



Network Functions Virtualisation (NFV) Release 3; Protocols and Data Models; RESTful protocols specification for the Or-Vnfm Reference Point

ETSI GS NFV-SOL 003 V3.6.1 (2022-01)

<https://standards.iteh.ai/catalog/standards/sist/1d5f726f-8e81-48af-8682-4e9afac284d1/etsi-gs-nfv-sol-003-v3-6-1-2022-01>

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-SOL003ed361

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 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

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/CommitteeSupportStaff.aspx>

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

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 2022.

All rights reserved.

Contents

Intellectual Property Rights	20
Foreword.....	20
Modal verbs terminology.....	20
1 Scope	21
2 References	21
2.1 Normative references	21
2.2 Informative references.....	22
3 Definition of terms, symbols and abbreviations.....	23
3.1 Terms.....	23
3.2 Symbols.....	23
3.3 Abbreviations	23
4 General aspects.....	24
4.1 Overview	24
4.2 Void.....	25
4.3 Void.....	25
4.4 Common data types.....	25
4.4.1 Structured data types.....	25
4.4.1.1 Introduction.....	25
4.4.1.2 Void.....	25
4.4.1.3 Void.....	25
4.4.1.3a Void.....	25
4.4.1.4 Void.....	25
4.4.1.5 Type: VnfInstanceSubscriptionFilter	25
4.4.1.6 Type: VimConnectionInfo	26
4.4.1.7 Type: ResourceHandle	27
4.4.1.8 Void.....	28
4.4.1.9 Void.....	28
4.4.1.10 Type: VnfExtCpData	28
4.4.1.10a Type: VnfExtCpConfig.....	28
4.4.1.10b Type: CpProtocolData.....	29
4.4.1.10c Type: IpOverEthernetAddressData	29
4.4.1.11 Type: ExtVirtualLinkData	31
4.4.1.12 Type: ExtManagedVirtualLinkData.....	31
4.4.1.13 Void.....	32
4.4.1.14 Type: ExtLinkPortData	32
4.4.2 Simple data types and enumerations.....	32
4.4.2.1 Introduction.....	32
4.4.2.2 Simple data types	32
4.4.2.3 Enumerations	33
4.4.2.3.1 Introduction	33
4.4.2.3.2 Enumeration: LcmCoordResultType.....	33
4.5 Void.....	33
4.6 Void.....	33
4.7 Void.....	33
5 VNF Lifecycle Management interface.....	33
5.1 Description	33
5.1a API version.....	34
5.2 Resource structure and methods.....	34
5.3 Sequence diagrams (informative).....	37
5.3.1 Flow of the creation of a VNF instance resource.....	37
5.3.2 Flow of the deletion of a VNF instance resource.....	38
5.3.3 Flow of VNF lifecycle management operations triggered by task resources.....	39
5.3.4 Flow of automatic invocation of VNF scaling and VNF healing.....	41
5.3.5 Flow of the Query VNF operation	43

