ETSI GS NFV-IFA 013 V4.2.1 (2021-05)

Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Os-Ma-nfvo reference point - Interface and Information Model Specification

Disclaimer

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

Reference
RGS/NFV-IFA013ed421

Keywords
interface, management, MANO, NFV,
orchestration, virtualisation

ETSI

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

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

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights 19
Foreword 19
Modal verbs terminology 19
1 Scope 20
2 References 20
2.1 Normative references 20
2.2 Informative references 20
3 Definition of terms, symbols and abbreviations 21
3.1 Terms 21
3.2 Symbols 21
3.3 Abbreviations 21
4 Overview of interfaces and information elements associated to the Os-Ma-nfvo reference point 21
4.1 Introduction 21
4.2 Relation to other NFV group specifications 22
4.3 Conventions 22
5 Reference point and interface requirements 23
5.1 Introduction 23
5.2 Os-Ma-nfvo reference point requirements 23
5.3 Interface requirements 23
5.3.1 NSD Management interface requirements 23
5.3.2 NS Lifecycle Management interface requirements 24
5.3.3 Void 26
5.3.4 NS Performance Management interface requirements 26
5.3.5 NS Fault Management interface requirements 28
5.3.6 VNF Package Management interface requirements 28
5.3.7 NFVI Capacity Information interface 29
5.3.8 Policy Management interface requirements 29
5.3.9 VNF Snapshot Package Management interface requirements 30
5.3.10 LCM Coordination interface requirements 30
6 OSS exposed interfaces 31
6.1 LCM Coordination interface 31
6.1.1 Description 31
6.1.2 CoordinateLcmOperation operation 31
6.1.2.1 Description 31
6.1.2.2 Input parameters 31
6.1.2.3 Output parameters 32
6.1.2.4 Operation results 32
7 NFVO exposed interfaces 32
7.1 Introduction 32
7.2 NSD Management interface 33
7.2.1 Description 33
7.2.2 Upload NSD operation 34
7.2.2.1 Description 34
7.2.2.2 Input parameters 34
7.2.2.3 Output parameters 34
7.2.2.4 Operation results 34
7.2.3 Void 34
7.2.4 Void 34
7.2.5 Update NSD Info operation 35
7.2.5.1 Description 35
7.2.5.2 Input parameters 35
7.2.5.3 Output parameters 35

7.2.5.4 Operation results 35
7.2.6 Delete NSD operation 35
7.2.6.1 Description 35
7.2.6.2 Input parameters 36
7.2.6.3 Output parameters 36
7.2.6.4 Operation results 36
7.2.7 Query NSD Info operation 36
7.2.7.1 Description 36
7.2.7.2 Input parameters 36
7.2.7.3 Output parameters 37
7.2.7.4 Operation results 37
7.2.8 Upload PNFD operation 37
7.2.8.1 Description 37
7.2.8.2 Input parameters 37
7.2.8.3 Output parameters 37
7.2.8.4 Operation results 37
7.2.9 Update PNFD Info operation 38
7.2.9.1 Description 38
7.2.9.2 Input parameters 38
7.2.9.3 Output parameters 38
7.2.9.4 Operation results 38
7.2.10 Delete PNFD operation 38
7.2.10.1 Description 38
7.2.10.2 Input parameters 39
7.2.10.3 Output parameters 39
7.2.10.4 Operation results 39
7.2.11 Query PNFD Info operation 39
7.2.11.1 Description 39
7.2.11.2 Input parameters 39
7.2.11.3 Output parameters 40
7.2.11.4 Operation results 40
7.2.12 Subscribe operation 40
7.2.12.1 Description 40
7.2.12.2 Input parameters 40
7.2.12.3 Output parameters 41
7.2.12.4 Operation results 41
7.2.13 Notify operation 41
7.2.13.1 Description 41
7.2.14 Terminate Subscription operation 41
7.2.14.1 Description 41
7.2.14.2 Input parameters 42
7.2.14.3 Output parameters 42
7.2.14.4 Operation results 42
7.2.15 Query Subscription Info operation 42
7.2.15.1 Description 42
7.2.15.2 Input parameters 42
7.2.15.3 Output parameters 42
7.2.15.4 Operation results 43
7.2.16 Create NSD Info operation 43
7.2.16.1 Description 43
7.2.16.2 Input parameters 43
7.2.16.3 Output parameters 43
7.2.16.4 Operation results 43
7.2.17 Fetch NSD operation 43
7.2.17.1 Description 43
7.2.17.2 Input parameters 44
7.2.17.3 Output parameters 44
7.2.17.4 Operation results 44
7.2.18 Create PNFD Info operation 44
7.2.18.1 Description 44
7.2.18.2 Input parameters 44
7.2.18.3 Output parameters 45

7.2.18.4 Operation results 45
7.2.19 Fetch PNFD operation 45
7.2.19.1 Description 45
7.2.19.2 Input parameters 45
7.2.19.3 Output parameters 45
7.2.19.4 Operation results 45
7.2.20 Fetch NSD Archive Artifacts operation 46
7.2.20.1 Description 46
7.2.20.2 Input parameters 46
7.2.20.3 Output parameters 46
7.2.20.4 Operation results 46
7.2.21 Fetch PNFD Archive Artifacts operation 46
7.2.21.1 Description 46
7.2.21.2 Input parameters 47
7.2.21.3 Output parameters 47
7.2.21.4 Operation results 47
7.3 NS Lifecycle Management interface 47
7.3.1 Description 47
7.3.2 Create NS Identifier operation 48
7.3.2.1 Description 48
7.3.2.2 Input parameters 48
7.3.2.3 Output parameters 48
7.3.2.4 Operation results 48
7.3.3 Instantiate NS operation 49
7.3.3.1 Description 49
7.3.3.2 Input parameters 49
7.3.3.3 Output parameters 52
7.3.3.4 Operation results 52
7.3.4 Scale NS operation 52
7.3.4.1 Description 52
7.3.4.2 Input parameters 53
7.3.4.3 Output parameters 53
7.3.4.4 Operation results 53
7.3.5 Update NS operation 54
7.3.5.1 Introduction 54
7.3.5.2 Input parameters 55
7.3.5.3 Output parameters 58
7.3.5.4 Operation results 59
7.3.6 Query NS operation 60
7.3.6.1 Description 60
7.3.6.2 Input parameters 60
7.3.6.3 Output parameters 60
7.3.6.4 Operation results 61
7.3.7 Terminate NS operation 61
7.3.7.1 Description 61
7.3.7.2 Input parameters 61
7.3.7.3 Output parameters 61
7.3.7.4 Operation results 62
7.3.8 Delete NS Identifier operation 62
7.3.8.1 Description 62
7.3.8.2 Input parameters 62
7.3.8.3 Output parameters 62
7.3.8.4 Operation results 62
7.3.9 Heal NS operation 63
7.3.9.1 Description 63
7.3.9.2 Input parameters 63
7.3.9.3 Output parameters 63
7.3.9.4 Operation results 63
7.3.10 Get Operation Status operation 64
7.3.10.1 Description 64
7.3.10.2 Input parameters 64
7.3.10.3 Output Parameters 64

7.3.10.4 Operation results 64
7.3.11 Subscribe operation 64
7.3.11.1 Description 64
7.3.11.2 Input parameters 65
7.3.11.3 Output parameters 65
7.3.11.4 Operation results 65
7.3.12 Notify operation 65
7.3.12.1 Description 65
7.3.13 Terminate Subscription operation 66
7.3.13.1 Description 66
7.3.13.2 Input parameters 66
7.3.13.3 Output parameters 66
7.3.13.4 Operation results 66
7.3.14 Query Subscription Info operation 66
7.3.14.1 Description 66
7.3.14.2 Input parameters 67
7.3.14.3 Output parameters 67
7.3.14.4 Operation results 67
7.4 Void 67
7.5 NS Performance Management interface 67
7.5.1 Description 67
7.5.2 Create PM Job operation 68
7.5.2.1 Description 68
7.5.2.2 Input parameters 68
7.5.2.3 Output parameters 69
7.5.2.4 Operation results 69
7.5.3 Delete PM Jobs operation 69
7.5.3.1 Description 69
7.5.3.2 Input parameters 69
7.5.3.3 Output parameters 69
7.5.3.4 Operation results 70
7.5.4 Subscribe operation 70
7.5.4.1 Description 70
7.5.4.2 Input parameters 70
7.5.4.3 Output parameters 70
7.5.4.4 Operation results 70
7.5.5 Notify operation 71
7.5.5.1 Description 71
7.5.6 Query PM Job operation 71
7.5.6.1 Description 71
7.5.6.2 Input parameters 71
7.5.6.3 Output parameters 71
7.5.6.4 Operation results 72
7.5.7 Create Threshold operation 72
7.5.7.1 Description 72
7.5.7.2 Input parameters 72
7.5.7.3 Output parameters 72
7.5.7.4 Operation results 72
7.5.8 Delete Thresholds operation 73
7.5.8.1 Description 73
7.5.8.2 Input parameters 73
7.5.8.3 Output parameters 73
7.5.8.4 Operation results 73
7.5.9 Query Threshold operation 73
7.5.9.1 Description 73
7.5.9.2 Input parameters 74
7.5.9.3 Output parameters 74
7.5.9.4 Operation results 74
7.6 NS Fault Management interface 74
7.6.1 Description 74
7.6.2 Subscribe operation 74
7.6.2.1 Description 74

7.6.2.2 Input parameters 75
7.6.2.3 Output parameters 75
7.6.2.4 Operation results 75
7.6.3 Notify operation 75
7.6.3.1 Description 75
7.6.4 Get Alarm List operation 76
7.6.4.1 Description 76
7.6.4.2 Input parameters 76
7.6.4.3 Output parameters 76
7.6.4.4 Operation results 76
7.6.5 Terminate Subscription operation 76
7.6.5.1 Description 76
7.6.5.2 Input parameters 77
7.6.5.3 Output parameters 77
7.6.5.4 Operation results 77
7.6.6 Query Subscription Info operation 77
7.6.6.1 Description 77
7.6.6.2 Input parameters 77
7.6.6.3 Output parameters 77
7.6.6.4 Operation results 78
7.6.7 Acknowledge Alarms operation 78
7.6.7.1 Description 78
7.6.7.2 Input parameters 78
7.6.7.3 Output parameters 78
7.6.7.4 Operation results 78
7.7 VNF Package management interface 79
7.7.1 Description 79
7.7.2 Upload VNF Package operation 79
7.7.2.1 Description 79
7.7.2.2 Input parameters 80
7.7.2.3 Output parameters 80
7.7.2.4 Operation results 80
7.7.3 Void 80
7.7.4 Void 80
7.7.5 Delete VNF Package operation 80
7.7.5.1 Description 80
7.7.5.2 Input parameters 81
7.7.5.3 Output parameters 81
7.7.5.4 Operation results 81
7.7.6 Query VNF Package Info operation 81
7.7.6.1 Description 81
7.7.6.2 Input parameters 81
7.7.6.3 Output parameters 82
7.7.6.4 Operation results 82
7.7.7 Subscribe operation 82
7.7.7.1 Description 82
7.7.7.2 Input parameters 82
7.7.7.3 Output parameters 83
7.7.7.4 Operation results 83
7.7.8 Notify operation 83
7.7.8.1 Description 83
7.7.9 Void 83
7.7.10 Fetch VNF Package operation 83
7.7.10.1 Description 83
7.7.10.2 Input parameters 84
7.7.10.3 Output parameters 84
7.7.10.4 Operation results 84
7.7.11 Fetch VNF Package Artifacts operation 84
7.7.11.1 Description 84
7.7.11.2 Input parameters 84
7.7.11.3 Output parameters 85
7.7.11.4 Operation results 85

