



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.

Reference

RGS/NFV-SOL003ed261

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 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.....	21
4.4.1.3 Void.....	21
4.4.1.3a Void.....	21
4.4.1.4 Void.....	21
4.4.1.5 Type: VnfInstanceSubscriptionFilter.....	21
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
4.4.2 Simple data types and enumerations	27
4.4.2.1 Introduction.....	27
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.....	36
5.3.5 Flow of the Query VNF operation	38
5.3.6 Flow of the Modify VNF Information operation	39
5.3.7 Flow of the Get Operation Status operation.....	41
5.3.8 Flow of managing subscriptions	42

5.3.9	Flow of sending notifications.....	43
5.3.10	Flow of retrying a VNF lifecycle management operation.....	44
5.3.11	Flow of rolling back a VNF lifecycle management operation.....	45
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	49
5.4.1	Introduction.....	49
5.4.1.1	Overview.....	49
5.4.1.2	Task resources that trigger VNF LCM operations	49
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.....	70
5.4.10.3.4	PATCH.....	70
5.4.10.3.5	DELETE.....	70
5.4.11	Resource: Change external VNF connectivity task.....	70
5.4.11.1	Description.....	70
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.....	72
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	87
5.4.18.3.5	DELETE	87
5.4.19	Resource: Individual subscription	87
5.4.19.1	Description	87
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	88
5.4.19.3.4	PATCH	88
5.4.19.3.5	DELETE	89
5.4.20	Resource: Notification endpoint	89
5.4.20.1	Description	89
5.4.20.2	Resource definition	89
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	95
5.5.2.4	Type: InstantiateVnfRequest	96
5.5.2.5	Type: ScaleVnfRequest	96
5.5.2.6	Type: ScaleVnfToLevelRequest	97
5.5.2.7	Type: ChangeVnfFlavourRequest	97
5.5.2.8	Type: TerminateVnfRequest	98
5.5.2.9	Type: HealVnfRequest	98
5.5.2.10	Type: OperateVnfRequest	99
5.5.2.11	Type: ChangeExtVnfConnectivityRequest	99
5.5.2.12	Type: VnfInfoModificationRequest	100
5.5.2.12a	Type: VnfInfoModifications	101
5.5.2.13	Type: VnfLcmOpOcc	101
5.5.2.14	Type: CancelMode	103
5.5.2.15	Type: LccnSubscriptionRequest	103
5.5.2.16	Type: LccnSubscription	104
5.5.2.17	Type: VnfLcmOperationOccurrenceNotification	104
5.5.2.18	Type: VnfIdentifierCreationNotification	106
5.5.2.19	Type: VnfIdentifierDeletionNotification	106
5.5.3	Referenced structured data types	106
5.5.3.1	Introduction	106
5.5.3.2	Type: ExtVirtualLinkInfo	107
5.5.3.3	Type: ExtManagedVirtualLinkInfo	107
5.5.3.4	Type: ScaleInfo	107
5.5.3.5	Type: VnfcResourceInfo	107
5.5.3.6	Type: VnfVirtualLinkResourceInfo	108
5.5.3.7	Type: VirtualStorageResourceInfo	108
5.5.3.8	Type: VnfLinkPortInfo	108
5.5.3.9	Type: ExtLinkPortInfo	109
5.5.3.9a	Type: ExtLinkPortData	109
5.5.3.9b	Type: CpProtocolInfo	110
5.5.3.10	Type: IpOverEthernetAddressInfo	110
5.5.3.11	Type: MonitoringParameter	111
5.5.3.12	Type: LifecycleChangeNotificationsFilter	111
5.5.3.13	Type: AffectedVnfc	112
5.5.3.14	Type: AffectedVirtualLink	113
5.5.3.15	Type: AffectedVirtualStorage	113
5.5.3.16	Type: LccnLinks	114
5.5.3.17	Type: VnfExtCpInfo	114
5.5.4	Referenced simple data types and enumerations	115
5.5.4.1	Introduction	115
5.5.4.2	Simple data types	115
5.5.4.3	Enumeration: VnfOperationalStateType	115
5.5.4.4	Enumeration: StopType	115
5.5.4.5	Enumeration: LcmOperationStateType	116
5.5.4.6	Enumeration: CancelModeType	116
5.5.4.7	Enumeration: LcmOperationType	116
5.6	Success and error states of VNF lifecycle management operations	117
5.6.1	Basic concepts for error handling (informative)	117
5.6.1.1	Motivation	117
5.6.1.2	Failure resolution strategies: Retry and Rollback	117
5.6.1.3	Error handling at VNF and NFVO	118
5.6.2	States and state transitions of a VNF lifecycle management operation occurrence	119
5.6.2.1	General	119
5.6.2.2	States of a VNF lifecycle management operation occurrence	119
5.6.2.3	Error handling operations that change the state of a VNF lifecycle management operation occurrence	122
5.6.3	Detailed flows for error handling	123
5.6.3.1	Immediate failure	123
5.6.3.2	Failure in "STARTING" state	123
5.6.3.3	Failure during actual LCM operation execution	124
5.6.3.4	LCM operation cancellation	126

