



Network Functions Virtualisation (NFV) Release 2; Protocols and Data Models; RESTful protocols specification for the Os-Ma-nfvo Reference Point

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-SOL005ed271

Keywords

API, NFV, protocol

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 prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

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 2020.

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 a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners.

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

Contents

Intellectual Property Rights	18
Foreword.....	18
Modal verbs terminology.....	18
1 Scope	19
2 References	19
2.1 Normative references	19
2.2 Informative references.....	20
3 Definition of terms, symbols and abbreviations.....	20
3.1 Terms.....	20
3.2 Symbols.....	21
3.3 Abbreviations	21
4 General Aspects.....	22
4.1 Overview	22
4.2 Void.....	22
4.3 Void.....	22
4.4 Common data types	23
4.4.1 Structured data types.....	23
4.4.1.1 Introduction	23
4.4.1.2 Void.....	23
4.4.1.3 Void.....	23
4.4.1.3a Void.....	23
4.4.1.4 Void.....	23
4.4.1.5 Type: NsInstanceSubscriptionFilter	23
4.4.1.6 Type: ResourceHandle	23
4.4.1.7 Void.....	24
4.4.2 Simple data types and enumerations	24
4.4.2.1 Introduction.....	24
4.4.2.2 Simple data types	24
4.4.2.3 Enumerations	24
4.5 Void.....	24
4.6 Void.....	24
4.7 Void.....	25
5 NSD Management interface	25
5.1 Description	25
5.1a API version.....	26
5.2 Resource structure and methods	26
5.3 Sequence diagrams (informative).....	28
5.3.1 Flow of the creation of an individual NS descriptor resource.....	28
5.3.2 Flow of the uploading of NSD archive content	29
5.3.3 Flow of the fetching of an onboarded NSD archive	30
5.3.4 Flow of the update of an individual NS descriptor resource.....	31
5.3.5 Flow of the deletion of an individual NS descriptor resource.....	32
5.3.6 Flow of the querying/reading of NS descriptor information.....	33
5.3.6a Flow of reading the NSD	34
5.3.6b Flow of fetching the manifest file of an onboarded NSD archive.....	34
5.3.7 Flow of the creation of an individual PNF descriptor resource	35
5.3.8 Flow of the uploading of PNFD archive	35
5.3.9 Flow of the fetching of an onboarded PNFD archive	36
5.3.10 Flow of the deletion of an individual PNF descriptor resource	37
5.3.11 Flow of the querying/reading of PNF descriptor information.....	37
5.3.11a Flow of reading the PNFD	39
5.3.11b Flow of fetching the manifest file of an onboarded PNFD archive	39
5.3.12 Flow of managing subscriptions	39
5.3.13 Flow of sending notifications.....	41