7.7.12 Void 85
7.7.13 Terminate Subscription operation 85
7.7.13.1 Description 85
7.7.13.2 Input parameters 85
7.7.13.3 Output parameters 85
7.7.13.4 Operation results 85
7.7.14 Query Subscription Info operation 86
7.7.14.1 Description 86
7.7.14.2 Input parameters 86
7.7.14.3 Output parameters 86
7.7.14.4 Operation results 86
7.7.15 Create VNF Package Info operation 86
7.7.15.1 Description 86
7.7.15.2 Input parameters 87
7.7.15.3 Output parameters 87
7.7.15.4 Operation results 87
7.7.16 Update VNF Package Info operation 87
7.7.16.1 Description 87
7.7.16.2 Input parameters 87
7.7.16.3 Output parameters 88
7.7.16.4 Operation results 88
7.8 NFVI Capacity Information interface 88
7.8.1 Description 88
7.8.2 Query NFVI capacity operation 88
7.8.2.1 Description 88
7.8.2.2 Input parameters 89
7.8.2.3 Output parameters 89
7.8.2.4 Operation results 89
7.8.3 Subscribe operation 89
7.8.3.1 Description 89
7.8.3.2 Input parameters 89
7.8.3.3 Output parameters 90
7.8.3.4 Operation results 90
7.8.4 Terminate Subscription operation 90
7.8.4.1 Description 90
7.8.4.2 Input parameters 90
7.8.4.3 Output parameters 91
7.8.4.4 Operation results 91
7.8.5 Query Subscription Info operation 91
7.8.5.1 Description 91
7.8.5.2 Input parameters 91
7.8.5.3 Output parameters 91
7.8.5.4 Operation results 91
7.8.6 Notify operation 92
7.8.6.1 Description 92
7.8.7 Create Capacity Threshold operation 92
7.8.7.1 Description 92
7.8.7.2 Input parameters 92
7.8.7.3 Output parameters 93
7.8.7.4 Operation results 93
7.8.8 Delete Capacity Thresholds operation 93
7.8.8.1 Description 93
7.8.8.2 Input parameters 93
7.8.8.3 Output parameters 93
7.8.8.4 Operation results 93
7.8.9 Query Capacity Threshold operation 94
7.8.9.1 Description 94
7.8.9.2 Input parameters 94
7.8.9.3 Output parameters 94
7.8.9.4 Operation results 94
7.9 Policy Management interface 94
7.9.1 Description 94

7.9.2 Transfer Policy operation....95
7.9.2.1 Description....95
7.9.2.2 Input parameters....95
7.9.2.3 Output parameters....95
7.9.2.4 Operation results....95
7.9.3 Delete Policy operation....96
7.9.3.1 Description....96
7.9.3.2 Input parameters....96
7.9.3.3 Output parameters....96
7.9.3.4 Operation results....96
7.9.4 Query Policy operation....96
7.9.4.1 Description....96
7.9.4.2 Input parameters....97
7.9.4.3 Output parameters....97
7.9.4.4 Operation results....97
7.9.5 Activate Policy operation....97
7.9.5.1 Description....97
7.9.5.2 Input parameters....97
7.9.5.3 Output parameters....98
7.9.5.4 Operation results....98
7.9.6 Deactivate Policy operation....98
7.9.6.1 Description....98
7.9.6.2 Input parameters....98
7.9.6.3 Output parameters....98
7.9.6.4 Operation results....99
7.9.7 Subscribe operation....99
7.9.7.1 Description....99
7.9.7.2 Input parameters....99
7.9.7.3 Output parameters....99
7.9.7.4 Operation results....99
7.9.8 Notify operation....100
7.9.8.1 Description....100
7.9.9 Terminate Subscription operation....100
7.9.9.1 Description....100
7.9.9.2 Input parameters....100
7.9.9.3 Output parameters....100
7.9.9.4 Operation results....100
7.9.10 Query Subscription Info operation....101
7.9.10.1 Description....101
7.9.10.2 Input parameters....101
7.9.10.3 Output parameters....101
7.9.10.4 Operation results....101
7.9.11 Associate Policy operation....101
7.9.11.1 Description....101
7.9.11.2 Input parameters....102
7.9.11.3 Output parameters....102
7.9.11.4 Operation results....102
7.9.12 Disassociate Policy operation....102
7.9.12.1 Description....102
7.9.12.2 Input parameters....102
7.9.12.3 Output parameters....103
7.9.12.4 Operation results....103
7.10 VNF Snapshot Package Management interface....103
7.10.1 Description....103
7.10.2 Create VNF Snapshot Package Info operation....103
7.10.2.1 Description....103
7.10.2.2 Input parameters....103
7.10.2.3 Output parameters....103
7.10.2.4 Operation results....104
7.10.3 Build VNF Snapshot Package operation....104
7.10.3.1 Description....104
7.10.3.2 Input parameters....104

7.10.3.3 Output parameters 104
7.10.3.4 Operation results 104
7.10.4 Upload VNF Snapshot Package operation 105
7.10.4.1 Description 105
7.10.4.2 Input parameters 105
7.10.4.3 Output parameters 105
7.10.4.4 Operation results 106
7.10.5 Extract VNF Snapshot Package operation 106
7.10.5.1 Description 106
7.10.5.2 Input parameters 106
7.10.5.3 Output parameters 106
7.10.5.4 Operation results 107
7.10.6 Fetch VNF Snapshot Package operation 107
7.10.6.1 Description 107
7.10.6.2 Input parameters 107
7.10.6.3 Output parameters 107
7.10.6.4 Operation results 107
7.10.7 Fetch VNF Snapshot Package Artifacts operation 107
7.10.7.1 Description 107
7.10.7.2 Input parameters 108
7.10.7.3 Output parameters 108
7.10.7.4 Operation results 108
7.10.8 Query VNF Snapshot Package Information operation 108
7.10.8.1 Description 108
7.10.8.2 Input parameters 109
7.10.8.3 Output parameters 109
7.10.8.4 Operation results 109
7.10.9 Delete VNF Snapshot Package operation 109
7.10.9.1 Description 109
7.10.9.2 Input parameters 110
7.10.9.3 Output parameters 110
7.10.9.4 Operation results 110
7.10.10 Update VNF Snapshot Package operation 110
7.10.10.1 Description 110
7.10.10.2 Input parameters 110
7.10.10.3 Output parameters 111
7.10.10.4 Operation results 111

8 Information elements exchanged 111
8.1 Introduction 111
8.2 Information elements related to NSD Management 112
8.2.1 Introduction 112
8.2.2 NsdInfo information element 112
8.2.2.1 Description 112
8.2.2.2 Attributes 112
8.2.3 Pnfd information element 113
8.2.3.1 Description 113
8.2.3.2 Attributes 113
8.2.4 PnfdInfo information element 113
8.2.4.1 Description 113
8.2.4.2 Attributes 113
8.2.5 Nsd information element 114
8.2.5.1 Description 114
8.2.5.2 Attributes 114
8.2.6 NsdOnBoardingNotification 114
8.2.6.1 Description 114
8.2.6.2 Trigger Conditions 114
8.2.6.3 Attributes 114
8.2.7 NsdChangeNotification 114
8.2.7.1 Description 114
8.2.7.2 Trigger Conditions 114
8.2.7.3 Attributes 115

8.2.8 NsdDeletionNotification 115
8.2.8.1 Description 115
8.2.8.2 Trigger Conditions 115
8.2.8.3 Attributes 115
8.2.9 PnfdOnBoardingNotification 115
8.2.9.1 Description 115
8.2.9.2 Trigger Conditions 115
8.2.9.3 Attributes 116
8.2.10 PnfdDeletionNotification 116
8.2.10.1 Description 116
8.2.10.2 Trigger Conditions 116
8.2.10.3 Attributes 116
8.2.11 NsdArchiveArtifactInformation information element 116
8.2.11.1 Description 116
8.2.11.2 Attributes 116
8.2.12 PnfdArchiveArtifactInformation information element 117
8.2.12.1 Description 117
8.2.12.2 Attributes 117
8.3 Information elements and notifications related to NS Lifecycle Management 117
8.3.1 Introduction 117
8.3.2 Information elements and notifications related to NS Lifecycle Changes 117
8.3.2.1 Introduction 117
8.3.2.2 NsLcmOperationOccurrenceNotification 117
8.3.2.2.1 Description 117
8.3.2.2.2 Trigger conditions 117
8.3.2.2.3 Attributes 118
8.3.2.3 AffectedVnf information element 119
8.3.2.3.1 Description 119
8.3.2.3.2 Attributes 119
8.3.2.4 AffectedPnf information element 120
8.3.2.4.1 Description 120
8.3.2.4.2 Attributes 120
8.3.2.5 AffectedVirtualLink information element 121
8.3.2.5.1 Description 121
8.3.2.5.2 Attributes 121
8.3.2.6 AffectedVnffg information element 121
8.3.2.6.1 Description 121
8.3.2.6.2 Attributes 121
8.3.2.7 AffectedNs information element 122
8.3.2.7.1 Description 122
8.3.2.7.2 Attributes 122
8.3.2.8 AffectedSap information element 122
8.3.2.8.1 Description 122
8.3.2.8.2 Attributes 122
8.3.2.9 NsIdentifierCreationNotification 123
8.3.2.9.1 Description 123
8.3.2.9.2 Trigger conditions 123
8.3.2.9.3 Attributes 123
8.3.2.10 NsIdentifierDeletionNotification 123
8.3.2.10.1 Description 123
8.3.2.10.2 Trigger conditions 123
8.3.2.10.3 Attributes 123
8.3.2.11 NsChangeNotification 124
8.3.2.11.1 Description 124
8.3.2.11.2 Trigger conditions 124
8.3.2.11.3 Attributes 124
8.3.3 Information elements related to NsInfo 125
8.3.3.1 Introduction 125
8.3.3.2 NsInfo information element 125
8.3.3.2.1 Description 125
8.3.3.2.2 Attributes 125
8.3.3.3 VnfInfo information element 126