5.3.6	Flow of the Modify VNF Information operation	44
5.3.7	Flow of the Get Operation Status operation.....	46
5.3.8	Flow of managing subscriptions	47
5.3.9	Flow of sending notifications.....	48
5.3.10	Flow of retrying a VNF lifecycle management operation.....	49
5.3.11	Flow of rolling back a VNF lifecycle management operation	50
5.3.12	Flow of failing a VNF lifecycle management operation.....	52
5.3.13	Flow of cancelling a VNF lifecycle management operation.....	53
5.3.14	Flow of creation of a VNF snapshot resource.....	55
5.3.15	Flow of the Query VNF Snapshot operation	55
5.3.16	Flow of modify VNF snapshot resource information	56
5.3.17	Flow of fetching the VNF state snapshot content	56
5.3.18	Flow of the deletion of a VNF snapshot resource.....	57
5.4	Resources	58
5.4.1	Introduction.....	58
5.4.1.1	Overview	58
5.4.1.2	Task resources that trigger VNF LCM operations	58
5.4.1a	Resource: API versions.....	60
5.4.2	Resource: VNF instances.....	60
5.4.2.1	Description	60
5.4.2.2	Resource definition	60
5.4.2.3	Resource methods	60
5.4.2.3.1	POST	60
5.4.2.3.2	GET	61
5.4.2.3.3	PUT	63
5.4.2.3.4	PATCH.....	63
5.4.2.3.5	DELETE	63
5.4.3	Resource: Individual VNF instance	63
5.4.3.1	Description	63
5.4.3.2	Resource definition	63
5.4.3.3	Resource methods	63
5.4.3.3.1	POST	63
5.4.3.3.2	GET	63
5.4.3.3.3	PUT	64
5.4.3.3.4	PATCH.....	64
5.4.3.3.5	DELETE.....	65
5.4.4	Resource: Instantiate VNF task	66
5.4.4.1	Description.....	66
5.4.4.2	Resource definition	66
5.4.4.3	Resource methods	66
5.4.4.3.1	POST	66
5.4.4.3.2	GET	68
5.4.4.3.3	PUT	68
5.4.4.3.4	PATCH.....	68
5.4.4.3.5	DELETE.....	68
5.4.5	Resource: Scale VNF task	68
5.4.5.1	Description.....	68
5.4.5.2	Resource definition	68
5.4.5.3	Resource methods	68
5.4.5.3.1	POST	68
5.4.5.3.2	GET	70
5.4.5.3.3	PUT	70
5.4.5.3.4	PATCH.....	70
5.4.5.3.5	DELETE.....	70
5.4.6	Resource: Scale VNF to Level task	70
5.4.6.1	Description.....	70
5.4.6.2	Resource definition	70
5.4.6.3	Resource methods	70
5.4.6.3.1	POST	70
5.4.6.3.2	GET	72
5.4.6.3.3	PUT	72
5.4.6.3.4	PATCH.....	72

5.4.6.3.5	DELETE	72
5.4.7	Resource: Change VNF Flavour task	72
5.4.7.1	Description	72
5.4.7.2	Resource definition	72
5.4.7.3	Resource methods	72
5.4.7.3.1	POST	72
5.4.7.3.2	GET	74
5.4.7.3.3	PUT	74
5.4.7.3.4	PATCH	74
5.4.7.3.5	DELETE	74
5.4.8	Resource: Terminate VNF task	74
5.4.8.1	Description	74
5.4.8.2	Resource definition	74
5.4.8.3	Resource methods	75
5.4.8.3.1	POST	75
5.4.8.3.2	GET	76
5.4.8.3.3	PUT	76
5.4.8.3.4	PATCH	76
5.4.8.3.5	DELETE	76
5.4.9	Resource: Heal VNF task	76
5.4.9.1	Description	76
5.4.9.2	Resource definition	76
5.4.9.3	Resource methods	77
5.4.9.3.1	POST	77
5.4.9.3.2	GET	78
5.4.9.3.3	PUT	78
5.4.9.3.4	PATCH	78
5.4.9.3.5	DELETE	78
5.4.10	Resource: Operate VNF task	78
5.4.10.1	Description	78
5.4.10.2	Resource definition	79
5.4.10.3	Resource methods	79
5.4.10.3.1	POST	79
5.4.10.3.2	GET	80
5.4.10.3.3	PUT	80
5.4.10.3.4	PATCH	80
5.4.10.3.5	DELETE	80
5.4.11	Resource: Change external VNF connectivity task	81
5.4.11.1	Description	81
5.4.11.2	Resource definition	81
5.4.11.3	Resource methods	81
5.4.11.3.1	POST	81
5.4.11.3.2	GET	82
5.4.11.3.3	PUT	82
5.4.11.3.4	PATCH	82
5.4.11.3.5	DELETE	83
5.4.11a	Resource: Change current VNF package task	83
5.4.11a.1	Description	83
5.4.11a.2	Resource definition	84
5.4.11a.3	Resource methods	84
5.4.11a.3.1	POST	84
5.4.11a.3.2	GET	86
5.4.11a.3.3	PUT	86
5.4.11a.3.4	PATCH	86
5.4.11a.3.5	DELETE	86
5.4.12	Resource: VNF LCM operation occurrences	87
5.4.12.1	Description	87
5.4.12.2	Resource definition	87
5.4.12.3	Resource methods	87
5.4.12.3.1	POST	87
5.4.12.3.2	GET	87
5.4.12.3.3	PUT	89

