



Network Functions Virtualisation (NFV) Release 2; Protocols and Data Models; RESTful protocols specification for the Or-Vnfm 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.

ReferenceRGS/NFV-SOL003ed271

KeywordsAPI, 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 2019.

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	17
Foreword.....	17
Modal verbs terminology.....	17
1 Scope	18
2 References	18
2.1 Normative references	18
2.2 Informative references.....	19
3 Definition of terms, symbols and abbreviations.....	19
3.1 Terms.....	19
3.2 Symbols.....	20
3.3 Abbreviations	20
4 General aspects.....	20
4.1 Overview	20
4.2 Void.....	21
4.3 Void.....	21
4.4 Common data types	21
4.4.1 Structured data types.....	21
4.4.1.1 Introduction.....	21
4.4.1.2 Void.....	22
4.4.1.3 Void.....	22
4.4.1.3a Void.....	22
4.4.1.4 Void.....	22
4.4.1.5 Type: VnfInstanceSubscriptionFilter.....	22
4.4.1.6 Type: VimConnectionInfo.....	22
4.4.1.7 Type: ResourceHandle.....	23
4.4.1.8 Void.....	24
4.4.1.9 Void.....	24
4.4.1.10 Type: VnfExtCpData	24
4.4.1.10a Type: VnfExtCpConfig.....	24
4.4.1.10b Type: CpProtocolData.....	25
4.4.1.10c Type: IpOverEthernetAddressData.....	25
4.4.1.11 Type: ExtVirtualLinkData.....	26
4.4.1.12 Type: ExtManagedVirtualLinkData.....	27
4.4.1.13 Void.....	27
4.4.1.14 Type: ExtLinkPortData	27
4.4.2 Simple data types and enumerations.....	28
4.4.2.1 Introduction.....	28
4.4.2.2 Simple data types	28
4.4.2.3 Enumerations	28
4.5 Void.....	28
4.6 Void.....	28
4.7 Void.....	28
5 VNF Lifecycle Management interface.....	28
5.1 Description	28
5.1a API version.....	29
5.2 Resource structure and methods.....	29
5.3 Sequence diagrams (informative).....	32
5.3.1 Flow of the creation of a VNF instance resource.....	32
5.3.2 Flow of the deletion of a VNF instance resource.....	33
5.3.3 Flow of VNF lifecycle management operations triggered by task resources.....	33
5.3.4 Flow of automatic invocation of VNF scaling and VNF healing.....	35
5.3.5 Flow of the Query VNF operation.....	37
5.3.6 Flow of the Modify VNF Information operation.....	38
5.3.7 Flow of the Get Operation Status operation.....	40

5.3.8	Flow of managing subscriptions	41
5.3.9	Flow of sending notifications.....	42
5.3.10	Flow of retrying a VNF lifecycle management operation.....	43
5.3.11	Flow of rolling back a VNF lifecycle management operation.....	44
5.3.12	Flow of failing a VNF lifecycle management operation.....	46
5.3.13	Flow of cancelling a VNF lifecycle management operation.....	47
5.4	Resources	48
5.4.1	Introduction.....	48
5.4.1.1	Overview.....	48
5.4.1.2	Task resources that trigger VNF LCM operations	48
5.4.1a	Resource: API versions.....	50
5.4.2	Resource: VNF instances.....	50
5.4.2.1	Description	50
5.4.2.2	Resource definition	50
5.4.2.3	Resource methods	51
5.4.2.3.1	POST	51
5.4.2.3.2	GET	51
5.4.2.3.3	PUT	53
5.4.2.3.4	PATCH.....	53
5.4.2.3.5	DELETE.....	53
5.4.3	Resource: Individual VNF instance	54
5.4.3.1	Description	54
5.4.3.2	Resource definition	54
5.4.3.3	Resource methods	54
5.4.3.3.1	POST	54
5.4.3.3.2	GET	54
5.4.3.3.3	PUT	55
5.4.3.3.4	PATCH.....	55
5.4.3.3.5	DELETE.....	56
5.4.4	Resource: Instantiate VNF task	56
5.4.4.1	Description	56
5.4.4.2	Resource definition	56
5.4.4.3	Resource methods	57
5.4.4.3.1	POST	57
5.4.4.3.2	GET	58
5.4.4.3.3	PUT	58
5.4.4.3.4	PATCH.....	58
5.4.4.3.5	DELETE.....	58
5.4.5	Resource: Scale VNF task.....	58
5.4.5.1	Description	58
5.4.5.2	Resource definition	58
5.4.5.3	Resource methods	58
5.4.5.3.1	POST	58
5.4.5.3.2	GET	60
5.4.5.3.3	PUT	60
5.4.5.3.4	PATCH.....	60
5.4.5.3.5	DELETE.....	60
5.4.6	Resource: Scale VNF to Level task	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	62
5.4.6.3.3	PUT	62
5.4.6.3.4	PATCH.....	62
5.4.6.3.5	DELETE.....	62
5.4.7	Resource: Change VNF Flavour task	62
5.4.7.1	Description	62
5.4.7.2	Resource definition	62
5.4.7.3	Resource methods	62
5.4.7.3.1	POST	62
5.4.7.3.2	GET	64