5.4	Resources	42
5.4.1	Introduction.....	42
5.4.1a	Resource: API versions.....	42
5.4.2	Resource: NS Descriptors.....	42
5.4.2.1	Description.....	42
5.4.2.2	Resource definition	42
5.4.2.3	Resource methods	43
5.4.2.3.1	POST	43
5.4.2.3.2	GET	43
5.4.2.3.3	PUT	44
5.4.2.3.4	PATCH.....	45
5.4.2.3.5	DELETE.....	45
5.4.3	Resource: Individual NS Descriptor	45
5.4.3.1	Description.....	45
5.4.3.2	Resource definition	45
5.4.3.3	Resource methods	45
5.4.3.3.1	POST	45
5.4.3.3.2	GET	45
5.4.3.3.3	PUT	46
5.4.3.3.4	PATCH.....	46
5.4.3.3.5	DELETE.....	47
5.4.4	Resource: NSD archive content.....	48
5.4.4.1	Description.....	48
5.4.4.2	Resource definition	48
5.4.4.3	Resource methods	49
5.4.4.3.1	POST	49
5.4.4.3.2	GET	49
5.4.4.3.3	PUT	50
5.4.4.3.4	PATCH.....	51
5.4.4.3.5	DELETE.....	51
5.4.4a	Resource: NSD	51
5.4.4a.1	Description.....	51
5.4.4a.2	Resource definition	52
5.4.4a.3	Resource methods	52
5.4.4a.3.1	POST	52
5.4.4a.3.2	GET	52
5.4.4a.3.3	PUT	53
5.4.4a.3.4	PATCH.....	54
5.4.4a.3.5	DELETE.....	54
5.4.4b	Resource: NSD archive manifest.....	54
5.4.4b.1	Description.....	54
5.4.4b.2	Resource definition	54
5.4.4b.3	Resource methods	54
5.4.4b.3.1	POST	54
5.4.4b.3.2	GET	54
5.4.4b.3.3	PUT	56
5.4.4b.3.4	PATCH.....	56
5.4.4b.3.5	DELETE.....	56
5.4.5	Resource: PNF Descriptors.....	56
5.4.5.1	Description.....	56
5.4.5.2	Resource definition	56
5.4.5.3	Resource methods	56
5.4.5.3.1	POST	56
5.4.5.3.2	GET	57
5.4.5.3.3	PUT	59
5.4.5.3.4	PATCH.....	59
5.4.5.3.5	DELETE.....	59
5.4.6	Resource: Individual PNF Descriptor.....	60
5.4.6.1	Description.....	60
5.4.6.2	Resource definition	60
5.4.6.3	Resource methods	60
5.4.6.3.1	POST	60

5.4.6.3.2	GET	60
5.4.6.3.3	PUT	61
5.4.6.3.4	PATCH	61
5.4.6.3.5	DELETE	61
5.4.7	Resource: PNFD archive content	62
5.4.7.1	Description	62
5.4.7.2	Resource definition	62
5.4.7.3	Resource methods	63
5.4.7.3.1	POST	63
5.4.7.3.2	GET	63
5.4.7.3.3	PUT	64
5.4.7.3.4	PATCH	65
5.4.7.3.5	DELETE	65
5.4.7a	Resource: PNFD	65
5.4.7a.1	Description	65
5.4.7a.2	Resource definition	66
5.4.7a.3	Resource methods	66
5.4.7a.3.1	POST	66
5.4.7a.3.2	GET	66
5.4.7a.3.3	PUT	67
5.4.7a.3.4	PATCH	68
5.4.7a.3.5	DELETE	68
5.4.7b	Resource: PNFD archive manifest	68
5.4.7b.1	Description	68
5.4.7b.2	Resource definition	68
5.4.7b.3	Resource methods	68
5.4.7b.3.1	POST	68
5.4.7b.3.2	GET	68
5.4.7b.3.3	PUT	70
5.4.7b.3.4	PATCH	70
5.4.7b.3.5	DELETE	70
5.4.8	Resource: Subscriptions	70
5.4.8.1	Description	70
5.4.8.2	Resource definition	70
5.4.8.3	Resource methods	70
5.4.8.3.1	POST	70
5.4.8.3.2	GET	71
5.4.8.3.3	PUT	72
5.4.8.3.4	PATCH	73
5.4.8.3.5	DELETE	73
5.4.9	Resource: Individual subscription	73
5.4.9.1	Description	73
5.4.9.2	Resource definition	73
5.4.9.3	Resource methods	73
5.4.9.3.1	POST	73
5.4.9.3.2	GET	73
5.4.9.3.3	PUT	74
5.4.9.3.4	PATCH	74
5.4.9.3.5	DELETE	74
5.4.10	Resource: Notification endpoint	74
5.4.10.1	Description	74
5.4.10.2	Resource definition	75
5.4.10.3	Resource methods	75
5.4.10.3.1	POST	75
5.4.10.3.2	GET	75
5.4.10.3.3	PUT	76
5.4.10.3.4	PATCH	76
5.4.10.3.5	DELETE	76
5.5	Data model	76
5.5.1	Introduction	76
5.5.2	Resource and notification data types	76
5.5.2.1	Type: NsdInfoModifications	76