5.4.12.3.4	PATCH	89
5.4.12.3.5	DELETE	89
5.4.13	Resource: Individual VNF LCM operation occurrence	90
5.4.13.1	Description	90
5.4.13.2	Resource definition	90
5.4.13.3	Resource methods	90
5.4.13.3.1	POST	90
5.4.13.3.2	GET	90
5.4.13.3.3	PUT	91
5.4.13.3.4	PATCH	91
5.4.13.3.5	DELETE	91
5.4.14	Resource: Retry operation task	91
5.4.14.1	Description	91
5.4.14.2	Resource definition	91
5.4.14.3	Resource methods	92
5.4.14.3.1	POST	92
5.4.14.3.2	GET	93
5.4.14.3.3	PUT	93
5.4.14.3.4	PATCH	93
5.4.14.3.5	DELETE	93
5.4.15	Resource: Rollback operation task	93
5.4.15.1	Description	93
5.4.15.2	Resource definition	93
5.4.15.3	Resource methods	93
5.4.15.3.1	POST	93
5.4.15.3.2	GET	94
5.4.15.3.3	PUT	94
5.4.15.3.4	PATCH	95
5.4.15.3.5	DELETE	95
5.4.16	Resource: Fail operation task	95
5.4.16.1	Description	95
5.4.16.2	Resource definition	95
5.4.16.3	Resource methods	95
5.4.16.3.1	POST	95
5.4.16.3.2	GET	96
5.4.16.3.3	PUT	96
5.4.16.3.4	PATCH	96
5.4.16.3.5	DELETE	97
5.4.17	Resource: Cancel operation task	97
5.4.17.1	Description	97
5.4.17.2	Resource definition	97
5.4.17.3	Resource methods	97
5.4.17.3.1	POST	97
5.4.17.3.2	GET	98
5.4.17.3.3	PUT	98
5.4.17.3.4	PATCH	99
5.4.17.3.5	DELETE	99
5.4.18	Resource: Subscriptions	99
5.4.18.1	Description	99
5.4.18.2	Resource definition	99
5.4.18.3	Resource methods	99
5.4.18.3.1	POST	99
5.4.18.3.2	GET	100
5.4.18.3.3	PUT	101
5.4.18.3.4	PATCH	102
5.4.18.3.5	DELETE	102
5.4.19	Resource: Individual subscription	102
5.4.19.1	Description	102
5.4.19.2	Resource definition	102
5.4.19.3	Resource methods	102
5.4.19.3.1	POST	102
5.4.19.3.2	GET	102

5.4.19.3.3	PUT	103
5.4.19.3.4	PATCH	103
5.4.19.3.5	DELETE	103
5.4.20	Resource: Notification endpoint	104
5.4.20.1	Description	104
5.4.20.2	Resource definition	104
5.4.20.3	Resource methods	104
5.4.20.3.1	POST	104
5.4.20.3.2	GET	104
5.4.20.3.3	PUT	105
5.4.20.3.4	PATCH	105
5.4.20.3.5	DELETE	105
5.4.21	Resource: Create VNF snapshot task	105
5.4.21.1	Description	105
5.4.21.2	Resource definition	105
5.4.21.3	Resource methods	106
5.4.21.3.1	POST	106
5.4.21.3.2	GET	107
5.4.21.3.3	PUT	108
5.4.21.3.4	PATCH	108
5.4.21.3.5	DELETE	108
5.4.22	Resource: Revert to VNF snapshot task	108
5.4.22.1	Description	108
5.4.22.2	Resource definition	108
5.4.22.3	Resource methods	109
5.4.22.3.1	POST	109
5.4.22.3.2	GET	110
5.4.22.3.3	PUT	110
5.4.22.3.4	PATCH	110
5.4.22.3.5	DELETE	110
5.4.23	Resource: VNF snapshots	110
5.4.23.1	Description	110
5.4.23.2	Resource definition	110
5.4.23.3	Resource methods	110
5.4.23.3.1	POST	110
5.4.23.3.2	GET	111
5.4.23.3.3	PUT	113
5.4.23.3.4	PATCH	113
5.4.23.3.5	DELETE	113
5.4.24	Resource: Individual VNF snapshot	113
5.4.24.1	Description	113
5.4.24.2	Resource definition	113
5.4.24.3	Resource methods	114
5.4.24.3.1	POST	114
5.4.24.3.2	GET	114
5.4.24.3.3	PUT	114
5.4.24.3.4	PATCH	114
5.4.24.3.5	DELETE	115
5.4.25.1	Description	116
5.4.25.2	Resource definition	116
5.4.25.3	Resource methods	117
5.4.25.3.1	POST	117
5.4.25.3.2	GET	117
5.4.25.3.3	PUT	118
5.4.25.3.4	PATCH	118
5.4.25.3.5	DELETE	118
5.5	Data model	118
5.5.1	Introduction	118
5.5.2	Resource and notification data types	118
5.5.2.1	Introduction	118
5.5.2.2	Type: VnfInstance	118
5.5.2.3	Type: CreateVnfRequest	124