5.4.7.3.3	PUT	64
5.4.7.3.4	PATCH	64
5.4.7.3.5	DELETE	64
5.4.8	Resource: Terminate VNF task	64
5.4.8.1	Description	64
5.4.8.2	Resource definition	64
5.4.8.3	Resource methods	64
5.4.8.3.1	POST	64
5.4.8.3.2	GET	65
5.4.8.3.3	PUT	65
5.4.8.3.4	PATCH	65
5.4.8.3.5	DELETE	65
5.4.9	Resource: Heal VNF task	66
5.4.9.1	Description	66
5.4.9.2	Resource definition	66
5.4.9.3	Resource methods	66
5.4.9.3.1	POST	66
5.4.9.3.2	GET	67
5.4.9.3.3	PUT	67
5.4.9.3.4	PATCH	68
5.4.9.3.5	DELETE	68
5.4.10	Resource: Operate VNF task	68
5.4.10.1	Description	68
5.4.10.2	Resource definition	68
5.4.10.3	Resource methods	69
5.4.10.3.1	POST	69
5.4.10.3.2	GET	70
5.4.10.3.3	PUT	71
5.4.10.3.4	PATCH	71
5.4.10.3.5	DELETE	71
5.4.11	Resource: Change external VNF connectivity task	71
5.4.11.1	Description	71
5.4.11.2	Resource definition	71
5.4.11.3	Resource methods	71
5.4.11.3.1	POST	71
5.4.11.3.2	GET	72
5.4.11.3.3	PUT	72
5.4.11.3.4	PATCH	72
5.4.11.3.5	DELETE	73
5.4.12	Resource: VNF LCM operation occurrences	73
5.4.12.1	Description	73
5.4.12.2	Resource definition	73
5.4.12.3	Resource methods	73
5.4.12.3.1	POST	73
5.4.12.3.2	GET	73
5.4.12.3.3	PUT	75
5.4.12.3.4	PATCH	75
5.4.12.3.5	DELETE	75
5.4.13	Resource: Individual VNF LCM operation occurrence	76
5.4.13.1	Description	76
5.4.13.2	Resource definition	76
5.4.13.3	Resource methods	76
5.4.13.3.1	POST	76
5.4.13.3.2	GET	76
5.4.13.3.3	PUT	77
5.4.13.3.4	PATCH	77
5.4.13.3.5	DELETE	77
5.4.14	Resource: Retry operation task	77
5.4.14.1	Description	77
5.4.14.2	Resource definition	77
5.4.14.3	Resource methods	78
5.4.14.3.1	POST	78