5.5.2.2	Type: NsdInfo	77
5.5.2.3	Type: CreateNsdInfoRequest	78
5.5.2.4	Type: PnfdInfoModifications	78
5.5.2.5	Type: PnfdInfo	79
5.5.2.6	Type: CreatePnfdInfoRequest	80
5.5.2.7	Type: NsdmSubscriptionRequest	80
5.5.2.8	Type: NsdmSubscription	80
5.5.2.9	Type: NsdOnboardingNotification	80
5.5.2.10	Type: NsdOnboardingFailureNotification	81
5.5.2.11	Type: NsdChangeNotification	81
5.5.2.12	Type: NsdDeletionNotification	82
5.5.2.13	Type: PnfdOnboardingNotification	82
5.5.2.14	Type: PnfdOnboardingFailureNotification	83
5.5.2.15	Type: PnfdDeletionNotification	83
5.5.3	Referenced structured data types	84
5.5.3.1	Introduction	84
5.5.3.2	Type: NsdmNotificationsFilter	84
5.5.3.3	Type: NsdmLinks	85
5.5.3.4	Type: PnfdmLinks	86
5.5.3.5	Type: NsdArchiveArtifactInfo	86
5.5.3.6	Type: PnfdArchiveArtifactInfo	86
5.5.4	Referenced simple data types and enumerations	87
5.5.4.1	Introduction	87
5.5.4.2	Simple data types	87
5.5.4.3	Enumeration: NsdOperationalStateType	87
5.5.4.4	Enumeration: NsdUsageStateType	87
5.5.4.5	Enumeration: NsdOnboardingStateType	87
5.5.4.6	Enumeration: PnfdOnboardingStateType	87
5.5.4.7	Enumeration: PnfdUsageStateType	88
6	NS Lifecycle Management interface	88
6.1	Description	88
6.1a	API version	89
6.2	Resource structure and methods	89
6.3	Sequence diagrams (informative)	91
6.3.1	Flow of the creation of an NS instance resource	91
6.3.2	Flow of the deletion of an NS instance resource	91
6.3.3	Flow of NS lifecycle management operations triggered by task resources	92
6.3.4	Flow of the get operations status operation	94
6.3.5	Flow of managing subscriptions	95
6.3.6	Flow of sending notifications	96
6.3.7	Flow of retrying an NS lifecycle management operation	97
6.3.8	Flow of rolling back an NS lifecycle management operation	98
6.3.9	Flow of continuing an NS lifecycle management operation	100
6.3.10	Flow of failing an NS lifecycle management operation	101
6.3.11	Flow of cancelling an NS lifecycle management operation	102
6.4	Resources	103
6.4.1	Introduction	103
6.4.1.1	Overview	103
6.4.1.2	Task resources that trigger NS LCM operations	103
6.4.1a	Resource: API versions	104
6.4.2	Resource: NS Instances	104
6.4.2.1	Description	104
6.4.2.2	Resource definition	104
6.4.2.3	Resource methods	105
6.4.2.3.1	POST	105
6.4.2.3.2	GET	105
6.4.2.3.3	PUT	107
6.4.2.3.4	PATCH	107
6.4.2.3.5	DELETE	108
6.4.3	Resource: Individual NS Instance	108
6.4.3.1	Description	108