8.3.3.3.1 Description 126
8.3.3.3.2 Attributes 126
8.3.3.4 InstantiatedVnfInfo information element 128
8.3.3.4.1 Description 128
8.3.3.4.2 Attributes 128
8.3.3.5 VnfcResourceInfo information element 130
8.3.3.5.1 Description 130
8.3.3.5.2 Attributes 130
8.3.3.6 VnfVirtualLinkResourceInfo information element 130
8.3.3.6.1 Description 130
8.3.3.6.2 Attributes 130
8.3.3.7 VirtualStorageResourceInfo information element 131
8.3.3.7.1 Description 131
8.3.3.7.2 Attributes 131
8.3.3.8 ResourceHandle information element 132
8.3.3.8.1 Description 132
8.3.3.8.2 Attributes 132
8.3.3.9 PnfInfo information element 132
8.3.3.9.1 Description 132
8.3.3.9.2 Attributes 132
8.3.3.10 NsVirtualLinkInfo information element 133
8.3.3.10.1 Description 133
8.3.3.10.2 Attributes 133
8.3.3.11 NsLinkPortInfo information element 133
8.3.3.11.1 Description 133
8.3.3.11.2 Attributes 134
8.3.3.12 SapInfo information element 134
8.3.3.12.1 Description 134
8.3.3.12.2 Attributes 134
8.3.3.13 VnffgInfo information element 134
8.3.3.13.1 Description 134
8.3.3.13.2 Attributes 134
8.3.3.14 PnfExtCpInfo information element 135
8.3.3.14.1 Description 135
8.3.3.14.2 Attributes 135
8.3.3.15 NfpInfo information element 135
8.3.3.15.1 Description 135
8.3.3.15.2 Attributes 135
8.3.3.16 NsScaleInfo information element 136
8.3.3.16.1 Description 136
8.3.3.16.2 Attributes 136
8.3.3.17 VnfExtCpInfo information element 136
8.3.3.17.1 Description 136
8.3.3.17.2 Attributes 136
8.3.3.18 ExtVirtualLinkInfo information element 137
8.3.3.18.1 Description 137
8.3.3.18.2 Attributes 137
8.3.3.19 ExtManagedVirtualLinkInfo information element 138
8.3.3.19.1 Description 138
8.3.3.19.2 Attributes 138
8.3.3.20 VnfLinkPortInfo information element 139
8.3.3.20.1 Description 139
8.3.3.20.2 Attributes 139
8.3.3.21 ScaleInfo information element 139
8.3.3.21.1 Description 139
8.3.3.21.2 Attributes 139
8.3.3.22 ExtLinkPortInfo information element 140
8.3.3.22.1 Description 140
8.3.3.22.2 Attributes 140
8.3.3.23 VnfcCpInfo information element 140
8.3.3.23.1 Description 140
8.3.3.23.2 Attributes 140

8.3.3.24 CpProtocolInfo information element 141
8.3.3.24.1 Description 141
8.3.3.24.2 Attributes 141
8.3.3.25 CpGroupInfo information element 141
8.3.3.25.1 Description 141
8.3.3.25.2 Attributes 141
8.3.3.26 CpPairInfo information element 142
8.3.3.26.1 Description 142
8.3.3.26.2 Attributes 142
8.3.3.27 VnfSnapshotInfo information element 142
8.3.3.27.1 Description 142
8.3.3.27.2 Attributes 142
8.3.3.28 VnfcSnapshotInfo information element 143
8.3.3.28.1 Description 143
8.3.3.28.2 Attributes 143
8.3.3.29 WanConnectionInfo information element 144
8.3.3.29.1 Description 144
8.3.3.29.2 Attributes 144
8.3.3.30 StorageSnapshotResource information element 144
8.3.3.30.1 Description 144
8.3.3.30.2 Attributes 144
8.3.3.31 TrunkPortsInfo information element 145
8.3.3.31.1 Description 145
8.3.3.31.2 Attributes 145
8.3.3.32 VipCpInfo information element 145
8.3.3.32.1 Description 145
8.3.3.32.2 Attributes 145
8.3.3.33 McioInfo information element 146
8.3.3.33.1 Description 146
8.3.3.33.2 Attributes 146
8.3.4 Information elements related to NS Lifecycle Management operations 146
8.3.4.1 Introduction 146
8.3.4.2 SapData information element 146
8.3.4.2.1 Description 146
8.3.4.2.2 Attributes 147
8.3.4.3 VnfInstanceData information element 147
8.3.4.3.1 Description 147
8.3.4.3.2 Attributes 147
8.3.4.4 VnfLocationConstraint information element 147
8.3.4.4.1 Description 147
8.3.4.4.2 Attributes 148
8.3.4.5 ParamsForVnf information element 148
8.3.4.5.1 Description 148
8.3.4.5.2 Attributes 148
8.3.4.6 ScaleNsData information element 148
8.3.4.6.1 Description 148
8.3.4.6.2 Attributes 148
8.3.4.7 ScaleNsByStepsData information element 149
8.3.4.7.1 Description 149
8.3.4.7.2 Attributes 150
8.3.4.8 ScaleNsToLevelData information element 150
8.3.4.8.1 Description 150
8.3.4.8.2 Attributes 150
8.3.4.9 ScaleVnfData information element 150
8.3.4.9.1 Description 150
8.3.4.9.2 Attributes 150
8.3.4.10 ScaleToLevelData information element 151
8.3.4.10.1 Description 151
8.3.4.10.2 Attributes 151
8.3.4.11 ScaleByStepData information element 152
8.3.4.11.1 Description 152
8.3.4.11.2 Attributes 152

8.3.4.12 InstantiateVnfData information element 152
8.3.4.12.1 Description 152
8.3.4.12.2 Attributes 152
8.3.4.13 ExtVirtualLinkData information element 154
8.3.4.13.1 Description 154
8.3.4.13.2 Attributes 154
8.3.4.14 VnfExtCpData information element 155
8.3.4.14.1 Description 155
8.3.4.14.2 Attributes 155
8.3.4.15 ChangeVnfFlavourData information element 155
8.3.4.15.1 Description 155
8.3.4.15.2 Attributes 155
8.3.4.16 OperateVnfData information element 156
8.3.4.16.1 Description 156
8.3.4.16.2 Attributes 156
8.3.4.17 ModifyVnfInfoData information element 157
8.3.4.17.1 Description 157
8.3.4.17.2 Attributes 157
8.3.4.18 Void 157
8.3.4.19 AssocNewNsdVersionData information element 158
8.3.4.19.1 Description 158
8.3.4.19.2 Attributes 158
8.3.4.20 MoveVnfInstanceData information element 158
8.3.4.20.1 Description 158
8.3.4.20.2 Attributes 158
8.3.4.21 AddVnffgData information element 159
8.3.4.21.1 Description 159
8.3.4.21.2 Attributes 159
8.3.4.22 UpdateVnffgData information element 159
8.3.4.22.1 Description 159
8.3.4.22.2 Attributes 159
8.3.4.23 NfpData information element 159
8.3.4.23.1 Description 159
8.3.4.23.2 Attributes 160
8.3.4.24 HealNsData information element 160
8.3.4.24.1 Description 160
8.3.4.24.2 Attributes 160
8.3.4.25 HealVnfData information element 161
8.3.4.25.1 Description 161
8.3.4.25.2 Attributes 161
8.3.4.26 AffinityOrAntiAffinityRule information element 161
8.3.4.26.1 Description 161
8.3.4.26.2 Attributes 161
8.3.4.27 ChangeNsFlavourData information element 162
8.3.4.27.1 Description 162
8.3.4.27.2 Attributes 162
8.3.4.28 ExtManagedVirtualLinkData information element 162
8.3.4.28.1 Description 162
8.3.4.28.2 Attributes 162
8.3.4.29 ChangeExtVnfConnectivityData information element 163
8.3.4.29.1 Description 163
8.3.4.29.2 Attributes 164
8.3.4.30 NfpRule Information element 164
8.3.4.30.1 Description 164
8.3.4.30.2 Attributes 164
8.3.4.31 PortRange Information element 165
8.3.4.31.1 Description 165
8.3.4.31.2 Attributes 165
8.3.4.32 AddPnfData information element 165
8.3.4.32.1 Description 165
8.3.4.32.2 Attributes 165
8.3.4.33 ModifyPnfData information element 166

8.3.4.33.1 Description 166
8.3.4.33.2 Attributes 166
8.3.4.34 PnfExtCpData information element 166
8.3.4.34.1 Description 166
8.3.4.34.2 Attributes 166
8.3.4.35 ExtLinkPortData information element 166
8.3.4.35.1 Description 166
8.3.4.35.2 Attributes 167
8.3.4.36 VnfExtCpConfig information element 167
8.3.4.36.1 Description 167
8.3.4.36.2 Attributes 167
8.3.4.37 NestedNsInstanceData information element 167
8.3.4.37.1 Description 167
8.3.4.37.2 Attributes 168
8.3.4.38 ParamsForNestedNs information element 168
8.3.4.38.1 Description 168
8.3.4.38.2 Attributes 168
8.3.4.39 RevertToSnapshotData information element 169
8.3.4.39.1 Description 169
8.3.4.39.2 Attributes 169
8.3.4.39a CreateSnapshotData information element 169
8.3.4.39a.1 Description 169
8.3.4.39a.2 Attributes 169
8.3.4.40 DeleteSnapshotData information element 169
8.3.4.40.1 Description 169
8.3.4.40.2 Attributes 169
8.3.4.41 PnfProfileData information element 170
8.3.4.41.1 Description 170
8.3.4.41.2 Attributes 170
8.3.4.42 VnfProfileData information element 170
8.3.4.42.1 Description 170
8.3.4.42.2 Attributes 171
8.3.4.43 WanConnectionData information element 171
8.3.4.43.1 Description 171
8.3.4.43.2 Attributes 171
8.3.4.44 VnfLinkPortData information element 171
8.3.4.44.1 Description 171
8.3.4.44.2 Attributes 171
8.3.4.45 ChangeVnfPackageData information element 172
8.3.4.45.1 Description 172
8.3.4.45.2 Attributes 172
8.3.4.46 ModifyWanConnectionInfoData information element 172
8.3.4.46.1 Description 172
8.3.4.46.2 Attributes 172
8.3.4.47 NestedNsLocationConstraint information element 173
8.3.4.47.1 Description 173
8.3.4.47.2 Attributes 173
8.3.4.48 TerminateNsData information element 173
8.3.4.48.1 Description 173
8.3.4.48.2 Attributes 173
8.3.4.49 TerminateVnfData information element 173
8.3.4.49.1 Description 173
8.3.4.49.2 Attributes 173
8.3.5 NsLcmCapacityShortageNotification 174
8.3.5.1 Description 174
8.3.5.2 Trigger conditions 174
8.3.5.3 Attributes 174
8.4 Information elements and notifications related to NS Performance Management 176
8.4.1 Introduction 176
8.4.2 ObjectSelection information element 176
8.4.2.1 Description 176
8.4.2.2 Attributes 176