5.4.14.3.2	GET	79
5.4.14.3.3	PUT	79
5.4.14.3.4	PATCH.....	79
5.4.14.3.5	DELETE.....	79
5.4.15	Resource: Rollback operation task.....	79
5.4.15.1	Description	79
5.4.15.2	Resource definition	79
5.4.15.3	Resource methods	79
5.4.15.3.1	POST	79
5.4.15.3.2	GET	80
5.4.15.3.3	PUT	80
5.4.15.3.4	PATCH.....	81
5.4.15.3.5	DELETE.....	81
5.4.16	Resource: Fail operation task.....	81
5.4.16.1	Description	81
5.4.16.2	Resource definition	81
5.4.16.3	Resource methods	81
5.4.16.3.1	POST	81
5.4.16.3.2	GET	82
5.4.16.3.3	PUT	82
5.4.16.3.4	PATCH.....	82
5.4.16.3.5	DELETE.....	83
5.4.17	Resource: Cancel operation task.....	83
5.4.17.1	Description	83
5.4.17.2	Resource definition	83
5.4.17.3	Resource methods	83
5.4.17.3.1	POST	83
5.4.17.3.2	GET	84
5.4.17.3.3	PUT	84
5.4.17.3.4	PATCH.....	85
5.4.17.3.5	DELETE.....	85
5.4.18	Resource: Subscriptions.....	85
5.4.18.1	Description	85
5.4.18.2	Resource definition	85
5.4.18.3	Resource methods	85
5.4.18.3.1	POST	85
5.4.18.3.2	GET	86
5.4.18.3.3	PUT	87
5.4.18.3.4	PATCH.....	88
5.4.18.3.5	DELETE.....	88
5.4.19	Resource: Individual subscription.....	88
5.4.19.1	Description	88
5.4.19.2	Resource definition	88
5.4.19.3	Resource methods	88
5.4.19.3.1	POST	88
5.4.19.3.2	GET	88
5.4.19.3.3	PUT	89
5.4.19.3.4	PATCH.....	89
5.4.19.3.5	DELETE.....	89
5.4.20	Resource: Notification endpoint	90
5.4.20.1	Description	90
5.4.20.2	Resource definition	90
5.4.20.3	Resource methods	90
5.4.20.3.1	POST	90
5.4.20.3.2	GET	90
5.4.20.3.3	PUT	91
5.4.20.3.4	PATCH.....	91
5.4.20.3.5	DELETE.....	91
5.5	Data model	91
5.5.1	Introduction.....	91
5.5.2	Resource and notification data types	91
5.5.2.1	Introduction.....	91

5.5.2.2	Type: VnfInstance	91
5.5.2.3	Type: CreateVnfRequest	96
5.5.2.4	Type: InstantiateVnfRequest	97
5.5.2.5	Type: ScaleVnfRequest	97
5.5.2.6	Type: ScaleVnfToLevelRequest	98
5.5.2.7	Type: ChangeVnfFlavourRequest	98
5.5.2.8	Type: TerminateVnfRequest	99
5.5.2.9	Type: HealVnfRequest	99
5.5.2.10	Type: OperateVnfRequest	100
5.5.2.11	Type: ChangeExtVnfConnectivityRequest	100
5.5.2.12	Type: VnfInfoModificationRequest	101
5.5.2.12a	Type: VnfInfoModifications	102
5.5.2.13	Type: VnfLcmOpOcc	103
5.5.2.14	Type: CancelMode	104
5.5.2.15	Type: LccnSubscriptionRequest	104
5.5.2.16	Type: LccnSubscription	105
5.5.2.17	Type: VnfLcmOperationOccurrenceNotification	105
5.5.2.18	Type: VnfIdentifierCreationNotification	107
5.5.2.19	Type: VnfIdentifierDeletionNotification	107
5.5.3	Referenced structured data types	107
5.5.3.1	Introduction	107
5.5.3.2	Type: ExtVirtualLinkInfo	108
5.5.3.3	Type: ExtManagedVirtualLinkInfo	108
5.5.3.4	Type: ScaleInfo	108
5.5.3.5	Type: VnfcResourceInfo	108
5.5.3.6	Type: VnfVirtualLinkResourceInfo	109
5.5.3.7	Type: VirtualStorageResourceInfo	110
5.5.3.8	Type: VnfLinkPortInfo	110
5.5.3.9	Type: ExtLinkPortInfo	111
5.5.3.9a	Void	111
5.5.3.9b	Type: CpProtocolInfo	111
5.5.3.10	Type: IpOverEthernetAddressInfo	111
5.5.3.11	Type: MonitoringParameter	112
5.5.3.12	Type: LifecycleChangeNotificationsFilter	112
5.5.3.13	Type: AffectedVnfc	113
5.5.3.14	Type: AffectedVirtualLink	115
5.5.3.15	Type: AffectedVirtualStorage	115
5.5.3.16	Type: LccnLinks	116
5.5.3.17	Type: VnfExtCpInfo	116
5.5.4	Referenced simple data types and enumerations	117
5.5.4.1	Introduction	117
5.5.4.2	Simple data types	117
5.5.4.3	Enumeration: VnfOperationalStateType	117
5.5.4.4	Enumeration: StopType	117
5.5.4.5	Enumeration: LcmOperationStateType	118
5.5.4.6	Enumeration: CancelModeType	118
5.5.4.7	Enumeration: LcmOperationType	118
5.6	Success and error states of VNF lifecycle management operations	119
5.6.1	Basic concepts for error handling (informative)	119
5.6.1.1	Motivation	119
5.6.1.2	Failure resolution strategies: Retry and Rollback	119
5.6.1.3	Error handling at VNFM and NFVO	120
5.6.2	States and state transitions of a VNF lifecycle management operation occurrence	121
5.6.2.1	General	121
5.6.2.2	States of a VNF lifecycle management operation occurrence	121
5.6.2.3	Error handling operations that change the state of a VNF lifecycle management operation occurrence	124
5.6.3	Detailed flows for error handling	125
5.6.3.1	Immediate failure	125
5.6.3.2	Failure in "STARTING" state	125
5.6.3.3	Failure during actual LCM operation execution	126
5.6.3.4	LCM operation cancellation	128