6.4.3.2	Resource definition	108
6.4.3.3	Resource methods	108
6.4.3.3.1	POST	108
6.4.3.3.2	GET	108
6.4.3.3.3	PUT	109
6.4.3.3.4	PATCH	109
6.4.3.3.5	DELETE	109
6.4.4	Resource: Instantiate NS task	110
6.4.4.1	Description	110
6.4.4.2	Resource definition	110
6.4.4.3	Resource methods	110
6.4.4.3.1	POST	110
6.4.4.3.2	GET	111
6.4.4.3.3	PUT	111
6.4.4.3.4	PATCH	111
6.4.4.3.5	DELETE	111
6.4.5	Resource: Scale NS task	112
6.4.5.1	Description	112
6.4.5.2	Resource definition	112
6.4.5.3	Resource methods	112
6.4.5.3.1	POST	112
6.4.5.3.2	GET	113
6.4.5.3.3	PUT	113
6.4.5.3.4	PATCH	113
6.4.5.3.5	DELETE	113
6.4.6	Resource: Update NS task	114
6.4.6.1	Description	114
6.4.6.2	Resource definition	114
6.4.6.3	Resource methods	114
6.4.6.3.1	POST	114
6.4.6.3.2	GET	115
6.4.6.3.3	PUT	115
6.4.6.3.4	PATCH	115
6.4.6.3.5	DELETE	115
6.4.7	Resource: Heal NS task	115
6.4.7.1	Description	115
6.4.7.2	Resource definition	116
6.4.7.3	Resource methods	116
6.4.7.3.1	POST	116
6.4.7.3.2	GET	117
6.4.7.3.3	PUT	117
6.4.7.3.4	PATCH	117
6.4.7.3.5	DELETE	117
6.4.8	Resource: Terminate NS task	117
6.4.8.1	Description	117
6.4.8.2	Resource definition	117
6.4.8.3	Resource methods	117
6.4.8.3.1	POST	117
6.4.8.3.2	GET	118
6.4.8.3.3	PUT	118
6.4.8.3.4	PATCH	118
6.4.8.3.5	DELETE	118
6.4.9	Resource: NS LCM operation occurrences	119
6.4.9.1	Description	119
6.4.9.2	Resource definition	119
6.4.9.3	Resource methods	119
6.4.9.3.1	POST	119
6.4.9.3.2	GET	119
6.4.9.3.3	PUT	120
6.4.9.3.4	PATCH	121
6.4.9.3.5	DELETE	121
6.4.10	Resource: Individual NS LCM operation occurrence	121

6.4.10.1	Description	121
6.4.10.2	Resource definition	121
6.4.10.3	Resource methods	121
6.4.10.3.1	POST	121
6.4.10.3.2	GET	121
6.4.10.3.3	PUT	122
6.4.10.3.4	PATCH	122
6.4.10.3.5	DELETE	122
6.4.11	Resource: Retry operation task	122
6.4.11.1	Description	122
6.4.11.2	Resource definition	122
6.4.11.3	Resource methods	123
6.4.11.3.1	POST	123
6.4.11.3.2	GET	124
6.4.11.3.3	PUT	124
6.4.11.3.4	PATCH	124
6.4.11.3.5	DELETE	124
6.4.12	Resource: Rollback operation task	124
6.4.12.1	Description	124
6.4.12.2	Resource definition	124
6.4.12.3	Resource methods	124
6.4.12.3.1	POST	124
6.4.12.3.2	GET	125
6.4.12.3.3	PUT	125
6.4.12.3.4	PATCH	126
6.4.12.3.5	DELETE	126
6.4.13	Resource: Continue operation task	126
6.4.13.1	Description	126
6.4.13.2	Resource definition	126
6.4.13.3	Resource methods	126
6.4.13.3.1	POST	126
6.4.13.3.2	GET	127
6.4.13.3.3	PUT	127
6.4.13.3.4	PATCH	127
6.4.13.3.5	DELETE	128
6.4.14	Resource: Fail operation task	128
6.4.14.1	Description	128
6.4.14.2	Resource definition	128
6.4.14.3	Resource methods	128
6.4.14.3.1	POST	128
6.4.14.3.2	GET	129
6.4.14.3.3	PUT	129
6.4.14.3.4	PATCH	129
6.4.14.3.5	DELETE	130
6.4.15	Resource: Cancel operation task	130
6.4.15.1	Description	130
6.4.15.2	Resource definition	130
6.4.15.3	Resource methods	130
6.4.15.3.1	POST	130
6.4.15.3.2	GET	131
6.4.15.3.3	PUT	131
6.4.15.3.4	PATCH	132
6.4.15.3.5	DELETE	132
6.4.16	Resource: Subscriptions	132
6.4.16.1	Description	132
6.4.16.2	Resource definition	132
6.4.16.3	Resource methods	132
6.4.16.3.1	POST	132
6.4.16.3.2	GET	133
6.4.16.3.3	PUT	134
6.4.16.3.4	PATCH	135
6.4.16.3.5	DELETE	135

