PM counter that is available at NFVO (e.g. derived from virtualised resource performance metrics) and the associated auto-scaling rules/policies.

E.4.2 VNF indicator

VNF indicators are declared in the VNFD and are provided either by the VNF or by the EM managing the VNF. These VNF indicators are forwarded to the NFVO by the VNFM managing the VNFs. These VNF indicators may also be used by the NFVO for its NS auto-scale functionality.

E.4.3 Auto-scale policies/rules

This should define the required scaling actions based on the monitoring parameters and/or VNF indicators. It should define in priority order e.g. what VNFs to scale or whether a VNF instance to be added or removed to the NS instance. In case of VNF scale it should specify also the Scale aspect of the VNF scale operation that can be used from NFVO towards the VNFM.

Annex F (informative): Authors & contributors

The following people have contributed to the present document:

Rapporteur:

Marc Flauw, Hewlett Packard Enterprise

Other contributors:

Anatoly Andrianov, Nokia Networks

Feng Aijua, Huawei

Ernie Bayha, Ericsson LM

Andy Bennett, Cisco

Gyula Bodog, Nokia Networks

Michael Brenner, Alcatel-Lucent

Haibin Chu, Ericsson LM

Yu Fang, Huawei

Stephen Fratini, Ericsson LM

Dmytro Gassanov, NetCracker

Deepanshu Gautam, Huawei

Xia Haitao, Huawei

Gang He, China Unicom

Drew Jordan, Sigma Systems

Ashiq Khan, DOCOMO Communications Lab

Michael Klotz, Deutsche Telekom AG

Anton Korchak, NetCracker

Ryosuke Kurebayashi, DOCOMO Communications Lab

Keisuke Kuroki, KDDI Corporation

Gerald Kunzmann, DOCOMO Communications Lab

Tommy Lindgren, Ericsson

Jie Miao, China Unicom

Kazuaki Obana, DOCOMO Communications Lab

Zhao Peng, China Mobile

Janusz Pieczerak, ORANGE

Uwe Rauschenbach, Nokia Networks

Nicola Santinelli, TELECOM ITALIA

Bertrand Souville, DOCOMO Communications Lab

Junjie Tong, China Unicom

Joan Triay, DOCOMO Communications Lab

Xu Ruiyue, Huawei

Amanda Xiang, Huawei

Jong-Hwa Yi, ETRI

Zarrar Yousaf, NEC Europe

Zhou Yan, Huawei

Annex G (informative): Change History

Date	Version	Information about changes
10 April 2017	V2.1.2	NFVIFA(17)000056r1, IFA007ed221 IFA013ed221 VNF Package Management modifications NFVIFA(17)000061r1, IFA013 Fix for disabled VNF Packages NFVIFA(17)000257r12, IFA007ed221 IFA008ed221 IFA013ed221 Fix to dynamic addresses in VnfExtCpData IE
23 May 2017	V2.1.3	NFVIFA(17)000130, IFA013ed211 Fix the title mismatching for Table 8.3.4.25.2-1 NFVIFA(17)000413r1, IFA013ed221: Merge NS LCM and the NS LCCN Interfaces
23 June 2017	V2.1.4	NFVIFA(17)000507r2, IFA013ed221 (mirror of 471r1) additionalParameters missing NFVIFA(17)000382r1, IFA013ed221_Update ScaleVnfData and ScaleByStepData Information Elements NFVIFA(17)000481, IFA013ed221 Add nsVirtualLinkInstanceId to the NsVirtualLinkInfo Information Element NFVIFA(17)000509, IFA013ed221: Add NS LCM Terminate Subscription and Query Subscription Info Operations NFVIFA(17)000510, IFA013ed221: Change attribute vnfInfo to vnfInfo in NsInfo IE NFVIFA(17)000527r2, IFA013ed221: Split Modify VNF Configuration in Update NS operation NFVIFA(17)000540, IFA013ed221: Add error handling operations NFVIFA(17)000565r1, IFA013ed221 (mirror of 193) ThresholdCrossedNotification trigger condition fix NFVIFA(17)000566, IFA013ed221 (mirror of 236r1) clarify description of MonitoringParameter NFVIFA(17)000569r1, IFA013ed221 bulk mirror of 116r2 438 473 469 NFVIFA(17)000573, IFA013ed231 (mirror of 427r2 and 531) VL and CP consistency NFVIFA(17)000574r2, IFA013ed231 8.3.4.24 Consistency of actionsHealing NFVIFA(17)000581, IFA013ed221: Change name of the "notificationType" attribute to "status" consistent with IFA007 NFVIFA(17)000595r3, IFA013 ed221 CR Align the usage of VNF instantiation state

History

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