6	VNF Performance Management interface.....	128
6.1	Description.....	128
6.1a	API version.....	128
6.2	Resource structure and methods.....	128
6.3	Sequence diagrams (informative).....	130
6.3.1	Flow of creating a PM job.....	130
6.3.1a	Flow of updating the callback URI of a PM job.....	131
6.3.2	Flow of querying/reading PM jobs.....	131
6.3.3	Flow of deleting a PM job.....	132
6.3.4	Flow of obtaining performance reports.....	133
6.3.5	Flow of creating a threshold.....	134
6.3.5a	Flow of updating the callback URI of a threshold.....	135
6.3.6	Flow of querying/reading thresholds.....	135
6.3.7	Flow of deleting thresholds.....	136
6.3.8	Void.....	137
6.3.9	Flow of sending notifications.....	137
6.4	Resources.....	137
6.4.1	Introduction.....	137
6.4.1a	Resource: API versions.....	137
6.4.2	Resource: PM jobs.....	138
6.4.2.1	Description.....	138
6.4.2.2	Resource definition.....	138
6.4.2.3	Resource methods.....	138
6.4.2.3.1	POST.....	138
6.4.2.3.2	GET.....	138
6.4.2.3.3	PUT.....	140
6.4.2.3.4	PATCH.....	140
6.4.2.3.5	DELETE.....	140
6.4.3	Resource: Individual PM job.....	141
6.4.3.1	Description.....	141
6.4.3.2	Resource definition.....	141
6.4.3.3	Resource methods.....	141
6.4.3.3.1	POST.....	141
6.4.3.3.2	GET.....	141
6.4.3.3.3	PUT.....	142
6.4.3.3.4	PATCH.....	142
6.4.3.3.5	DELETE.....	142
6.4.4	Resource: Individual performance report.....	143
6.4.4.1	Description.....	143
6.4.4.2	Resource definition.....	143
6.4.4.3	Resource methods.....	143
6.4.4.3.1	POST.....	143
6.4.4.3.2	GET.....	143
6.4.4.3.3	PUT.....	144
6.4.4.3.4	PATCH.....	144
6.4.4.3.5	DELETE.....	144
6.4.5	Resource: Thresholds.....	144
6.4.5.1	Description.....	144
6.4.5.2	Resource definition.....	144
6.4.5.3	Resource methods.....	145
6.4.5.3.1	POST.....	145
6.4.5.3.2	GET.....	145
6.4.5.3.3	PUT.....	146
6.4.5.3.4	PATCH.....	146
6.4.5.3.5	DELETE.....	146
6.4.6	Resource: Individual threshold.....	147
6.4.6.1	Description.....	147
6.4.6.2	Resource definition.....	147
6.4.6.3	Resource methods.....	147
6.4.6.3.1	POST.....	147
6.4.6.3.2	GET.....	147
6.4.6.3.3	PUT.....	148