6.4.17	Resource: Individual subscription.....	135
6.4.17.1	Description	135
6.4.17.2	Resource definition	135
6.4.17.3	Resource methods	135
6.4.17.3.1	POST	135
6.4.17.3.2	GET	135
6.4.17.3.3	PUT	136
6.4.17.3.4	PATCH.....	136
6.4.17.3.5	DELETE.....	136
6.4.18	Resource: Notification endpoint	137
6.4.18.1	Description	137
6.4.18.2	Resource definition	137
6.4.18.3	Resource methods	137
6.4.18.3.1	POST	137
6.4.18.3.2	GET	138
6.4.18.3.3	PUT	138
6.4.18.3.4	PATCH.....	138
6.4.18.3.5	DELETE.....	138
6.5	Data model	138
6.5.1	Introduction.....	138
6.5.2	Resource and notification data types	138
6.5.2.1	Introduction	138
6.5.2.2	Type: LccnSubscriptionRequest	139
6.5.2.3	Type: NsLcmOpOcc	139
6.5.2.4	Type: LccnSubscription	141
6.5.2.5	Type: NsLcmOperationOccurrenceNotification	141
6.5.2.6	Type: NsIdentifierCreationNotification	142
6.5.2.7	Type: NsIdentifierDeletionNotification.....	143
6.5.2.8	Type: NsChangeNotification.....	143
6.5.2.9	Type: CreateNsRequest.....	144
6.5.2.10	Type: NsInstance.....	145
6.5.2.11	Type: InstantiateNsRequest.....	146
6.5.2.12	Type: UpdateNsRequest.....	147
6.5.2.13	Type: HealNsRequest.....	149
6.5.2.14	Type: ScaleNsRequest	150
6.5.2.15	Type: TerminateNsRequest.....	150
6.5.2.16	Type: CancelMode.....	150
6.5.3	Referenced structured data types	150
6.5.3.1	Introduction	150
6.5.3.2	Type: AffectedVnf	151
6.5.3.3	Type: AffectedPnf	151
6.5.3.4	Type: AffectedVirtualLink.....	152
6.5.3.5	Type: AffectedVnffg.....	152
6.5.3.6	Type: AffectedNs	153
6.5.3.7	Type: AffectedSap	153
6.5.3.8	Type: LifecycleChangeNotificationsFilter.....	154
6.5.3.9	Type: LccnLinks	155
6.5.3.10	Type: SapData.....	155
6.5.3.11	Type: CpProtocolData.....	155
6.5.3.12	Type: IpOverEthernetAddressData	156
6.5.3.13	Type: PnfInfo	157
6.5.3.14	Type: AddPnfData	157
6.5.3.15	Type: ModifyPnfData	157
6.5.3.16	Type: PnfExtCpData	157
6.5.3.17	Type: PnfExtCpInfo	158
6.5.3.18	Type: IpOverEthernetAddressInfo	158
6.5.3.19	Type: VnfInstanceData	159
6.5.3.19a	Type: NestedNsInstanceData	159
6.5.3.20	Type: VnfLocationConstraint	159
6.5.3.21	Type: LocationConstraints	159
6.5.3.21a	Type: ParamsForNestedNs.....	160
6.5.3.22	Type: ParamsForVnf.....	160