8.4.3 PmJob information element 177
8.4.3.1 Description 177
8.4.3.2 Attributes 177
8.4.4 Threshold information element 178
8.4.4.1 Description 178
8.4.4.2 Attributes 178
8.4.5 PerformanceReport information element 178
8.4.5.1 Description 178
8.4.5.2 Attributes 178
8.4.6 PerformanceReportEntry information element 179
8.4.6.1 Description 179
8.4.6.2 Attributes 179
8.4.7 PerformanceValueEntry information element 179
8.4.7.1 Description 179
8.4.7.2 Attributes 179
8.4.8 PerformanceInformationAvailableNotification 180
8.4.8.1 Description 180
8.4.8.2 Trigger Conditions 180
8.4.8.3 Attributes 180
8.4.9 ThresholdCrossedNotification 180
8.4.9.1 Description 180
8.4.9.2 Trigger Conditions 180
8.4.9.3 Attributes 181
8.5 Information elements and notifications NS Fault management 181
8.5.1 Introduction 181
8.5.2 AlarmNotification 181
8.5.2.1 Description 181
8.5.2.2 Trigger conditions 181
8.5.2.3 Attributes 181
8.5.3 AlarmClearedNotification 182
8.5.3.1 Description 182
8.5.3.2 Trigger conditions 182
8.5.3.3 Attributes 182
8.5.4 Alarm information element 182
8.5.4.1 Description 182
8.5.4.2 Attributes 182
8.5.5 AlarmListRebuiltNotification 183
8.5.5.1 Description 183
8.5.5.2 Trigger conditions 183
8.5.5.3 Attributes 183
8.5.6 FaultyComponentInfo information element 184
8.5.6.1 Description 184
8.5.6.2 Attributes 184
8.5.7 FaultyResourceInfo information element 184
8.5.7.1 Description 184
8.5.7.2 Attributes 184
8.6 Information elements and notifications related to VNF Package 184
8.6.1 Introduction 184
8.6.2 VnfPkgInfo information element 185
8.6.2.1 Description 185
8.6.2.2 Attributes 185
8.6.3 Vnfd information element 186
8.6.3.1 Description 186
8.6.3.2 Attributes 186
8.6.4 VnfPackageSoftwareImageInfo information element 186
8.6.4.1 Description 186
8.6.4.2 Attributes 186
8.6.5 Void 187
8.6.6 VnfPackageArtifactInformation information element 187
8.6.6.1 Description 187
8.6.6.2 Attributes 187
8.6.7 Void 187

8.6.8 VnfPackageOnBoardingNotification 187
8.6.8.1 Description 187
8.6.8.2 Trigger Conditions 187
8.6.8.3 Attributes 187
8.6.9 VnfPackageChangeNotification 188
8.6.9.1 Description 188
8.6.9.2 Trigger Conditions 188
8.6.9.3 Attributes 188
8.7 Information elements and notifications related to NFVI Capacity Information 188
8.7.1 Description 188
8.7.2 CapacityThresholdCrossedNotification 189
8.7.2.1 Description 189
8.7.2.2 Trigger Conditions 189
8.7.2.3 Attributes 189
8.7.3 NfviCapacityThreshold information element 189
8.7.3.1 Description 189
8.7.3.2 Attributes 189
8.8 Information elements and notifications related to Policy Management 190
8.8.1 Introduction 190
8.8.2 Information elements related to Policy Management Operations 190
8.8.2.1 Introduction 190
8.8.2.2 PolicyInfo information element 190
8.8.2.2.1 Description 190
8.8.2.2.2 Attributes 190
8.8.3 PolicyChangeNotification 191
8.8.3.1 Description 191
8.8.3.2 Trigger Conditions 191
8.8.3.3 Attributes 191
8.8.4 PolicyConflictNotification 191
8.8.4.1 Description 191
8.8.4.2 Trigger Conditions 191
8.8.4.3 Attributes 191
8.9 Information elements related to VNF Snapshot Package Management 192
8.9.1 Introduction 192
8.9.2 VnfSnapshotPkgInfo information element 192
8.9.2.1 Description 192
8.9.2.2 Attributes 192
8.9.3 SnapshotPkgArtifactInformation information element 193
8.9.3.1 Description 193
8.9.3.2 Attributes 193
8.9.4 VnfcSnapshotImageInfo information element 193
8.9.4.1 Description 193
8.9.4.2 Attributes 193

Annex A (informative): Principles related to VNF lifecycle management and NS lifecycle management 195

Annex B (informative): Use cases for VNF reuse and referencing in NSs 196

B.1 Re-use of VNFs from a terminated NS 196

B.2 Creation of VNF instances in anticipation of future NS demand 196

B.3 Bottom-up NS instantiation 197

B.4 Shared VNF instances 197

Annex C (informative): Message flows for supporting use cases with fine grained NS lifecycle management 198

C.1 Introduction 198

C.2 New NS with VNF pools 198

C.3 New NS utilizing VNF instances from a VNF pool NS 200
C.4 Terminating NS instance with retained VNF instances 201

Annex D (informative): State models 203

D.1 VNF Package state model 203
D.1.1 Introduction 203
D.1.2 State model 203

D.2 NSD state model 204
D.2.1 Introduction 204
D.2.2 State model 204

D.3 NS state model 206
D.3.1 Introduction 206
D.3.2 State model 206

Annex E (informative): NS scaling 207

E.1 Forms of NS scaling 207

E.2 NS scaling triggers 207
E.2.1 NS auto-scale 207
E.2.2 NS scale triggered by OSS/BSS 207

E.3 Relation to NS DF 208

E.4 Input and tools for NS auto-scaling 208
E.4.1 Monitoring parameter 208
E.4.2 VNF indicator 208
E.4.3 Auto-scale policies/rules 208

Annex F (informative): Example interaction flows 209

F.1 LCM Coordination flow 209

Annex G (informative): Change history 210

History 213

Intellectual Property Rights

Essential patents

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

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

Trademarks

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

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.

Foreword

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

Modal verbs terminology

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

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

1 Scope

The present document defines the interfaces supported over the Os-Ma-nfvo reference point of the NFV-MANO architectural framework ETSI GS NFV 006 [i.10] as well as the information elements exchanged over those interfaces.

2 References

2.1 Normative references

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

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

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

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

[1] ETSI GS NFV-IFA 010: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Functional requirements specification".

[2] ETSI GS NFV-IFA 011: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; VNF Descriptor and Packaging Specification".

[3] ETSI GS NFV-IFA 014: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Network Service Templates Specification".

[4] Recommendation ITU-T X.733: "Information technology - Open Systems Interconnection - Systems Management: Alarm reporting function".

[5] ETSI GS NFV-IFA 027: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Performance Measurements Specification".

2.2 Informative references

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

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

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

[i.1] ETSI GR NFV 003: "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

[i.2] Void.

[i.3] ISO/IEC 9646-7: "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 7: Implementation Conformance Statements".

[i.4] ETSI GS NFV-IFA 005: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Or-Vi reference point - Interface and Information Model Specification".

[i.5] ETSI GS NFV-IFA 007: "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Or-Vnfm reference point - Interface and Information Model Specification".

[i.6] ETSI GS NFV-IFA 009: "Network Functions Virtualisation (NFV); Management and Orchestration; Report on Architectural Options".

[i.7] Void.

[i.8] IEEE 802.1Q™-2018: "IEEE Standard for Local and metropolitan area networks - Bridges and Bridged Networks".

[i.9] ETSI GR NFV-IFA 023: "Network Functions Virtualisation (NFV); Management and Orchestration; Report on Policy Management in Mano; Release 3".

[i.10] ETSI GS NFV 006: "Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Architectural Framework Specification".

3 Definition of terms, symbols and abbreviations

3.1 Terms

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

Compute MCIO: MCIO which declarative descriptor specifies compute infrastructure resource requests

Network MCIO: MCIO which declarative descriptor specifies network infrastructure resource requests

Storage MCIO: MCIO which declarative descriptor specifies storage infrastructure resource requests

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GR NFV 003 [i.1] apply.

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

4.1 Introduction

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

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

- Network Service Descriptor (NSD) Management (produced by the NFVO, consumed by the OSS/BSS).
- Network Service (NS) Lifecycle Management (produced by the NFVO, consumed by the OSS/BSS).
- NS Performance Management (produced by the NFVO, consumed by the OSS/BSS).
- NS Fault Management (produced by the NFVO, consumed by the OSS/BSS).
- VNF Package Management (produced by the NFVO, consumed by the OSS/BSS).

- Policy Management (produced by the NFVO, consumed by the OSS/BSS).
- VNF Snapshot Package Management (produced by the NFVO, consumed by the OSS/BSS).
- LCM Coordination (produced by OSS/BSS, consumed by NFVO).

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

4.2 Relation to other NFV group specifications

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

- Management and Orchestration - Report on Architectural Options ETSI GS NFV-IFA 009 [i.6]:
  - ETSI GS NFV-IFA 009 [i.6] provides architectural options that can influence the way some of the Os-Ma-nfvo interfaces are used or might even suggest the need for extension.
- Management and Orchestration - Functional requirements specification ETSI GS NFV-IFA 010 [1]:
  - Interfaces associated with the Os-Ma-nfvo reference point are based on the functional requirements specified in ETSI GS NFV-IFA 010 [1] for the NFVO FB.
- Management and Orchestration - Or-Vnfm reference point - Interface and Information Model Specification ETSI GS NFV-IFA 007 [i.5].

4.3 Conventions

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

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

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

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

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

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

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

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

5 Reference point and interface requirements

5.1 Introduction

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

5.2 Os-Ma-nfvo reference point requirements

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

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

Numbering	Functional requirement description
Os-Ma-nfvo.001	The Os-Ma-nfvo reference point shall support the NSD Management interface produced by the NFVO.
Os-Ma-nfvo.002	The Os-Ma-nfvo reference point shall support the NS Lifecycle Management interface produced by the NFVO.
Os-Ma-nfvo.003	Void.
Os-Ma-nfvo.004	The Os-Ma-nfvo reference point shall support the NS Performance Management interface produced by the NFVO.
Os-Ma-nfvo.005	The Os-Ma-nfvo reference point shall support the NS Fault Management interface produced by the NFVO.
Os-Ma-nfvo.006	The Os-Ma-nfvo reference point shall support the VNF Package Management interface produced by the NFVO.
Os-Ma-nfvo.007	Any interaction on the Os-Ma-nfvo reference point concerning a VNF shall be associated with at least one NS instance.
Os-Ma-nfvo.008	The Os-Ma-nfvo reference point shall support the NFVI Capacity Information interface produced by the NFVO.
Os-Ma-nfvo.009	The Os-Ma-nfvo reference point shall support the Policy Management interface produced by the NFVO.
Os-Ma-nfvo.010	The Os-Ma-nfvo reference point shall support the VNF Snapshot Package Management interface produced by the NFVO.
Os-Ma-nfvo.011	The Os-Ma-nfvo reference point may support the LCM Coordination interface produced by OSS/BSS (see note).
NOTE: The dependency on the LCM Coordination interface is declared in the NSD.	