5.5.2.4	Type: InstantiateVnfRequest	124
5.5.2.5	Type: ScaleVnfRequest	125
5.5.2.6	Type: ScaleVnfToLevelRequest	126
5.5.2.7	Type: ChangeVnfFlavourRequest	126
5.5.2.8	Type: TerminateVnfRequest	128
5.5.2.9	Type: HealVnfRequest	128
5.5.2.10	Type: OperateVnfRequest	129
5.5.2.11	Type: ChangeExtVnfConnectivityRequest	129
5.5.2.11a	Type: ChangeCurrentVnfPkgRequest	130
5.5.2.12	Type: VnfInfoModificationRequest	132
5.5.2.12a	Type: VnfInfoModifications	132
5.5.2.13	Type: VnfLcmOpOcc	133
5.5.2.14	Type: CancelMode	136
5.5.2.15	Type: LccnSubscriptionRequest	136
5.5.2.16	Type: LccnSubscription	137
5.5.2.17	Type: VnfLcmOperationOccurrenceNotification	137
5.5.2.18	Type: VnfIdentifierCreationNotification	140
5.5.2.19	Type: VnfIdentifierDeletionNotification	140
5.5.2.20	Type: CreateVnfSnapshotInfoRequest	141
5.5.2.21	Type: CreateVnfSnapshotRequest	141
5.5.2.22	Type: VnfSnapshotInfo	141
5.5.2.23	Type: VnfSnapshot	142
5.5.2.24	Type: VnfSnapshotInfoModificationRequest	143
5.5.2.25	Type: VnfSnapshotInfoModifications	143
5.5.2.26	Type: RevertToVnfSnapshotRequest	143
5.5.3	Referenced structured data types	143
5.5.3.1	Introduction	143
5.5.3.2	Type: ExtVirtualLinkInfo	144
5.5.3.3	Type: ExtManagedVirtualLinkInfo	144
5.5.3.4	Type: ScaleInfo	144
5.5.3.5	Type: VnfcResourceInfo	145
5.5.3.6	Type: VnfVirtualLinkResourceInfo	146
5.5.3.7	Type: VirtualStorageResourceInfo	146
5.5.3.8	Type: VnfLinkPortInfo	147
5.5.3.9	Type: ExtLinkPortInfo	148
5.5.3.9a	Void	149
5.5.3.9b	Type: CpProtocolInfo	149
5.5.3.10	Type: IpOverEthernetAddressInfo	149
5.5.3.11	Type: MonitoringParameter	150
5.5.3.12	Type: LifecycleChangeNotificationsFilter	150
5.5.3.13	Type: AffectedVnfc	151
5.5.3.14	Type: AffectedVirtualLink	153
5.5.3.14a	Type: AffectedExtLinkPort	154
5.5.3.14b	Type: AffectedVipCp	154
5.5.3.15	Type: AffectedVirtualStorage	155
5.5.3.16	Type: LccnLinks	156
5.5.3.17	Type: VnfExtCpInfo	156
5.5.3.18	Type: VnfLinkPortData	157
5.5.3.19	Type: VnfcSnapshotInfo	157
5.5.3.20	Type: VnfStateSnapshotInfo	158
5.5.3.21	Type: ModificationsTriggeredByVnfPkgChange	159
5.5.3.22	Type: VipCpInfo	160
5.5.4	Referenced simple data types and enumerations	160
5.5.4.1	Introduction	160
5.5.4.2	Simple data types	160
5.5.4.3	Enumeration: VnfOperationalStateType	160
5.5.4.4	Enumeration: StopType	161
5.5.4.5	Enumeration: LcmOperationStateType	161
5.5.4.6	Enumeration: CancelModeType	161
5.5.4.7	Enumeration: LcmOperationType	162
5.5.4.8	Enumeration: LcmOpOccNotificationVerbosityType	162
5.6	Success and error states of VNF lifecycle management operations	162