6	VNF Performance Management interface.....	126
6.1	Description.....	126
6.1a	API version.....	126
6.2	Resource structure and methods.....	127
6.3	Sequence diagrams (informative).....	128
6.3.1	Flow of creating a PM job.....	128
6.3.2	Flow of querying/reading PM jobs.....	129
6.3.3	Flow of deleting a PM job.....	129
6.3.4	Flow of obtaining performance reports.....	130
6.3.5	Flow of creating a threshold.....	131
6.3.6	Flow of querying/reading thresholds.....	131
6.3.7	Flow of deleting thresholds.....	132
6.3.8	Flow of managing subscriptions.....	133
6.3.9	Flow of sending notifications.....	134
6.4	Resources.....	135
6.4.1	Introduction.....	135
6.4.1a	Resource: API versions.....	135
6.4.2	Resource: PM jobs.....	135
6.4.2.1	Description.....	135
6.4.2.2	Resource definition.....	135
6.4.2.3	Resource methods.....	135
6.4.2.3.1	POST.....	135
6.4.2.3.2	GET.....	136
6.4.2.3.3	PUT.....	137
6.4.2.3.4	PATCH.....	137
6.4.2.3.5	DELETE.....	137
6.4.3	Resource: Individual PM job.....	138
6.4.3.1	Description.....	138
6.4.3.2	Resource definition.....	138
6.4.3.3	Resource methods.....	138
6.4.3.3.1	POST.....	138
6.4.3.3.2	GET.....	138
6.4.3.3.3	PUT.....	139
6.4.3.3.4	PATCH.....	139
6.4.3.3.5	DELETE.....	139
6.4.4	Resource: Individual performance report.....	139
6.4.4.1	Description.....	139
6.4.4.2	Resource definition.....	139
6.4.4.3	Resource methods.....	140
6.4.4.3.1	POST.....	140
6.4.4.3.2	GET.....	140
6.4.4.3.3	PUT.....	140
6.4.4.3.4	PATCH.....	140
6.4.4.3.5	DELETE.....	140
6.4.5	Resource: Thresholds.....	141
6.4.5.1	Description.....	141
6.4.5.2	Resource definition.....	141
6.4.5.3	Resource methods.....	141
6.4.5.3.1	POST.....	141
6.4.5.3.2	GET.....	142
6.4.5.3.3	PUT.....	143
6.4.5.3.4	PATCH.....	143
6.4.5.3.5	DELETE.....	143
6.4.6	Resource: Individual threshold.....	143
6.4.6.1	Description.....	143
6.4.6.2	Resource definition.....	144
6.4.6.3	Resource methods.....	144
6.4.6.3.1	POST.....	144
6.4.6.3.2	GET.....	144
6.4.6.3.3	PUT.....	144
6.4.6.3.4	PATCH.....	145
6.4.6.3.5	DELETE.....	145