5.3 Interface requirements

5.3.1 NSD Management interface requirements

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

Table 5.3.1-1: NSD management interface requirements

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

Numbering	Functional requirement description
Os-Ma-nfvo.Nsd.007	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications about the on-boarding of NSDs.
Os-Ma-nfvo.Nsd.008	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications as a result of changes on NSD states.
Os-Ma-nfvo.Nsd.009	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support on-boarding Physical Network Function Descriptor (PNFD).
Os-Ma-nfvo.Nsd.010	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support updating a PNFD.
Os-Ma-nfvo.Nsd.011	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting a PNFD.
Os-Ma-nfvo.Nsd.012	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying PNFDs.
Os-Ma-nfvo.Nsd.013	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support subscribing to notifications related to NSD management changes.
Os-Ma-nfvo.Nsd.014	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support fetching an NSD, or selected artifacts contained in an NSD archive.
Os-Ma-nfvo.Nsd.015	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support fetching a PNFD, or selected artifacts contained in an NSD archive.
Os-Ma-nfvo.Nsd.016	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications about the on-boarding of PNFDs.
Os-Ma-nfvo.Nsd.017	The NSD Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications about the deletion of PNFDs.
NOTE: Updating an NSD can include changing the operational state of the NSD and updating the user defined data.	

5.3.2 NS Lifecycle Management interface requirements

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

Table 5.3.2-1: Network service lifecycle management interface requirements

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

Numbering	Functional requirement description
Os-Ma-nfvo.NsLcm.014	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support scaling a VNF instance explicitly as part of the scaling of an NS.
Os-Ma-nfvo.NsLcm.015	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying information about a VNF instance as part of the query of an NS.
Os-Ma-nfvo.NsLcm.016	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support healing a VNF instance explicitly as part of the healing of an NS.
Os-Ma-nfvo.NsLcm.017	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support changing the state of a VNF instance explicitly as part of the update of an NS. See note 5.
Os-Ma-nfvo.NsLcm.018	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support changing the deployment flavour (DF) of a VNF instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.019	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support modifying information and/or the configuration parameters of a VNF instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.036	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support changing the external connectivity of a VNF instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.021	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing additional affinity or anti-affinity rules when instantiating an NS.
Os-Ma-nfvo.NsLcm.035	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support creating an NS instance identifier and the associated instance of an NS information element.
Os-Ma-nfvo.NsLcm.022	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting an NS instance identifier and the associated instance of an NS information element.
Os-Ma-nfvo.NsLcm.023	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support adding/removing an existing nested NS instance to/from an NS instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.024	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support adding new Service Access Point (SAP) to an NS and removing existing SAP from an NS explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.025	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support associating a new NSD version to an existing NS instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.026	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support moving existing VNF instance(s) from one NS instance (source) to another NS instance (destination) explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.027	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support adding a new VNFFG to an NS instance, remove existing VNF Forwarding Graph (VNFFG) and updating a VNFFG from an NS instance explicitly as part of the update of an NS.
Os-Ma-nfvo.NsLcm.028	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying the status of an NS lifecycle management operation.
Os-Ma-nfvo.NsLcm.029	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing to the OSS/BSS notifications about changes of an NS instance that are related to NS lifecycle management operations.
Os-Ma-nfvo.NsLcm.030	Notifications provided on the NS lifecycle interface produced by the NFVO on the Os-Ma-nfvo reference point shall contain information about the type of the NS lifecycle change, the addition/deletion/modification of VNFs and/or Physical Network Functions (PNFs), about change in the connectivity between elements of the NS. See note 7.
Os-Ma-nfvo.NsLcm.031	Notifications provided on the NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall contain information about the VLs and VNFFGs that are added/modified/deleted as part of the NS lifecycle operation. See note 6.
Os-Ma-nfvo.NsLcm.032	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support notifying the result (successful or failed) of NS instantiation with indicating the NS instance Id.
Or-Ma-nfvo.NsLcm.033	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing to the OSS/BSS notifications about creation and deletion of an NS instance identifier and the associated instance of an NS information element, further referred to as NS identifier creation/deletion notifications.
Or-Ma-nfvo.NsLcm.034	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support subscribing to NS lifecycle change notifications and to NS identifier creation/deletion notifications.

Numbering	Functional requirement description
Or-Ma-nfvo.NsLcm.037	The NS Lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the capability to invoke NS error handling operation(s) after the NS life cycle operation occurrence fails. See note 8 and note 9.
Or-Ma-nfvo.NsLcm.038	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support creating VNF Snapshots explicitly as part of the update of an NS.
Or-Ma-nfvo.NsLcm.039	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support reverting to VNF Snapshots explicitly as part of the update of an NS.
Or-Ma-nfvo.NsLcm.040	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting available VNF Snapshot information explicitly as part of the update of an NS.
Or-Ma-nfvo.NsLcm.041	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying information about available VNF Snapshots as part of the query of an NS.
Os-Ma-nfvo.NsLcm.043	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support changing the current VNF package of a VNF instance explicitly as part of the update of an NS.
Or-Ma-nfvo.NsLcm.042	Notifications provided on the NS lifecycle interface produced by the NFVO on the Os-Ma-nfvo reference point shall include appropriate error information. See note 10.
Os-Ma-nfvo.NsLcm.044	The NS lifecycle management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support modifying information and/or the configuration parameters of WAN connectivity as part of the update of an NS.
Os-Ma-nfvo.NsLcm.045	The NS lifecycle Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing descriptor identifiers of NSD constituents to override the identifiers indicated in the NSD when these constituents are instantiated or added to the NS. See notes 11 and 12.
NOTE 1: The existing VNF instance(s) may need to be modified as part of NS instantiation. NOTE 2: The existing VNF instance(s) may need to be modified as part of NS update. NOTE 3: A VNF instance or a nested NS instance can be shared between NS instances managed by the same NFVO. NOTE 4: If the VNF instance being removed is no longer part of any NS instance, it will be terminated. NOTE 5: Changing the state of a VNF instance refers to starting or stopping a VNF instance. These operations are complementary to instantiating or terminating a VNF. NOTE 6: This provides information about VLs and VNFFGs points used by the NS and whose creation was triggered by the NFVO. NOTE 7: The modification of VNFs includes VNF scaling, change of VNF flavours, VNF healing, change of VNF operational state, modification of VNF information, and/or VNF configuration parameters and the change of VNF external connectivity. NOTE 8: The details of the error handling operation(s) are part of the protocol design. NOTE 9: It depends on the NS capabilities whether and how the operation(s) are supported by a particular NS. NOTE 10: In case of resource shortage and parallel LCM operations, appropriate error information includes information about pre-emption by higher priority operations. NOTE 11: This requirement does not imply a modification of the NSD. NOTE 12: When it affects new instances of VNFs, nested NSs or PNFs, the descriptor referred in the interface will be used for the instantiation. When it affects existing instances of VNFs and nested NSs to be included in the NS, these instances shall have been based on the descriptors indicated in the interface.	

5.3.3 Void

5.3.4 NS Performance Management interface requirements

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

Table 5.3.4-1: Network service performance management interface requirements

Numbering	Requirements description
Os-Ma-nfvo.NsPm.001	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to control the collection and reporting of performance information for NSs.
Os-Ma-nfvo.NsPm.002	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the capability to notify the availability of performance information. See note 1.
Os-Ma-nfvo.NsPm.003	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall expose the type of performance information that the NFVO can collect from the NSs.
Os-Ma-nfvo.NsPm.004	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to create a PM job specifying the type of resource(s) and performance information that the OSS/BSS requires.
Os-Ma-nfvo.NsPm.005	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to create a PM job specifying the granularity for collection and reporting of performance information on NSs.
Os-Ma-nfvo.NsPm.006	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to delete one or more explicitly identified PM job(s).
Os-Ma-nfvo.NsPm.007	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support periodic collection of performance information (bounded or unbounded).
Os-Ma-nfvo.NsPm.008	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference shall support the grouping of measurements. See note 2.
Os-Ma-nfvo.NsPm.009	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to manage the thresholds on the performance information collected by the NFVO for NSs. See note 3.
Os-Ma-nfvo.NsPm.010	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the capability to notify about a threshold defined for a specified metric of NSs being crossed.
Os-Ma-nfvo.NsPm.011	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to receive notifications related to threshold crossing.
Os-Ma-nfvo.NsPm.012	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point should support querying the list of active PM jobs and defined threshold conditions by the consumer entity that created them.
Os-Ma-nfvo.NsPm.013	The NS Performance Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the deletion of threshold conditions on the performance information collected by the NFVO for NSs.

NOTE 1: Performance information on a given NS results from either collected performance information of the virtualised resources impacting the connectivity of this NS instance or VNF performance information issued by the VNFM for the VNFs that is part of this NS instance. The latter performance information also results from collected performance information of the virtualised resources.

NOTE 2: The group does not imply any modification/aggregation of performance measurements data and may be viewed as an alias for a pre-defined list of measurements. The group can be created by VNF, by NS, by virtual machine, etc.

NOTE 3: Management of thresholds includes creation, deletion and query of the thresholds on the performance information collected.

5.3.5 NS Fault Management interface requirements

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

Table 5.3.5-1: Network service fault management interface requirements

Numbering	Requirements description
Os-Ma-nfvo.NsFm.001	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support collecting NSs fault information. See note.
Os-Ma-nfvo.NsFm.002	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing alarm notifications related to faults on NSs to the OSS/BSS.
Os-Ma-nfvo.NsFm.003	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing a notification when there is a change in the alarm information on NS.
Os-Ma-nfvo.NsFm.004	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the sending of notification to the OSS/BSS when an alarm on an NS has been created.
Os-Ma-nfvo.NsFm.005	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the sending of notification to the OSS/BSS when an alarm on an NS has been cleared.
Os-Ma-nfvo.NsFm.006	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall allow unambiguous identification of the alarm on an NS sent to the OSS/BSS.
Os-Ma-nfvo.NsFm.007	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall allow unambiguous identification of the NS causing the alarm.
Os-Ma-nfvo.NsFm.008	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall allow unambiguous identification of the alarm cause.
Os-Ma-nfvo.NsFm.009	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support alarm acknowledgement.
Os-Ma-nfvo.NsFm.010	The NS Fault Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support the sending of notification to the OSS/BSS when the alarm list has been rebuilt.
NOTE: Fault information on a given NS instance can include the information related to the alarm (e.g. alarm created, alarm cleared, etc.), alarm cause(s) and identification of this NS instance and fault information concerning the virtualised resources supporting the constituent VNFs for this NS instance and the virtualised resources supporting the connectivity of this NS instance.	

5.3.6 VNF Package Management interface requirements

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

Table 5.3.6-1: VNF Package management interface requirements

Numbering	Functional requirement description
Os-Ma-nfvo.VnfPkgm.001	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support on-boarding a VNF Package.
Os-Ma-nfvo.VnfPkgm.002	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support disabling a VNF Package.
Os-Ma-nfvo.VnfPkgm.003	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support enabling a VNF Package.
Os-Ma-nfvo.VnfPkgm.004	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying VNF Package information. See note 1.
Os-Ma-nfvo.VnfPkgm.005	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting a VNF Package.
Os-Ma-nfvo.VnfPkgm.006	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications about the on-boarding of VNF Packages.
Os-Ma-nfvo.VnfPkgm.007	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications as a result of changes on VNF Package states.
Os-Ma-nfvo.VnfPkgm.008	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support fetching a VNF Package, or selected artifacts contained in a package.