5.6.1	Basic concepts for error handling (informative)	162
5.6.1.1	Motivation	162
5.6.1.2	Failure resolution strategies: Retry and Rollback	163
5.6.1.3	Error handling at VNFM and NFVO	163
5.6.2	States and state transitions of a VNF lifecycle management operation occurrence.....	164
5.6.2.1	General	164
5.6.2.2	States of a VNF lifecycle management operation occurrence.....	165
5.6.2.3	Error handling operations that change the state of a VNF lifecycle management operation occurrence	168
5.6.3	Detailed flows for error handling.....	169
5.6.3.1	Immediate failure	169
5.6.3.2	Failure in "STARTING" state	169
5.6.3.3	Failure during actual LCM operation execution	170
5.6.3.4	LCM operation cancellation.....	172
5.7	Handling of security-sensitive attributes	172
6	VNF Performance Management interface.....	172
6.1	Description	172
6.1a	API version.....	172
6.2	Resource structure and methods.....	173
6.3	Sequence diagrams (informative).....	174
6.3.1	Flow of creating a PM job	174
6.3.1a	Flow of updating the callback URI of a PM job	175
6.3.2	Flow of querying/reading PM jobs	175
6.3.3	Flow of deleting a PM job	176
6.3.4	Flow of obtaining performance reports.....	177
6.3.5	Flow of creating a threshold	178
6.3.5a	Flow of updating the callback URI of a threshold	179
6.3.6	Flow of querying/reading thresholds	179
6.3.7	Flow of deleting thresholds.....	180
6.3.8	Void	181
6.3.9	Flow of sending notifications.....	181
6.4	Resources	181
6.4.1	Introduction.....	181
6.4.1a	Resource: API versions.....	181
6.4.2	Resource: PM jobs	182
6.4.2.1	Description	182
6.4.2.2	Resource definition	182
6.4.2.3	Resource methods	182
6.4.2.3.1	POST	182
6.4.2.3.2	GET	183
6.4.2.3.3	PUT	185
6.4.2.3.4	PATCH.....	185
6.4.2.3.5	DELETE.....	185
6.4.3	Resource: Individual PM job	185
6.4.3.1	Description	185
6.4.3.2	Resource definition	185
6.4.3.3	Resource methods	185
6.4.3.3.1	POST	185
6.4.3.3.2	GET	185
6.4.3.3.3	PUT	186
6.4.3.3.4	PATCH.....	186
6.4.3.3.5	DELETE	187
6.4.4	Resource: Individual performance report	187
6.4.4.1	Description	187
6.4.4.2	Resource definition	188
6.4.4.3	Resource methods	188
6.4.4.3.1	POST	188
6.4.4.3.2	GET	188
6.4.4.3.3	PUT	188
6.4.4.3.4	PATCH.....	189
6.4.4.3.5	DELETE.....	189