6.4.7	Resource: Subscriptions.....	145
6.4.7.1	Description.....	145
6.4.7.2	Resource definition.....	145
6.4.7.3	Resource methods.....	146
6.4.7.3.1	POST.....	146
6.4.7.3.2	GET.....	146
6.4.7.3.3	PUT.....	147
6.4.7.3.4	PATCH.....	148
6.4.7.3.5	DELETE.....	148
6.4.8	Resource: Individual subscription.....	148
6.4.8.1	Description.....	148
6.4.8.2	Resource definition.....	148
6.4.8.3	Resource methods.....	148
6.4.8.3.1	POST.....	148
6.4.8.3.2	GET.....	148
6.4.8.3.3	PUT.....	149
6.4.8.3.4	PATCH.....	149
6.4.8.3.5	DELETE.....	149
6.4.9	Resource: Notification endpoint.....	150
6.4.9.1	Description.....	150
6.4.9.2	Resource definition.....	150
6.4.9.3	Resource methods.....	150
6.4.9.3.1	POST.....	150
6.4.9.3.2	GET.....	150
6.4.9.3.3	PUT.....	151
6.4.9.3.4	PATCH.....	151
6.4.9.3.5	DELETE.....	151
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	Type: PmSubscriptionRequest.....	151
6.5.2.3	Type: PmSubscription.....	152
6.5.2.4	Type: ThresholdCrossedNotification.....	152
6.5.2.5	Type: PerformanceInformationAvailableNotification.....	153
6.5.2.6	Type: CreatePmJobRequest.....	154
6.5.2.7	Type: PmJob.....	155
6.5.2.8	Type: CreateThresholdRequest.....	155
6.5.2.9	Type: Threshold.....	155
6.5.2.10	Type: PerformanceReport.....	156
6.5.3	Referenced structured data types.....	156
6.5.3.1	Introduction.....	156
6.5.3.2	Type: PmNotificationsFilter.....	156
6.5.3.3	Type: PmJobCriteria.....	156
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	168
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	171
7.4.4.3.5	DELETE	171
7.4.5	Resource: Individual subscription	171
7.4.5.1	Description	171
7.4.5.2	Resource definition	171
7.4.5.3	Resource methods	171
7.4.5.3.1	POST	171
7.4.5.3.2	GET	171
7.4.5.3.3	PUT	172
7.4.5.3.4	PATCH	172
7.4.5.3.5	DELETE	172
7.4.6	Resource: Notification endpoint	173
7.4.6.1	Description	173
7.4.6.2	Resource definition	173
7.4.6.3	Resource methods	173
7.4.6.3.1	POST	173
7.4.6.3.2	GET	173
7.4.6.3.3	PUT	174
7.4.6.3.4	PATCH	174
7.4.6.3.5	DELETE	174
7.5	Data Model	174
7.5.1	Introduction	174
7.5.2	Resource and notification data types	174
7.5.2.1	Introduction	174
7.5.2.2	Type: FmSubscriptionRequest	174
7.5.2.3	Type: FmSubscription	175
7.5.2.4	Type: Alarm	175
7.5.2.5	Type: AlarmNotification	176
7.5.2.6	Type: AlarmClearedNotification	177
7.5.2.7	Type: AlarmListRebuiltNotification	177
7.5.2.8	Type: AlarmModifications	177
7.5.3	Referenced structured data types	178
7.5.3.1	Introduction	178
7.5.3.2	Type: FmNotificationsFilter	178
7.5.3.3	Type: FaultyResourceInfo	178
7.5.4	Referenced simple data types and enumerations	178
7.5.4.1	Introduction	178
7.5.4.2	Simple data types	179