Numbering	Functional requirement description
Os-Ma-nfvo.VnfPkgm.009	Void.
Os-Ma-nfvo.VnfPkgm.010	The VNF Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support updating VNF Package information. See note 2.
NOTE 1: VNF Package information can include information such as release date, vendor info, manifest, VNFD, SW image meta-data, files contained in the VNF Package, etc. NOTE 2: Updating VNF package information can include changing the operational state of the VNF package and updating the user defined data.	

5.3.7 NFVI Capacity Information interface

Table 5.3.7-1 specifies requirements applicable to the NFVI Capacity Information interface produced by the NFVO on the Os-Ma-nfvo reference point.

Table 5.3.7-1: NFVI Capacity Information interface requirements

Numbering	Functional requirement description
Os-Ma-nfvo.NfviCi.001	The NFVI Capacity Information interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying NFVI capacity information.
Os-Ma-nfvo.NfviCi.002	The NFVI Capacity Information interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing notifications about capacity shortage and capacity threshold crossing in the NFVI. See note 1.
Os-Ma-nfvo.NfviCi.00a	The NFVI Capacity Information interface produced by the NFVO on the Os-Ma-nfvo reference point shall enable the OSS/BSS to manage the thresholds on the NFVI capacity information collected by the NFVO. See note 2.
NOTE 1: Capacity shortage definitions are declared via NFVI capacity thresholds. NOTE 2: Management of thresholds includes creation, deletion and query of the thresholds on the NFVI capacity information collected.	

5.3.8 Policy Management interface requirements

Table 5.3.8-1 specifies requirements applicable to the policy management interface produced by the NFVO on the Os-Ma-nfvo reference point.

Table 5.3.8-1: Policy management interface requirements

Numbering	Requirements description
Os-Ma-nfvo.Plcm.001	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support transferring NFV-MANO policies. See notes 1, 2 and 3.
Os-Ma-nfvo.Plcm.002	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting NFV-MANO policies. See note 1.
Os-Ma-nfvo.Plcm.003	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying NFV-MANO policies. See note 1.
Os-Ma-nfvo.Plcm.004	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support activating NFV-MANO policies. See note 1.
Os-Ma-nfvo.Plcm.005	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deactivating NFV-MANO policies. See note 1.
Os-Ma-nfvo.Plcm.006	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing to the OSS/BSS notifications about changes of a policy that are related to operations of transferring policy, deleting policy, activating policy, deactivating policy, associating policy and disassociating policy.
Os-Ma-nfvo.Plcm.007	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support providing to the OSS/BSS notifications about any detected policy conflicts.
Os-Ma-nfvo.Plcm.008	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support subscribing to policy management related notifications.
Os-Ma-nfvo.Plcm.009	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support associating NFV-MANO policies to NS instances.
Os-Ma-nfvo.Plcm.010	The Policy Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support disassociating NFV-MANO policies from NS instances.

Numbering	Requirements description
NOTE 1: For this reference point, NFV-MANO policies include policies applied in NS lifecycle management (instantiation, scaling, update, healing and termination). NOTE 2: The case of transferring NFV-MANO policy applies when: - a new policy is imported from the OSS, which results in the creation of a new policy locally; or - the changes for an existing policy are imported from the OSS, which results in the update of a policy locally. NOTE 3: The OSS/BSS could also transfer, via the NFVO, NFV-MANO polices (related to VNF lifecycle management) to the VNFM, or transfer NFV-MANO polices (related to virtualised resource management) to the VIM.	

5.3.9 VNF Snapshot Package Management interface requirements

Table 5.3.9-1 specifies requirements applicable to the VNF Snapshot Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point.

Table 5.3.9-1: VNF Snapshot Package Management interface requirements

Numbering	Requirement
Os-Ma-nfvo.VnfSnapPkgm.001	The VNF Snapshot Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support creating VNF Snapshot Package information elements.
Os-Ma-nfvo.VnfSnapPkgm.002	The VNF Snapshot Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support building VNF Snapshot Packages.
Os-Ma-nfvo.VnfSnapPkgm.003	The VNF Snapshot Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support uploading VNF Snapshot Packages.
Os-Ma-nfvo.VnfSnapPkgm.004	The VNF Snapshot Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support extracting VNF Snapshot Packages.
Os-Ma-nfvo.VnfSnapPkgm.005	The VNF Snapshot Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support querying information about available VNF Snapshot Packages (see note).
Os-Ma-nfvo.VnfSnapPkgm.006	The VNF Snapshot Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support fetching a VNF Snapshot Package, or selected artifacts contained in a VNF Snapshot Package.
Os-Ma-nfvo.VnfSnapPkgm.007	The VNF Snapshot Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support deleting VNF Snapshot Packages.
Os-Ma-nfvo.VnfSnapPkgm.008	The VNF Snapshot Package Management interface produced by the NFVO on the Os-Ma-nfvo reference point shall support updating a VNF Snapshot Package.
NOTE: VNF Snapshot Package information can include information such as creation date, configuration data of included snapshots, and files contained in the VNF Snapshot Package.	

5.3.10 LCM Coordination interface requirements

Table 5.3.10-1 specifies requirements applicable to the LCM Coordination interface produced by the OSS/BSS on the Os-Ma-nfvo reference point.

Table 5.3.10-1: LCM Coordination interface requirements

Numbering	Requirement
Os-Ma-nfvo.LcmCoord.001	The LCM Coordination interface produced by the OSS/BSS on the Os-Ma-nfvo reference point shall support receiving requests to coordinate LCM operations affecting an NS instance and its VNF instances.

6 OSS exposed interfaces

6.1 LCM Coordination interface

6.1.1 Description

This interface enables a NFVO to request OSS/BSS to coordinate LCM operations.

The following operations shall be supported:

- CoordinateLcmOperation.

6.1.2 CoordinateLcmOperation operation

6.1.2.1 Description

This operation enables an NFVO to request the coordination of an LCM operation with management operations executed in the OSS/BSS on an NS instance by invoking a coordination action towards the OSS/BSS. The coordination can be needed at various stages of the LCM operation. The corresponding execution of the LCM operation within the NFVO will be paused until the response is received or no response is received until the expiration of a timer.

The response includes an indication whether to continue or abort the LCM operation execution and may include additional information.

The CoordinateLcmOperation operation allows the following use cases:

1) Following the execution of a coordination action, the execution of the LCM operation occurrence is continued immediately.

2) Following the execution of a coordination action, the execution of the LCM operation occurrence is aborted immediately.

3) Following the execution of a coordination action, the execution of the LCM operation occurrence is continued after a delay.

4) Following the execution of a coordination action, the same coordination action is retried after a delay.

Table 6.1.2.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

NOTE: The cancellation of a pending coordination is part of the protocol design.

Table 6.1.2.1-1: CoordinateLcmOperation operation

Information Flow	Requirement	Direction
CoordinateLcmOperationRequest	Mandatory	NFVO → OSS/BSS
CoordinateLcmOperationResponse	Mandatory	OSS/BSS → NFVO

6.1.2.2 Input parameters

The input parameters sent when invoking the operation are provided in table 6.1.2.2-1.

Table 6.1.2.2-1: CoordinateLcmOperation operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Uniquely identifies the NS instance.
lifecycleOperationOccurrenceId	M	1	Identifier	The identifier of LCM operation occurrence concerning the coordination.
lcmOperationType	M	1	Not specified	Indicates the type of LCM operation concerning the coordination.

Parameter	Qualifier	Cardinality	Content	Description
coordinationActionName	M	1	Identifier	Indicates the LCM coordination action.
inputParam	M	0..N	KeyValuePair	Additional parameters passed as input to the coordination action.
NOTE 1: Void. NOTE 2: Void.				

6.1.2.3 Output parameters

The output parameters sent when responding to the operation is provided in table 6.1.2.3-1.

Table 6.1.2.3-1: CoordinateLcmOperation operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
coordinationResult	M	1	Enum	Indicates how the LCM operation execution to be continued. VALUES: • ABORT • CONTINUE • CONTINUE_AFTER_DELAY • RETRY_AFTER_DELAY • etc. See note.
operationResumeDelay	M	0..1	TimeDuration	The time interval to wait until the new coordination request is to be sent ("RETRY AFTER DELAY" coordinationResult) or the LCM operation execution can be resumed ("CONTINUE AFTER DELAY" coordinationResult). Shall be present in case the coordinationResult indicates a delay and shall be absent otherwise. See note.
outputParam	M	0..N	KeyValuePair	Additional information on the result of the coordination with the OSS/BSS, e.g. on the reason for the indicated coordinationResult.
NOTE: Either the NFVO or the OSS/BSS can execute the waiting cycle in the use cases that include a delay. In case the OSS/BSS executes the waiting cycle, the "operationAction" values "CONTINUE_AFTER_DELAY" and "RETRY_AFTER_DELAY" and the "operationResumeDelay" attribute are not applicable.				

6.1.2.4 Operation results

As a result of this operation, the OSS/BSS shall indicate to the NFVO in the CoordinateLcmOperationResponse message whether the requested coordination with the OSS/BSS was successful or not.

Upon reception of a CoordinateLcmOperationResponse message, the NFVO has gathered information from the OSS/BSS to continue, delay (if applicable) or abort an LCM operation based on the success of the coordination with the OSS/BSS.

7 NFVO exposed interfaces

7.1 Introduction

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

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

7.2 NSD Management interface

7.2.1 Description

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

The following operations are defined for this interface:

- Create NSD Info.
- Upload NSD.
- Fetch NSD.
- Update NSD Info.
- Delete NSD.
- Query NSD Info.
- Fetch NSD Archive Artifacts.
- Create PNFD Info.
- Upload PNFD.
- Fetch PNFD.
- Update PNFD Info.
- Delete PNFD.
- Query PNFD Info.
- Fetch PNFD Archive Artifacts.
- Subscribe, for subscribing to notifications related to NSD and PNFD management changes.
- Notify, for delivering notifications related to NSD and PNFD management changes.
- Terminate Subscription operation: for terminating a particular subscription related to NSD and PNFD management changes.
- Query Subscription Info operation: for querying subscription information related to NSD and PNFD management changes.

In the present document, the on-boarding of an NSD or PNFD includes:

1) Creating an NSD information object or PNFD information object.
2) Uploading the NSD or PNFD.
3) Processing the NSD or PNFD, including validation, inside the NFVO.

An NSD or PNFD is referred as "on-boarded" only after these three procedures are successfully accomplished.

7.2.2 Upload NSD operation

7.2.2.1 Description

This operation will upload an NSD to the NFVO. An NSD information object shall be created a priori via the Create NSD Info operation. Only one NSD is allowed per NSD information object.