6.4.5	Resource: Thresholds.....	189
6.4.5.1	Description	189
6.4.5.2	Resource definition	189
6.4.5.3	Resource methods	189
6.4.5.3.1	POST	189
6.4.5.3.2	GET	190
6.4.5.3.3	PUT	191
6.4.5.3.4	PATCH.....	191
6.4.5.3.5	DELETE	191
6.4.6	Resource: Individual threshold	191
6.4.6.1	Description	191
6.4.6.2	Resource definition	192
6.4.6.3	Resource methods	192
6.4.6.3.1	POST	192
6.4.6.3.2	GET	192
6.4.6.3.3	PUT	192
6.4.6.3.4	PATCH.....	192
6.4.6.3.5	DELETE	193
6.4.7	Void	194
6.4.8	Void	194
6.4.9	Resource: Notification endpoint	194
6.4.9.1	Description	194
6.4.9.2	Resource definition	194
6.4.9.3	Resource methods	194
6.4.9.3.1	POST	194
6.4.9.3.2	GET	195
6.4.9.3.3	PUT	195
6.4.9.3.4	PATCH.....	196
6.4.9.3.5	DELETE	196
6.5	Data Model.....	196
6.5.1	Introduction.....	196
6.5.2	Resource and notification data types	196
6.5.2.1	Introduction.....	196
6.5.2.2	Void.....	196
6.5.2.3	Void.....	196
6.5.2.4	Type: ThresholdCrossedNotification	196
6.5.2.5	Type: PerformanceInformationAvailableNotification	197
6.5.2.6	Type: CreatePmJobRequest	198
6.5.2.7	Type: PmJob	199
6.5.2.8	Type: CreateThresholdRequest	199
6.5.2.9	Type: Threshold	200
6.5.2.10	Type: PerformanceReport	201
6.5.2.11	Type: ThresholdModifications	201
6.5.2.12	Type: PmJobModifications	202
6.5.3	Referenced structured data types	202
6.5.3.1	Introduction.....	202
6.5.3.2	Void.....	202
6.5.3.3	Type: PmJobCriteria	202
6.5.3.4	Type: ThresholdCriteria	203
6.5.4	Referenced simple data types and enumerations	203
6.5.4.1	Introduction.....	203
6.5.4.2	Simple data types	203
6.5.4.3	Enumeration: CrossingDirectionType.....	204
7	VNF Fault Management interface.....	204
7.1	Description	204
7.1a	API version.....	204
7.2	Resource structure and methods.....	205
7.3	Sequence diagrams (informative).....	205
7.3.1	Flow of the Get Alarm List operation.....	205
7.3.2	Flow of acknowledging alarm	206
7.3.3	Flow of managing subscriptions	207

7.3.4	Flow of sending notifications.....	208
7.4	Resources	209
7.4.1	Introduction.....	209
7.4.2	Resource: Alarms.....	209
7.4.2.1	Description	209
7.4.2.2	Resource definition	209
7.4.2.3	Resource methods	209
7.4.2.3.1	POST	209
7.4.2.3.2	GET	209
7.4.2.3.3	PUT	210
7.4.2.3.4	PATCH.....	211
7.4.2.3.5	DELETE.....	211
7.4.3	Resource: Individual alarm.....	211
7.4.3.1	Description	211
7.4.3.2	Resource definition	211
7.4.3.3	Resource methods	211
7.4.3.3.1	POST	211
7.4.3.3.2	GET	211
7.4.3.3.3	PUT	212
7.4.3.3.4	PATCH.....	212
7.4.3.3.5	DELETE.....	213
7.4.4	Resource: Subscriptions.....	213
7.4.4.1	Description	213
7.4.4.2	Resource definition	213
7.4.4.3	Resource methods	213
7.4.4.3.1	POST	213
7.4.4.3.2	GET	215
7.4.4.3.3	PUT	216
7.4.4.3.4	PATCH.....	216
7.4.4.3.5	DELETE.....	216
7.4.5	Resource: Individual subscription.....	216
7.4.5.1	Description	216
7.4.5.2	Resource definition	216
7.4.5.3	Resource methods	216
7.4.5.3.1	POST	216
7.4.5.3.2	GET	216
7.4.5.3.3	PUT	217
7.4.5.3.4	PATCH.....	217
7.4.5.3.5	DELETE.....	217
7.4.6	Resource: Notification endpoint	218
7.4.6.1	Description	218
7.4.6.2	Resource definition	218
7.4.6.3	Resource methods	218
7.4.6.3.1	POST	218
7.4.6.3.2	GET	219
7.4.6.3.3	PUT	219
7.4.6.3.4	PATCH.....	219
7.4.6.3.5	DELETE.....	219
7.5	Data Model.....	220
7.5.1	Introduction.....	220
7.5.2	Resource and notification data types	220
7.5.2.1	Introduction.....	220
7.5.2.2	Type: FmSubscriptionRequest	220
7.5.2.3	Type: FmSubscription.....	220
7.5.2.4	Type: Alarm	220
7.5.2.5	Type: AlarmNotification.....	222
7.5.2.6	Type: AlarmClearedNotification.....	222
7.5.2.7	Type: AlarmListRebuiltNotification	223
7.5.2.8	Type: AlarmModifications.....	223
7.5.3	Referenced structured data types	224
7.5.3.1	Introduction.....	224
7.5.3.2	Type: FmNotificationsFilter.....	224