6.5.3.23	Type: AffinityOrAntiAffinityRule	160
6.5.3.24	Type: InstantiateVnfData	161
6.5.3.25	Type: ChangeVnfFlavourData	162
6.5.3.26	Type: ExtVirtualLinkData	162
6.5.3.27	Type: ExtManagedVirtualLinkData	163
6.5.3.28	Type: ExtLinkPortData	163
6.5.3.29	Type: VnfExtCpData	163
6.5.3.30	Type: VnfExtCpConfig	164
6.5.3.31	Type: OperateVnfData	164
6.5.3.32	Type: ModifyVnfInfoData	165
6.5.3.33	Type: ChangeExtVnfConnectivityData	165
6.5.3.34	Type: AssocNewNsdVersionData	165
6.5.3.35	Type: MoveVnfInstanceData	166
6.5.3.36	Type: AddVnffgData	166
6.5.3.37	Type: UpdateVnffgData	166
6.5.3.38	Type: NfpData	167
6.5.3.39	Type: ChangeNsFlavourData	167
6.5.3.40	Type: NfpRule	167
6.5.3.41	Type: Mask	168
6.5.3.42	Type: PortRange	169
6.5.3.43	Type: HealNsData	169
6.5.3.44	Type: HealVnfData	169
6.5.3.45	Type: ScaleNsData	170
6.5.3.46	Type: ScaleNsByStepsData	170
6.5.3.47	Type: ScaleNsToLevelData	171
6.5.3.48	Type: NsScaleInfo	171
6.5.3.49	Type: ScaleVnfData	171
6.5.3.50	Type: ScaleToLevelData	172
6.5.3.51	Type: VnfScaleInfo	172
6.5.3.52	Type: ScaleByStepData	172
6.5.3.53	Type: NsVirtualLinkInfo	173
6.5.3.54	Void	173
6.5.3.55	Type: NsLinkPortInfo	173
6.5.3.56	Type: NsCpHandle	173
6.5.3.57	Type: VnfInstance	174
6.5.3.58	Type: CpProtocolInfo	177
6.5.3.59	Type: ExtManagedVirtualLinkInfo	177
6.5.3.60	Type: VnfcResourceInfo	178
6.5.3.61	Type: VnfVirtualLinkResourceInfo	178
6.5.3.62	Type: ExtVirtualLinkInfo	179
6.5.3.63	Type: ExtLinkPortInfo	179
6.5.3.64	Type: VnfLinkPortInfo	179
6.5.3.65	Type: VnffgInfo	180
6.5.3.66	Type: NfpInfo	181
6.5.3.67	Type: SapInfo	181
6.5.3.68	Type: NsMonitoringParameter	181
6.5.3.69	Type: VnfMonitoringParameter	182
6.5.3.70	Type: VnfExtCpInfo	182
6.5.3.71	Type: CpGroupInfo	182
6.5.3.72	Type: CpPairInfo	183
6.5.3.73	Type: ForwardingBehaviour InputParameters	183
6.5.4	Referenced simple data types and enumerations	184
6.5.4.1	Introduction	184
6.5.4.2	Simple data types	184
6.5.4.3	Enumeration: NsLcmOpType	184
6.5.4.4	Enumeration: NsLcmOperationStateType	184
6.5.4.5	Enumeration: NsComponentType	185
6.5.4.6	Enumeration: LcmOpNameForChangeNotificationType	185
6.5.4.7	Enumeration: LcmOpOccStatusForChangeNotificationType	185
6.5.4.8	Enumeration: OperationalStates	186
6.5.4.9	Enumeration: StopType	186
6.5.4.10	Enumeration: CancelModeType	186