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

All descriptors needed by the NSD: VNFD, PNFD and NSD for nested NSs shall be on-boarded before this operation.

The user may use this operation to upload a new NSD version, which can be associated to an NS instance with the Update NS operation (see clause 7.3.5). Different NSD versions have the same "nsdInvariantId" attribute, but different "nsdId" attributes and different NSD information objects. The design of different NSD versions and their business logic is out of scope of the present document.

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

Table 7.2.2.1-1: Upload NSD operation

Message	Requirement	Direction
UploadNsdRequest	Mandatory	OSS/BSS → NFVO
UploadNsdResponse	Mandatory	NFVO → OSS/BSS

7.2.2.2 Input parameters

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

Table 7.2.2.2-1: Upload NSD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier	Identifier of the NSD information object associated with the NSD to be uploaded.
nsd	M	1	Nsd	NSD to be uploaded.

7.2.2.3 Output parameters

No output parameter.

7.2.2.4 Operation results

The result of the operation indicates whether the uploading and processing, including validation, of the NSD has been successful or not with a standard success/error result.

After a successful result, the NSD and all referenced descriptors are known to and validated by the NFVO. The associated NSD information object is updated with the information populated from the validated NSD. The NSD is on-boarded and is in "ENABLED, NOT_IN_USE" state, allowing its use for NS lifecycle management. See the NSD state model in clause D.2.

7.2.3 Void

7.2.4 Void

7.2.5 Update NSD Info operation

7.2.5.1 Description

This operation will update the user defined data and/or the operational state of an existing NSD information object. The usage state shall not change as a result of the operation.

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

Table 7.2.5.1-1: Update NSD Info operation

Message	Requirement	Direction
UpdateNsdInfoRequest	Mandatory	OSS/BSS → NFVO
UpdateNsdInfoResponse	Mandatory	NFVO → OSS/BSS

7.2.5.2 Input parameters

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

Table 7.2.5.2-1: Update NSD Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier	Identifier of the NSD information object to be updated.
userDefinedData	O	0..N	KeyValuePair	User defined data to be updated. For existing keys, the value is replaced. See note.
operationalState	M	0..1	Enum	Operational state of the on-boarded NSD. VALUES: • ENABLED • DISABLED See note.
NOTE: At least one of the two parameters shall be present. If NSD is not on-boarded, the operation is used only to update existing or add additional user defined data using the userDefinedData parameter.				

7.2.5.3 Output parameters

No output parameter.

7.2.5.4 Operation results

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

7.2.6 Delete NSD operation

7.2.6.1 Description

This operation will delete one or more NSD(s). The associated NSD information objects will be deleted as well.

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

An NSD can only be deleted when there is no NS instance using it and the operational state is DISABLED.

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

Table 7.2.6.1-1: Delete NSD operation

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

7.2.6.2 Input parameters

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

Table 7.2.6.2-1: Delete NSD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1..N	Identifier	Identifier of the NSD information objects to be deleted. See note.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to delete multiple NSDs in one request, or as a series of requests that delete one NSD at a time.				

7.2.6.3 Output parameters

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

Table 7.2.6.3-1: Delete NSD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deletedNsdInfoId	M	1..N	Identifier	Identifier of the deleted NSD information objects.

7.2.6.4 Operation results

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

7.2.7 Query NSD Info operation

7.2.7.1 Description

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

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

Table 7.2.7.1-1: Query NSD Info operation

Message	Requirement	Direction
QueryNsdInfoRequest	Mandatory	OSS/BSS → NFVO
QueryNsdInfoResponse	Mandatory	NFVO → OSS/BSS

7.2.7.2 Input parameters

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

Table 7.2.7.2-1: Query NSD Info operation input parameters

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

7.2.7.3 Output parameters

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

Table 7.2.7.3-1: Query NSD Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	NsdInfo	Details of the NSD information objects matching the input filter.

7.2.7.4 Operation results

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

7.2.8 Upload PNFD operation

7.2.8.1 Description

This operation will upload a PNFD to the NFVO, making it available to be used by NSDs. A PNFD information object shall be created a priori via the Create PNFD Info operation. Only one PNFD is allowed per PNFD information object.

The user may use this operation to upload a new PNFD version. Different PNFD versions have the same "pnfdInvariantId" attribute, but different "pnfdId" attributes and different PNFD information objects. The design of different PNFD versions and their business logic is out of scope of the present document.

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

Table 7.2.8.1-1: Upload PNFD operation

Message	Requirement	Direction
UploadPnfdRequest	Mandatory	OSS/BSS → NFVO
UploadPnfdResponse	Mandatory	NFVO → OSS/BSS

7.2.8.2 Input parameters

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

Table 7.2.8.2-1: Upload PNFD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier	Identifier of the PNFD information object associated with the PNFD to be uploaded.
pnfdArchive	M	1	Binary	Archive file containing the PNFD to be uploaded.

7.2.8.3 Output parameters

No output parameter.

7.2.8.4 Operation results

The result of the operation indicates whether the uploading and processing, including validation, of the PNFD has been successful or not with a standard success/error result.

Once on-boarded, the PNFD is known to and validated by the NFVO. The associated PNFD information object is updated with the information populated from the validated PNFD.

7.2.9 Update PNFD Info operation

7.2.9.1 Description

This operation will update the user defined data of an existing PNFD information object.

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

Table 7.2.9.1-1: Update PNFD Info operation

Message	Requirement	Direction
UpdatePnfdInfoRequest	Mandatory	OSS/BSS → NFVO
UpdatePnfdInfoResponse	Mandatory	NFVO → OSS/BSS

7.2.9.2 Input parameters

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

Table 7.2.9.2-1: Update PNFD Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier	Identifier of the PNFD information object to be updated.
userDefinedData	O	0..N	KeyValuePair	User defined data to be updated. For existing Keys, the value is replaced.

7.2.9.3 Output parameters

No output parameter.

7.2.9.4 Operation results

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

7.2.10 Delete PNFD operation

7.2.10.1 Description

This operation will delete one or more PNFDs. The associated PNFD information objects will be deleted as well.

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

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

Table 7.2.10.1-1: Delete PNFD operation

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

7.2.10.2 Input parameters

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

Table 7.2.10.2-1: Delete PNFD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1..N	Identifier	Identifier of the PNFD information object(s) to be deleted.
applyOnAllVersions	O	0..1	Boolean	Indicates if the delete operation is to be applied on all versions of these PNFD instances. By default, if not present, it applies only on the current version.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to delete multiple PNFD in one request, or as a series of requests that delete one PNFD at a time.				

7.2.10.3 Output parameters

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

Table 7.2.10.3-1: Delete PNFD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deletedPnfdInfoId	M	1..N	Identifier	Identifier of the deleted PNFD information objects.

7.2.10.4 Operation results

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

7.2.11 Query PNFD Info operation

7.2.11.1 Description

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

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

Table 7.2.11.1-1: Query PNFD Info operation

Message	Requirement	Direction
QueryPnfdInfoRequest	Mandatory	OSS/BSS → NFVO
QueryPnfdInfoResponse	Mandatory	NFVO → OSS/BSS

7.2.11.2 Input parameters

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

Table 7.2.11.2-1: Query PNFD Info operation input parameters

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

7.2.11.3 Output parameters

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

Table 7.2.11.3-1: Query PNFD Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	PnfdInfo	Details of the PNFD information objects matching the input filter.

7.2.11.4 Operation results

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

7.2.12 Subscribe operation

7.2.12.1 Description

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

NOTE: Specification of the filtering mechanism is part of the protocol design.

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

Table 7.2.12.1-1: Subscribe operation

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

7.2.12.2 Input parameters

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

Table 7.2.12.2-1: Subscribe operation input parameters

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

7.2.12.3 Output parameters

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

Table 7.2.12.3-1: Subscribe operation output parameters

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

7.2.12.4 Operation results

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

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

7.2.13 Notify operation

7.2.13.1 Description

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

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

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

Table 7.2.13.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

The following notification is sent by this operation:

- NsdOnBoardingNotification. See clause 8.2.6.
- NsdChangeNotification. See clause 8.2.7.
- NsdDeletionNotification. See clause 8.2.8.
- PnfdOnBoardingNotification. See clause 8.2.9.
- PnfdDeletionNotification. See clause 8.2.10.

7.2.14 Terminate Subscription operation

7.2.14.1 Description

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

Table 7.2.14.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.14.1-1: Terminate Subscription operation

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

7.2.14.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.14.2-1.

Table 7.2.14.2-1: Terminate Subscription operation input parameters

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

7.2.14.3 Output parameters

No output parameter.

7.2.14.4 Operation results

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

7.2.15 Query Subscription Info operation

7.2.15.1 Description

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

Table 7.2.15.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.15.1-1: Query Subscription operation

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

7.2.15.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.15.2-1.

Table 7.2.15.2-1: Query Subscription Info operation input parameters

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

7.2.15.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.15.3-1.

Table 7.2.15.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query.

7.2.15.4 Operation results

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

7.2.16 Create NSD Info operation

7.2.16.1 Description

This operation will create an NSD information object in the NFVO for the NSD to be uploaded.

Table 7.2.16.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.16.1-1: Create NSD Info operation

Message	Requirement	Direction
CreateNsdInfoRequest	Mandatory	OSS/BSS → NFVO
CreateNsdInfoResponse	Mandatory	NFVO → OSS/BSS

7.2.16.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.16.2-1.

Table 7.2.16.2-1: Create NSD Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
userDefinedData	O	0..N	KeyValuePair	User defined data for the NSD to be uploaded.

7.2.16.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.16.3-1.

Table 7.2.16.3-1: Create NSD Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier	Identifier of the created NSD information object.

7.2.16.4 Operation results

The result of the operation indicates whether the creation of NSD information object has been successful or not with a standard success/error result.

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

7.2.17 Fetch NSD operation

7.2.17.1 Description

This operation will fetch an NSD from the NFVO.

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

Table 7.2.17.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.17.1-1: Fetch NSD operation

Message	Requirement	Direction
FetchNsdRequest	Mandatory	OSS/BSS → NFVO
FetchNsdResponse	Mandatory	NFVO → OSS/BSS

7.2.17.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.17.2-1.

Table 7.2.17.2-1: Fetch NSD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier	Identifier of the NSD information object associated with the NSD to be fetched.

7.2.17.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.17.3-1.

Table 7.2.17.3-1: Fetch NSD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
nsd	M	1	Nsd	The fetched NSD.

7.2.17.4 Operation results

The result of the operation indicates whether the fetching of the NSD has been successful or not in the NFVO with a standard success/error result.

7.2.18 Create PNFD Info operation

7.2.18.1 Description

This operation will create a PNFD information object in the NFVO for the PNFD to be uploaded.

Table 7.2.18.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.18.1-1: Create PNFD Info operation

Message	Requirement	Direction
CreatePnfdInfoRequest	Mandatory	OSS/BSS → NFVO
CreatePnfdInfoResponse	Mandatory	NFVO → OSS/BSS

7.2.18.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.18.2-1.