6.4.6.3.4	PATCH	148
6.4.6.3.5	DELETE	148
6.4.7	Void	149
6.4.8	Void	149
6.4.9	Resource: Notification endpoint	149
6.4.9.1	Description	149
6.4.9.2	Resource definition	149
6.4.9.3	Resource methods	149
6.4.9.3.1	POST	149
6.4.9.3.2	GET	150
6.4.9.3.3	PUT	150
6.4.9.3.4	PATCH	150
6.4.9.3.5	DELETE	150
6.5	Data Model	151
6.5.1	Introduction	151
6.5.2	Resource and notification data types	151
6.5.2.1	Introduction	151
6.5.2.2	Void	151
6.5.2.3	Void	151
6.5.2.4	Type: ThresholdCrossedNotification	151
6.5.2.5	Type: PerformanceInformationAvailableNotification	152
6.5.2.6	Type: CreatePmJobRequest	153
6.5.2.7	Type: PmJob	154
6.5.2.8	Type: CreateThresholdRequest	154
6.5.2.9	Type: Threshold	155
6.5.2.10	Type: PerformanceReport	156
6.5.2.11	Type: ThresholdModifications	156
6.5.2.12	Type: PmJobModifications	157
6.5.3	Referenced structured data types	157
6.5.3.1	Introduction	157
6.5.3.2	Void	157
6.5.3.3	Type: PmJobCriteria	157
6.5.3.4	Type: ThresholdCriteria	158
6.5.4	Referenced simple data types and enumerations	158
6.5.4.1	Introduction	158
6.5.4.2	Simple data types	158
6.5.4.3	Enumeration: CrossingDirectionType	158
7	VNF Fault Management interface	159
7.1	Description	159
7.1a	API version	159
7.2	Resource structure and methods	159
7.3	Sequence diagrams (informative)	160
7.3.1	Flow of the Get Alarm List operation	160
7.3.2	Flow of acknowledging alarm	161
7.3.3	Flow of managing subscriptions	162
7.3.4	Flow of sending notifications	163
7.4	Resources	164
7.4.1	Introduction	164
7.4.2	Resource: Alarms	164
7.4.2.1	Description	164
7.4.2.2	Resource definition	164
7.4.2.3	Resource methods	164
7.4.2.3.1	POST	164
7.4.2.3.2	GET	164
7.4.2.3.3	PUT	165
7.4.2.3.4	PATCH	166
7.4.2.3.5	DELETE	166
7.4.3	Resource: Individual alarm	166
7.4.3.1	Description	166
7.4.3.2	Resource definition	166
7.4.3.3	Resource methods	166

7.4.3.3.1	POST	166
7.4.3.3.2	GET	166
7.4.3.3.3	PUT	167
7.4.3.3.4	PATCH	167
7.4.3.3.5	DELETE	168
7.4.4	Resource: Subscriptions	168
7.4.4.1	Description	168
7.4.4.2	Resource definition	169
7.4.4.3	Resource methods	169
7.4.4.3.1	POST	169
7.4.4.3.2	GET	170
7.4.4.3.3	PUT	171
7.4.4.3.4	PATCH	172
7.4.4.3.5	DELETE	172
7.4.5	Resource: Individual subscription	172
7.4.5.1	Description	172
7.4.5.2	Resource definition	172
7.4.5.3	Resource methods	172
7.4.5.3.1	POST	172
7.4.5.3.2	GET	172
7.4.5.3.3	PUT	173
7.4.5.3.4	PATCH	173
7.4.5.3.5	DELETE	173
7.4.6	Resource: Notification endpoint	174
7.4.6.1	Description	174
7.4.6.2	Resource definition	174
7.4.6.3	Resource methods	174
7.4.6.3.1	POST	174
7.4.6.3.2	GET	175
7.4.6.3.3	PUT	175
7.4.6.3.4	PATCH	175
7.4.6.3.5	DELETE	175
7.5	Data Model	175
7.5.1	Introduction	175
7.5.2	Resource and notification data types	175
7.5.2.1	Introduction	175
7.5.2.2	Type: FmSubscriptionRequest	176
7.5.2.3	Type: FmSubscription	176
7.5.2.4	Type: Alarm	176
7.5.2.5	Type: AlarmNotification	177
7.5.2.6	Type: AlarmClearedNotification	178
7.5.2.7	Type: AlarmListRebuiltNotification	178
7.5.2.8	Type: AlarmModifications	179
7.5.3	Referenced structured data types	179
7.5.3.1	Introduction	179
7.5.3.2	Type: FmNotificationsFilter	179
7.5.3.3	Type: FaultyResourceInfo	179
7.5.4	Referenced simple data types and enumerations	180
7.5.4.1	Introduction	180
7.5.4.2	Simple data types	180
7.5.4.3	Enumeration: PerceivedSeverityType	180
7.5.4.4	Enumeration: EventType	180
7.5.4.5	Enumeration: FaultyResourceType	181
8	VNF Indicator interface	181
8.1	Description	181
8.1a	API version	181
8.2	Resource structure and methods	182
8.3	Sequence diagrams (informative)	183
8.3.1	Flow of querying VNF indicators	183
8.3.2	Flow of reading a VNF indicator	183
8.3.3	Flow of managing subscriptions	184