6.6	Handling of errors during NS lifecycle management operations.....	187
6.6.1	Basic concepts (informative)	187
6.6.1.1	Motivation	187
6.6.1.2	Failure resolution strategies: Retry, Rollback and Continue	187
6.6.1.3	Error handling at NFVO and OSS/BSS	187
6.6.2	States and state transitions of an NS lifecycle management operation occurrence.....	189
6.6.2.1	General	189
6.6.2.2	States of an NS lifecycle management operation occurrence.....	189
6.6.2.3	Error handling operations that change the state of an NS lifecycle operation.....	192
6.6.3	Detailed flows	192
6.6.3.1	Immediate failure	192
6.6.3.2	Failure during actual NS LCM operation execution	193
6.6.3.3	LCM operation cancellation.....	194
7	NS Performance Management interface.....	195
7.1	Description	195
7.1a	API version.....	195
7.2	Resource structure and methods.....	195
7.3	Sequence diagrams (informative).....	196
7.3.1	Flow of creating a PM job	196
7.3.1a	Flow of updating the callback URI of a PM job	197
7.3.2	Flow of querying/reading PM jobs	198
7.3.3	Flow of deleting a PM job	199
7.3.4	Flow of obtaining performance reports.....	200
7.3.5	Flow of creating a threshold	201
7.3.5a	Flow of updating the callback URI of a threshold	201
7.3.6	Flow of querying/reading thresholds	202
7.3.7	Flow of deleting thresholds.....	203
7.3.8	Void	204
7.3.9	Flow of sending notifications.....	204
7.4	Resources	204
7.4.1	Introduction.....	204
7.4.1a	Resource: API versions.....	204
7.4.2	Resource: PM jobs	205
7.4.2.1	Description	205
7.4.2.2	Resource definition	205
7.4.2.3	Resource methods	205
7.4.2.3.1	POST	205
7.4.2.3.2	GET	206
7.4.2.3.3	PUT	207
7.4.2.3.4	PATCH.....	207
7.4.2.3.5	DELETE.....	208
7.4.3	Resource: Individual PM job	208
7.4.3.1	Description	208
7.4.3.2	Resource definition	208
7.4.3.3	Resource methods	208
7.4.3.3.1	POST	208
7.4.3.3.2	GET	208
7.4.3.3.3	PUT	209
7.4.3.3.4	PATCH.....	209
7.4.3.3.5	DELETE.....	210
7.4.4	Resource: Individual performance report	210
7.4.4.1	Description	210
7.4.4.2	Resource definition	210
7.4.4.3	Resource methods	211
7.4.4.3.1	POST	211
7.4.4.3.2	GET	211
7.4.4.3.3	PUT	211
7.4.4.3.4	PATCH.....	211
7.4.4.3.5	DELETE.....	211
7.4.5	Resource: Thresholds.....	211
7.4.5.1	Description	211

7.4.5.2	Resource definition	212
7.4.5.3	Resource methods	212
7.4.5.3.1	POST	212
7.4.5.3.2	GET	212
7.4.5.3.3	PUT	214
7.4.5.3.4	PATCH	214
7.4.5.3.5	DELETE	214
7.4.6	Resource: Individual threshold	214
7.4.6.1	Description	214
7.4.6.2	Resource definition	214
7.4.6.3	Resource methods	214
7.4.6.3.1	POST	214
7.4.6.3.2	GET	214
7.4.6.3.3	PUT	215
7.4.6.3.4	PATCH	215
7.4.6.3.5	DELETE	216
7.4.7	Void	216
7.4.8	Void	216
7.4.9	Resource: Notification endpoint	216
7.4.9.1	Description	216
7.4.9.2	Resource definition	216
7.4.9.3	Resource methods	217
7.4.9.3.1	POST	217
7.4.9.3.2	GET	217
7.4.9.3.3	PUT	218
7.4.9.3.4	PATCH	218
7.4.9.3.5	DELETE	218
7.5	Data Model	218
7.5.1	Introduction	218
7.5.2	Resource and notification data types	218
7.5.2.1	Introduction	218
7.5.2.2	Void	218
7.5.2.3	Void	218
7.5.2.4	Type: ThresholdCrossedNotification	218
7.5.2.5	Type: PerformanceInformationAvailableNotification	219
7.5.2.6	Type: CreatePmJobRequest	220
7.5.2.7	Type: PmJob	221
7.5.2.8	Type: CreateThresholdRequest	222
7.5.2.9	Type: Threshold	223
7.5.2.10	Type: PerformanceReport	224
7.5.2.11	Type: ThresholdModifications	224
7.5.2.12	Type: PmJobModifications	225
7.5.3	Referenced structured data types	225
7.5.3.1	Introduction	225
7.5.3.2	Void	225
7.5.3.3	Type: PmJobCriteria	225
7.5.3.4	Type: ThresholdCriteria	226
7.5.4	Referenced simple data types and enumerations	226
7.5.4.1	Introduction	226
7.5.4.2	Simple data types	226
7.5.4.3	Enumeration: CrossingDirectionType	227
8	NS Fault Management interface	227
8.1	Description	227
8.1a	API version	227
8.2	Resource structure and methods	227
8.3	Sequence diagrams (informative)	228
8.3.1	Flow of the Get Alarm List operation	228
8.3.2	Flow of acknowledging alarm	229
8.3.3	Flow of managing subscriptions	230
8.3.4	Flow of sending notifications	231
8.4	Resources	232