7.5.3.3	Type: FaultyResourceInfo	224
7.5.4	Referenced simple data types and enumerations	224
7.5.4.1	Introduction	224
7.5.4.2	Simple data types	225
7.5.4.3	Enumeration: PerceivedSeverityType	225
7.5.4.4	Enumeration: EventType	225
7.5.4.5	Enumeration: FaultyResourceType	225
8	VNF Indicator interface	226
8.1	Description	226
8.1a	API version	226
8.2	Resource structure and methods	226
8.3	Sequence diagrams (informative)	227
8.3.1	Flow of querying VNF indicators	227
8.3.2	Flow of reading a VNF indicator	228
8.3.3	Flow of managing subscriptions	229
8.3.4	Flow of sending notifications	230
8.4	Resources	231
8.4.1	Introduction	231
8.4.1a	Resource: API versions	231
8.4.2	Resource: VNF indicators	231
8.4.2.1	Description	231
8.4.2.2	Resource definition	231
8.4.2.3	Resource methods	231
8.4.2.3.1	POST	231
8.4.2.3.2	GET	231
8.4.2.3.3	PUT	232
8.4.2.3.4	PATCH	233
8.4.2.3.5	DELETE	233
8.4.3	Resource: VNF indicators related to a VNF instance	233
8.4.3.1	Description	233
8.4.3.2	Resource definition	233
8.4.3.3	Resource methods	233
8.4.3.3.1	POST	233
8.4.3.3.2	GET	233
8.4.3.3.3	PUT	234
8.4.3.3.4	PATCH	235
8.4.3.3.5	DELETE	235
8.4.4	Resource: Individual VNF indicator	235
8.4.4.1	Description	235
8.4.4.2	Resource definition	235
8.4.4.3	Resource methods	235
8.4.4.3.1	POST	235
8.4.4.3.2	GET	235
8.4.4.3.3	PUT	236
8.4.4.3.4	PATCH	236
8.4.4.3.5	DELETE	236
8.4.5	Resource: Subscriptions	236
8.4.5.1	Description	236
8.4.5.2	Resource definition	236
8.4.5.3	Resource methods	236
8.4.5.3.1	POST	236
8.4.5.3.2	GET	238
8.4.5.3.3	PUT	239
8.4.5.3.4	PATCH	239
8.4.5.3.5	DELETE	239
8.4.6	Resource: Individual subscription	239
8.4.6.1	Description	239
8.4.6.2	Resource definition	239
8.4.6.3	Resource methods	240
8.4.6.3.1	POST	240
8.4.6.3.2	GET	240