8.3.4	Flow of sending notifications.....	185
8.4	Resources	186
8.4.1	Introduction.....	186
8.4.1a	Resource: API versions.....	186
8.4.2	Resource: VNF indicators.....	186
8.4.2.1	Description.....	186
8.4.2.2	Resource definition	186
8.4.2.3	Resource methods	186
8.4.2.3.1	POST	186
8.4.2.3.2	GET	186
8.4.2.3.3	PUT	187
8.4.2.3.4	PATCH.....	188
8.4.2.3.5	DELETE.....	188
8.4.3	Resource: VNF indicators related to a VNF instance	188
8.4.3.1	Description.....	188
8.4.3.2	Resource definition	188
8.4.3.3	Resource methods	188
8.4.3.3.1	POST	188
8.4.3.3.2	GET	188
8.4.3.3.3	PUT	189
8.4.3.3.4	PATCH.....	190
8.4.3.3.5	DELETE.....	190
8.4.4	Resource: Individual VNF indicator.....	190
8.4.4.1	Description.....	190
8.4.4.2	Resource definition	190
8.4.4.3	Resource methods	190
8.4.4.3.1	POST	190
8.4.4.3.2	GET	190
8.4.4.3.3	PUT	191
8.4.4.3.4	PATCH.....	191
8.4.4.3.5	DELETE.....	191
8.4.5	Resource: Subscriptions.....	191
8.4.5.1	Description.....	191
8.4.5.2	Resource definition	191
8.4.5.3	Resource methods	192
8.4.5.3.1	POST	192
8.4.5.3.2	GET	193
8.4.5.3.3	PUT	194
8.4.5.3.4	PATCH.....	195
8.4.5.3.5	DELETE.....	195
8.4.6	Resource: Individual subscription.....	195
8.4.6.1	Description.....	195
8.4.6.2	Resource definition	195
8.4.6.3	Resource methods	195
8.4.6.3.1	POST	195
8.4.6.3.2	GET	195
8.4.6.3.3	PUT	196
8.4.6.3.4	PATCH.....	196
8.4.6.3.5	DELETE.....	196
8.4.7	Resource: Notification endpoint	197
8.4.7.1	Description.....	197
8.4.7.2	Resource definition	197
8.4.7.3	Resource methods	197
8.4.7.3.1	POST	197
8.4.7.3.2	GET	197
8.4.7.3.3	PUT	198
8.4.7.3.4	PATCH.....	198
8.4.7.3.5	DELETE.....	198
8.5	Data model	198
8.5.1	Introduction.....	198
8.5.2	Resource and notification data types	198
8.5.2.1	Introduction.....	198