7.5.4.3	Enumeration: PerceivedSeverityType	179
7.5.4.4	Enumeration: EventType	179
7.5.4.5	Enumeration: FaultyResourceType	179
8	VNF Indicator interface	180
8.1	Description	180
8.1a	API version	180
8.2	Resource structure and methods	180
8.3	Sequence diagrams (informative)	181
8.3.1	Flow of querying VNF indicators	181
8.3.2	Flow of reading a VNF indicator	182
8.3.3	Flow of managing subscriptions	183
8.3.4	Flow of sending notifications	184
8.4	Resources	185
8.4.1	Introduction	185
8.4.1a	Resource: API versions	185
8.4.2	Resource: VNF indicators	185
8.4.2.1	Description	185
8.4.2.2	Resource definition	185
8.4.2.3	Resource methods	185
8.4.2.3.1	POST	185
8.4.2.3.2	GET	185
8.4.2.3.3	PUT	186
8.4.2.3.4	PATCH	186
8.4.2.3.5	DELETE	186
8.4.3	Resource: VNF indicators related to a VNF instance	186
8.4.3.1	Description	186
8.4.3.2	Resource definition	187
8.4.3.3	Resource methods	187
8.4.3.3.1	POST	187
8.4.3.3.2	GET	187
8.4.3.3.3	PUT	188
8.4.3.3.4	PATCH	188
8.4.3.3.5	DELETE	188
8.4.4	Resource: Individual VNF indicator	188
8.4.4.1	Description	188
8.4.4.2	Resource definition	189
8.4.4.3	Resource methods	189
8.4.4.3.1	POST	189
8.4.4.3.2	GET	189
8.4.4.3.3	PUT	189
8.4.4.3.4	PATCH	190
8.4.4.3.5	DELETE	190
8.4.5	Resource: Subscriptions	190
8.4.5.1	Description	190
8.4.5.2	Resource definition	190
8.4.5.3	Resource methods	190
8.4.5.3.1	POST	190
8.4.5.3.2	GET	191
8.4.5.3.3	PUT	192
8.4.5.3.4	PATCH	192
8.4.5.3.5	DELETE	192
8.4.6	Resource: Individual subscription	193
8.4.6.1	Description	193
8.4.6.2	Resource definition	193
8.4.6.3	Resource methods	193
8.4.6.3.1	POST	193
8.4.6.3.2	GET	193
8.4.6.3.3	PUT	194
8.4.6.3.4	PATCH	194
8.4.6.3.5	DELETE	194
8.4.7	Resource: Notification endpoint	194

8.4.7.1	Description	194
8.4.7.2	Resource definition	194
8.4.7.3	Resource methods	195
8.4.7.3.1	POST	195
8.4.7.3.2	GET	195
8.4.7.3.3	PUT	196
8.4.7.3.4	PATCH	196
8.4.7.3.5	DELETE	196
8.5	Data model	196
8.5.1	Introduction	196
8.5.2	Resource and notification data types	196
8.5.2.1	Introduction	196
8.5.2.2	Type: VnfIndicator	196
8.5.2.3	Type: VnfIndicatorSubscriptionRequest	197
8.5.2.4	Type: VnfIndicatorSubscription	197
8.5.2.5	Type: VnfIndicatorValueChangeNotification	197
8.5.3	Referenced structured data types	198
8.5.3.1	Introduction	198
8.5.3.2	Type: VnfIndicatorNotificationsFilter	198
8.5.4	Referenced simple data types and enumerations	198
9	VNF Lifecycle Operation Granting interface	199
9.1	Description	199
9.1a	API version	200
9.2	Resource structure and methods	200
9.3	Sequence diagrams (informative)	200
9.3.1	Flow of grant request with synchronous response	200
9.3.2	Flow of grant request with asynchronous response	201
9.4	Resources	202
9.4.1	Introduction	202
9.4.1a	Resource: API versions	202
9.4.2	Resource: Grants	202
9.4.2.1	Description	202
9.4.2.2	Resource definition	202
9.4.2.3	Resource methods	203
9.4.2.3.1	POST	203
9.4.2.3.2	GET	203
9.4.2.3.3	PUT	204
9.4.2.3.4	PATCH	204
9.4.2.3.5	DELETE	204
9.4.3	Resource: Individual grant	204
9.4.3.1	Description	204
9.4.3.2	Resource definition	204
9.4.3.3	Resource methods	204
9.4.3.3.1	POST	204
9.4.3.3.2	GET	204
9.4.3.3.3	PUT	205
9.4.3.3.4	PATCH	205
9.4.3.3.5	DELETE	205
9.5	Data model	205
9.5.1	Introduction	205
9.5.2	Resource and notification data types	205
9.5.2.1	Introduction	205
9.5.2.2	Type: GrantRequest	206
9.5.2.3	Type: Grant	207
9.5.3	Referenced structured data types	209
9.5.3.1	Introduction	209
9.5.3.2	Type: ResourceDefinition	210
9.5.3.3	Type: GrantInfo	210
9.5.3.4	Type: ZoneInfo	211
9.5.3.5	Type: ZoneGroupInfo	212
9.5.3.6	Type: PlacementConstraint	212