8.4.6.3.3	PUT	240
8.4.6.3.4	PATCH	240
8.4.6.3.5	DELETE	240
8.4.7	Resource: Notification endpoint	241
8.4.7.1	Description	241
8.4.7.2	Resource definition	241
8.4.7.3	Resource methods	241
8.4.7.3.1	POST	241
8.4.7.3.2	GET	242
8.4.7.3.3	PUT	242
8.4.7.3.4	PATCH	242
8.4.7.3.5	DELETE	242
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: VnfIndicator	243
8.5.2.3	Type: VnfIndicatorSubscriptionRequest	243
8.5.2.4	Type: VnfIndicatorSubscription	243
8.5.2.5	Type: VnfIndicatorValueChangeNotification	244
8.5.2.6	Type: SupportedIndicatorsChangeNotification	244
8.5.3	Referenced structured data types	245
8.5.3.1	Introduction	245
8.5.3.2	Type: VnfIndicatorNotificationsFilter	245
8.5.4	Referenced simple data types and enumerations	245
9	VNF Lifecycle Operation Granting interface	246
9.1	Description	246
9.1a	API version	247
9.2	Resource structure and methods	247
9.3	Sequence diagrams (informative)	248
9.3.1	Flow of grant request with synchronous response	248
9.3.2	Flow of grant request with asynchronous response	248
9.4	Resources	249
9.4.1	Introduction	249
9.4.1a	Resource: API versions	249
9.4.2	Resource: Grants	250
9.4.2.1	Description	250
9.4.2.2	Resource definition	250
9.4.2.3	Resource methods	250
9.4.2.3.1	POST	250
9.4.2.3.2	GET	251
9.4.2.3.3	PUT	251
9.4.2.3.4	PATCH	251
9.4.2.3.5	DELETE	251
9.4.3	Resource: Individual grant	251
9.4.3.1	Description	251
9.4.3.2	Resource definition	252
9.4.3.3	Resource methods	252
9.4.3.3.1	POST	252
9.4.3.3.2	GET	252
9.4.3.3.3	PUT	253
9.4.3.3.4	PATCH	253
9.4.3.3.5	DELETE	253
9.5	Data model	253
9.5.1	Introduction	253
9.5.2	Resource and notification data types	253
9.5.2.1	Introduction	253
9.5.2.2	Type: GrantRequest	254
9.5.2.3	Type: Grant	256
9.5.3	Referenced structured data types	258
9.5.3.1	Introduction	258

9.5.3.2	Type: ResourceDefinition	258
9.5.3.3	Type: GrantInfo.....	259
9.5.3.4	Type: ZoneInfo	260
9.5.3.5	Type: ZoneGroupInfo	261
9.5.3.6	Type: PlacementConstraint	261
9.5.3.7	Type: VimConstraint.....	262
9.5.3.8	Type: ConstraintResourceRef	262
9.5.3.9	Type: VimComputeResourceFlavour.....	263
9.5.3.10	Type: VimSoftwareImage	263
9.5.3.11	Type: SnapshotResourceDefinition.....	264
9.5.4	Referenced simple data types and enumerations	266
9.5.4.1	Introduction.....	266
9.5.4.2	Simple data types	266
9.5.4.3	Enumeration: GrantedLcmOperationType	266
10	VNF Package Management interface	266
10.1	Description	266
10.1a	API version.....	267
10.2	Resource structure and methods	267
10.3	Sequence diagrams (informative).....	269
10.3.1	Flow of querying/reading VNF package information	269
10.3.2	Flow of reading the VNFD of an on-boarded VNF package	270
10.3.2a	Flow of fetching the VNF package manifest	270
10.3.3	Flow of fetching an on-boarded VNF package	271
10.3.4	Flow of fetching a VNF package artifact.....	271
10.3.4a	Flow of bulk-fetching VNF package artifacts that are not images.....	272
10.3.5	Flow of managing subscriptions	273
10.3.6	Flow of sending notifications.....	275
10.4	Resources	276
10.4.1	Introduction.....	276
10.4.1a	Resource: API versions.....	276
10.4.2	Resource: VNF packages.....	276
10.4.2.1	Description	276
10.4.2.2	Resource definition	276
10.4.2.3	Resource methods	276
10.4.2.3.1	POST	276
10.4.2.3.2	GET	276
10.4.2.3.3	PUT	278
10.4.2.3.4	PATCH.....	278
10.4.2.3.5	DELETE.....	278
10.4.3	Resource: Individual VNF package	278
10.4.3.1	Description	278
10.4.3.2	Resource definition	278
10.4.3.3	Resource methods	279
10.4.3.3.1	POST	279
10.4.3.3.2	GET	279
10.4.3.3.3	PUT	279
10.4.3.3.4	PATCH.....	279
10.4.3.3.5	DELETE.....	280
10.4.4	Resource: VNFD in an individual VNF package.....	280
10.4.4.1	Description	280
10.4.4.2	Resource definition	280
10.4.4.3	Resource methods	280
10.4.4.3.1	POST	280
10.4.4.3.2	GET	280
10.4.4.3.3	PUT	282
10.4.4.3.4	PATCH.....	283
10.4.4.3.5	DELETE.....	283
10.4.4a	Resource: Manifest in an individual VNF package	283
10.4.4a.1	Description	283
10.4.4a.2	Resource definition	283
10.4.4a.3	Resource methods	283