8.5.2.2	Type: VnfIndicator	198
8.5.2.3	Type: VnfIndicatorSubscriptionRequest	199
8.5.2.4	Type: VnfIndicatorSubscription	199
8.5.2.5	Type: VnfIndicatorValueChangeNotification	199
8.5.3	Referenced structured data types	200
8.5.3.1	Introduction	200
8.5.3.2	Type: VnfIndicatorNotificationsFilter	200
8.5.4	Referenced simple data types and enumerations	200
9	VNF Lifecycle Operation Granting interface	201
9.1	Description	201
9.1a	API version	202
9.2	Resource structure and methods	202
9.3	Sequence diagrams (informative)	202
9.3.1	Flow of grant request with synchronous response	202
9.3.2	Flow of grant request with asynchronous response	203
9.4	Resources	204
9.4.1	Introduction	204
9.4.1a	Resource: API versions	204
9.4.2	Resource: Grants	204
9.4.2.1	Description	204
9.4.2.2	Resource definition	204
9.4.2.3	Resource methods	205
9.4.2.3.1	POST	205
9.4.2.3.2	GET	205
9.4.2.3.3	PUT	206
9.4.2.3.4	PATCH	206
9.4.2.3.5	DELETE	206
9.4.3	Resource: Individual grant	206
9.4.3.1	Description	206
9.4.3.2	Resource definition	206
9.4.3.3	Resource methods	206
9.4.3.3.1	POST	206
9.4.3.3.2	GET	206
9.4.3.3.3	PUT	207
9.4.3.3.4	PATCH	207
9.4.3.3.5	DELETE	207
9.5	Data model	207
9.5.1	Introduction	207
9.5.2	Resource and notification data types	207
9.5.2.1	Introduction	207
9.5.2.2	Type: GrantRequest	208
9.5.2.3	Type: Grant	209
9.5.3	Referenced structured data types	211
9.5.3.1	Introduction	211
9.5.3.2	Type: ResourceDefinition	211
9.5.3.3	Type: GrantInfo	212
9.5.3.4	Type: ZoneInfo	213
9.5.3.5	Type: ZoneGroupInfo	213
9.5.3.6	Type: PlacementConstraint	214
9.5.3.7	Type: VimConstraint	214
9.5.3.8	Type: ConstraintResourceRef	215
9.5.3.9	Type: VimComputeResourceFlavour	216
9.5.3.10	Type: VimSoftwareImage	216
9.5.4	Referenced simple data types and enumerations	217
9.5.4.1	Introduction	217
9.5.4.2	Simple data types	217
9.5.4.3	Enumeration: GrantedLcmOperationType	217
10	VNF Package Management interface	217
10.1	Description	217
10.1a	API version	218

10.2	Resource structure and methods	218
10.3	Sequence diagrams (informative)	220
10.3.1	Flow of querying/reading VNF package information	220
10.3.2	Flow of reading the VNFD of an on-boarded VNF package	221
10.3.2a	Flow of fetching the VNF package manifest	221
10.3.3	Flow of fetching an on-boarded VNF package	222
10.3.4	Flow of fetching a VNF package artifact	222
10.3.4a	Flow of bulk-fetching VNF package artifacts that are not images	223
10.3.5	Flow of managing subscriptions	224
10.3.6	Flow of sending notifications	226
10.4	Resources	227
10.4.1	Introduction	227
10.4.1a	Resource: API versions	227
10.4.2	Resource: VNF packages	227
10.4.2.1	Description	227
10.4.2.2	Resource definition	227
10.4.2.3	Resource methods	227
10.4.2.3.1	POST	227
10.4.2.3.2	GET	227
10.4.2.3.3	PUT	229
10.4.2.3.4	PATCH	229
10.4.2.3.5	DELETE	229
10.4.3	Resource: Individual VNF package	230
10.4.3.1	Description	230
10.4.3.2	Resource definition	230
10.4.3.3	Resource methods	230
10.4.3.3.1	POST	230
10.4.3.3.2	GET	230
10.4.3.3.3	PUT	231
10.4.3.3.4	PATCH	231
10.4.3.3.5	DELETE	231
10.4.4	Resource: VNFD in an individual VNF package	231
10.4.4.1	Description	231
10.4.4.2	Resource definition	231
10.4.4.3	Resource methods	232
10.4.4.3.1	POST	232
10.4.4.3.2	GET	232
10.4.4.3.3	PUT	234
10.4.4.3.4	PATCH	234
10.4.4.3.5	DELETE	234
10.4.4a	Resource: Manifest in an individual VNF package	234
10.4.4a.1	Description	234
10.4.4a.2	Resource definition	235
10.4.4a.3	Resource methods	235
10.4.4a.3.1	POST	235
10.4.4a.3.2	GET	235
10.4.4a.3.3	PUT	236
10.4.4a.3.4	PATCH	236
10.4.4a.3.5	DELETE	236
10.4.5	Resource: VNF package content	237
10.4.5.1	Description	237
10.4.5.2	Resource definition	237
10.4.5.3	Resource methods	237
10.4.5.3.1	POST	237
10.4.5.3.2	GET	237
10.4.5.3.3	PUT	238
10.4.5.3.4	PATCH	239
10.4.5.3.5	DELETE	239
10.4.5a	Resource: VNF package artifacts	239
10.4.5a.1	Description	239
10.4.5a.2	Resource definition	239
10.4.5a.3	Resource methods	239