8.4.1	Introduction.....	232
8.4.1a	Resource: API versions.....	232
8.4.2	Resource: Alarms.....	232
8.4.2.1	Description.....	232
8.4.2.2	Resource definition.....	232
8.4.2.3	Resource methods.....	232
8.4.2.3.1	POST.....	232
8.4.2.3.2	GET.....	233
8.4.2.3.3	PUT.....	234
8.4.2.3.4	PATCH.....	234
8.4.2.3.5	DELETE.....	234
8.4.3	Resource: Individual alarm.....	234
8.4.3.1	Description.....	234
8.4.3.2	Resource definition.....	235
8.4.3.3	Resource methods.....	235
8.4.3.3.1	POST.....	235
8.4.3.3.2	GET.....	235
8.4.3.3.3	PUT.....	235
8.4.3.3.4	PATCH.....	236
8.4.3.3.5	DELETE.....	236
8.4.4	Resource: Subscriptions.....	237
8.4.4.1	Description.....	237
8.4.4.2	Resource definition.....	237
8.4.4.3	Resource methods.....	237
8.4.4.3.1	POST.....	237
8.4.4.3.2	GET.....	238
8.4.4.3.3	PUT.....	239
8.4.4.3.4	PATCH.....	240
8.4.4.3.5	DELETE.....	240
8.4.5	Resource: Individual subscription.....	240
8.4.5.1	Description.....	240
8.4.5.2	Resource definition.....	240
8.4.5.3	Resource methods.....	240
8.4.5.3.1	POST.....	240
8.4.5.3.2	GET.....	240
8.4.5.3.3	PUT.....	241
8.4.5.3.4	PATCH.....	241
8.4.5.3.5	DELETE.....	241
8.4.6	Resource: Notification endpoint.....	242
8.4.6.1	Description.....	242
8.4.6.2	Resource definition.....	242
8.4.6.3	Resource methods.....	242
8.4.6.3.1	POST.....	242
8.4.6.3.2	GET.....	242
8.4.6.3.3	PUT.....	243
8.4.6.3.4	PATCH.....	243
8.4.6.3.5	DELETE.....	243
8.5	Data Model.....	243
8.5.1	Introduction.....	243
8.5.2	Resource and notification data types.....	243
8.5.2.1	Introduction.....	243
8.5.2.2	Type: FmSubscriptionRequest.....	243
8.5.2.3	Type: FmSubscription.....	244
8.5.2.4	Type: Alarm.....	244
8.5.2.5	Type: AlarmNotification.....	245
8.5.2.6	Type: AlarmClearedNotification.....	245
8.5.2.7	Type: AlarmListRebuiltNotification.....	246
8.5.2.8	Type: AlarmModifications.....	246
8.5.3	Referenced structured data types.....	247
8.5.3.1	Introduction.....	247
8.5.3.2	Type: FmNotificationsFilter.....	247
8.5.3.3	Type: FaultyResourceInfo.....	247