9.5.3.7	Type: VimConstraint.....	213
9.5.3.8	Type: ConstraintResourceRef	213
9.5.3.9	Type: VimComputeResourceFlavour.....	214
9.5.3.10	Type: VimSoftwareImage	215
9.5.4	Referenced simple data types and enumerations	216
9.5.4.1	Introduction.....	216
9.5.4.2	Simple data types	216
9.5.4.3	Enumeration: GrantedLcmOperationType.....	216
10	VNF Package Management interface.....	216
10.1	Description	216
10.1a	API version.....	216
10.2	Resource structure and methods.....	217
10.3	Sequence diagrams (informative).....	219
10.3.1	Flow of querying/reading VNF package information	219
10.3.2	Flow of reading the VNFD of an on-boarded VNF package	220
10.3.3	Flow of fetching an on-boarded VNF package	220
10.3.4	Flow of fetching a VNF package artifact.....	221
10.3.5	Flow of managing subscriptions	222
10.3.6	Flow of sending notifications.....	223
10.4	Resources	224
10.4.1	Introduction.....	224
10.4.1a	Resource: API versions.....	224
10.4.2	Resource: VNF packages.....	224
10.4.2.1	Description	224
10.4.2.2	Resource definition	224
10.4.2.3	Resource methods	224
10.4.2.3.1	POST	224
10.4.2.3.2	GET	224
10.4.2.3.3	PUT	226
10.4.2.3.4	PATCH.....	226
10.4.2.3.5	DELETE.....	226
10.4.3	Resource: Individual VNF package	226
10.4.3.1	Description	226
10.4.3.2	Resource definition	226
10.4.3.3	Resource methods	227
10.4.3.3.1	POST	227
10.4.3.3.2	GET	227
10.4.3.3.3	PUT	227
10.4.3.3.4	PATCH.....	227
10.4.3.3.5	DELETE.....	227
10.4.4	Resource: VNFD in an individual VNF package.....	227
10.4.4.1	Description	227
10.4.4.2	Resource definition	228
10.4.4.3	Resource methods	228
10.4.4.3.1	POST	228
10.4.4.3.2	GET	228
10.4.4.3.3	PUT	229
10.4.4.3.4	PATCH.....	229
10.4.4.3.5	DELETE.....	229
10.4.5	Resource: VNF package content.....	229
10.4.5.1	Description	229
10.4.5.2	Resource definition	230
10.4.5.3	Resource methods	230
10.4.5.3.1	POST	230
10.4.5.3.2	GET	230
10.4.5.3.3	PUT	231
10.4.5.3.4	PATCH.....	231
10.4.5.3.5	DELETE.....	231
10.4.6	Resource: Individual VNF package artifact.....	231
10.4.6.1	Description	231
10.4.6.2	Resource definition	232