Table 7.2.18.2-1: Create PNFD Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
userDefinedData	O	0..N	KeyValuePair	User defined data for the PNFD to be uploaded.

7.2.18.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.18.3-1.

Table 7.2.18.3-1: Create PNFD Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier	Identifier of the created PNFD information object.

7.2.18.4 Operation results

The result of the operation indicates whether the creation of PNFD information object has been successful or not with a standard success/error result.

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

7.2.19 Fetch PNFD operation

7.2.19.1 Description

This operation will fetch a PNFD from the NFVO.

Table 7.2.19.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.2.19.1-1: Fetch PNFD operation

Message	Requirement	Direction
FetchPnfdRequest	Mandatory	OSS/BSS → NFVO
FetchPnfdResponse	Mandatory	NFVO → OSS/BSS

7.2.19.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.19.2-1.

Table 7.2.19.2-1: Fetch PNFD operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier	Identifier of the PNFD information object associated with the PNFD to be fetched.

7.2.19.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.19.3-1.

Table 7.2.19.3-1: Fetch PNFD operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfdArchive	M	1	Binary	The archive file containing the PNFD.

7.2.19.4 Operation results

The result of the operation indicates whether the fetching of the PNFD has been successful or not in the NFVO with a standard success/error result.

7.2.20 Fetch NSD Archive Artifacts operation

7.2.20.1 Description

This operation enables the OSS/BSS to fetch selected artifacts contained in an NSD archive. Artifacts are addressed using selector information that can be obtained using the Query NSD Info operation.

Table 7.2.20.1-1 lists the information flow exchanged between the OSS and the NFVO.

Table 7.2.20.1-1: Fetch NSD Archive Artifacts operation

Message	Requirement	Direction
FetchNsdArchiveArtifactsRequest	Mandatory	OSS → NFVO
FetchNsdArchiveArtifactsResponse	Mandatory	NFVO → OSS

7.2.20.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.20.2-1.

Table 7.2.20.2-1: Fetch NSD Archive Artifacts operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier	Identifier of NSD information object associated with the NSD archive artifacts to be fetched.
artifactSelector	M	1..N	Not specified	Selector to address an individual NSD archive artifact, or list of selectors to address multiple of those. See note.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to obtain multiple artifacts in one go, or as a series of operations that obtain one artifact at a time.				

7.2.20.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.20.3-1.

Table 7.2.20.3-1: Fetch NSD Archive Artifacts operation output parameters

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

7.2.20.4 Operation results

After successful operation, the NFVO has provided to the OSS a copy/copies of the requested artifact(s) contained in the NSD archive.

7.2.21 Fetch PNFD Archive Artifacts operation

7.2.21.1 Description

This operation enables the OSS/BSS to fetch selected artifacts contained in a PNFD archive. Artifacts are addressed using selector information that can be obtained using the Query PNFD Info operation.

Table 7.2.21.1-1 lists the information flow exchanged between the OSS and the NFVO.

Table 7.2.21.1-1: Fetch PNFD Archive Artifacts operation

Message	Requirement	Direction
FetchPnfdArchiveArtifactsRequest	Mandatory	OSS → NFVO
FetchPnfdArchiveArtifactsResponse	Mandatory	NFVO → OSS

7.2.21.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.21.2-1.

Table 7.2.21.2-1: Fetch PNFD Archive Artifacts operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier	Identifier of PNFD information object associated with the PNFD archive artifacts to be fetched.
artifactSelector	M	1..N	Not specified	Selector to address an individual PNFD archive artifact, or list of selectors to address multiple of those. See note.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to obtain multiple artifacts in one go, or as a series of operations that obtain one artifact at a time.				

7.2.21.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.21.3-1.

Table 7.2.21.3-1: Fetch PNFD Archive Artifacts operation output parameters

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

7.2.21.4 Operation results

After successful operation, the NFVO has provided to the OSS a copy/copies of the requested artifact(s) contained in the PNFD archive.

7.3 NS Lifecycle Management interface

7.3.1 Description

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

The following operations are defined for this interface:

- Create NS Identifier.
- Instantiate NS.
- Scale NS.
- Update NS.
- Query NS.
- Terminate NS.

- Delete NS Identifier.
- Heal NS.
- Get Operation Status.

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

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

7.3.2 Create NS Identifier operation

7.3.2.1 Description

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

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

Table 7.3.2.1-1: Create NS Identifier operation

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

7.3.2.2 Input parameters

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

Table 7.3.2.2-1: Create NS Identifier operation input parameters

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

7.3.2.3 Output parameters

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

Table 7.3.2.3-1: Create NS Identifier operation output parameters

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

7.3.2.4 Operation results

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

7.3.3 Instantiate NS operation

7.3.3.1 Description

This operation will instantiate an NS, will run a feasibility check of the NS instantiation, or will run a feasibility check followed by the actual NS instantiation. This operation can only be used with an NS instance in the NOT_INSTANTIATED state.

NOTE 1: When only feasibility check is requested without instantiating the NS instance, OSS/BSS needs to first create a (temporary) NS instance Id using the Create NS Identifier operation (see clause 7.3.2) that is used as input parameter to the Instantiate NS operation for the feasibility check.

NOTE 2: After the termination of the NS instance or after completion of feasibility check when "feasibility check only" was requested, the OSS/BSS is responsible to delete the NS instance Id. Refer to Delete NS Identifier operation (see clause 7.3.8).

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

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

If the NS LCM operation is a scheduled operation, it shall be possible to modify the schedule time, whereas it shall not be possible to change any other operational parameters. Specification of handling the change of the schedule time is part of the protocol design.

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

Table 7.3.3.1-1: Instantiate NS operation

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

7.3.3.2 Input parameters

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

Table 7.3.3.2-1: Instantiate NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the instance of the NS.
flavourId	M	1	Identifier (Reference to NsDf)	Flavour of the NSD used to instantiate this NS. See note 1.
sapData	M	0..N	SapData	Create data concerning the SAPs of this NS.
addPnfData	M	0..N	AddPnfData	Information on the PNF(s) that are part of this NS.
vnfInstanceData	M	0..N	VnfInstanceData	Specify an existing VNF instance to be used in the NS. If needed, the VNF Profile to be used for this VNF instance is also provided. See note 2.
nestedNsInstanceData	M	0..N	NestedNsInstanceData	Specify an existing NS instance to be used as a nested NS within the NS. If needed, the NS Profile to be used for this nested NS instance is also provided. See notes 3 and 4.

Parameter	Qualifier	Cardinality	Content	Description
locationConstraints	M	0..N	VnfLocationConstraint	Defines the location constraints for the VNF to be instantiated as part of the NS instantiation. An example can be a constraint for the VNF to be in a specific geographic location.
nestedNsLocationConstraints	M	0..N	NestedNsLocationConstraint	Defines the location constraints for the nested NS to be instantiated as part of the NS instantiation. An example can be a constraint for the nested NS to be in a specific geographic location.
additionalParamForNs	M	0..N	KeyValuePair	Allows the OSS/BSS to provide additional parameter(s) at the composite NS level (as opposed to the VNF level, which is covered in additionalParamForVnf, and as opposed to the nested NS level, which is covered in additionalParamForNestedNs).
additionalParamForNestedNs	M	0..N	ParamsForNestedNs	Allows the OSS/BSS to provide additional parameter(s) per nested NS instance (as opposed to the composite NS level, which is covered in additionalParamForNs, and as opposed to the VNF level, which is covered in additionalParamForVnf). This is for nested NS instances that are to be created by the NFVO as part of the NS instantiation and not for existing nested NS instances that are referenced for reuse.
additionalParamForVnf	M	0..N	ParamsForVnf	Allows the OSS/BSS to provide additional parameter(s) per VNF instance (as opposed to the composite NS level, which is covered in additionalParamForNs, and as opposed to the nested NS level, which is covered in additionalParamForNestedNs). This is for VNFs that are to be created by the NFVO as part of the NS instantiation and not for existing VNF that are referenced for reuse.
startTime	M	0..1	DateTime	Indicates the earliest time to instantiate the NS. Cardinality "0" indicates the NS instantiation takes place immediately. See note 6.
nsInstantiationLevelId	M	0..1	Identifier	Identifies one of the NS instantiation levels declared in the DF applicable to this NS instance. See note 11.

Parameter	Qualifier	Cardinality	Content	Description
targetNsScaleLevelInfo	M	0..N	NsScaleInfo	This attribute is applicable for NS target scale level instantiation. This attribute defines the target NS scale level to which the NS instance is to be instantiated for each NS scaling aspect of the DF applicable to this NS instance. See notes 11 and 12.
wanConnectionData	M	0..N	WanConnectionData	Information for connecting VNs to the WAN when VLs are deployed across a WAN. See note 5.
additionalAffinityOrAntiAffinityRule	M	0..N	AffinityOrAntiAffinityRule	Specifies additional affinity or anti-affinity constraint for the VNF instances to be instantiated as part of the NS instantiation. Shall not conflict with rules already specified in the NSD.
feasibilityCheckReserve	M	0..1	Enum	Specifies the feasibility check and reserve option for the NS instantiation operation. VALUES: • NO_FEASIBILITY_CHECK • FEASIBILITY_CHECK_ONLY • FEASIBILITY_CHECK_WITH_OPERATION • FEASIBILITY_CHECK_WITH_RESERVATION_AND_OPERATION Default option is NO_FEASIBILITY_CHECK. If not present, default value applies. See notes 6, 7, 8, 9 and 10.

NOTE 1: The NsDf information element is defined in ETSI GS NFV-IFA 014 [3], clause 6.3.2.
NOTE 2: The DF of the VNF instance shall match the VNF DF present in the associated VNF Profile. The VNFD of the VNF instance shall match the one in the associated VNF Profile if no overridingVnfdId is provided in the interface as part of vnfInstanceData, or the overridingVnfdId provided in the interface if it is provided.
NOTE 3: The NS DF of each nested NS shall be one of the allowed flavours in the associated NSD (as referenced in the nestedNsd attribute of the NSD of the NS to be instantiated).
NOTE 4: The NSD of each referenced NSs (i.e. each nestedInstanceId) shall match the one of the nested NSD in the composite NSD if no overridingNsdId is provided in the interface as part of NestedNsInstanceData, or the overridingNsdId provided in the interface if it is provided.
NOTE 5: When the NS is deployed over several sites, the VLs of this NS will include VNs in each site connected over the WAN. In this case, the wanConnectionData provides the needed information required to connect each VN to the WAN.
NOTE 6: When feasibilityCheckReserve is set to FEASIBILITY_CHECK_ONLY, the startTime parameter shall be ignored.
NOTE 7: When feasibilityCheckReserve is set to NO_FEASIBILITY_CHECK, feasibility check is not performed. Resources might be reserved as triggered by means outside the present operation request (e.g. configuration or policies), e.g. if it is a schedule NS instantiation operation. See clause A.2.3 in ETSI GS NFV-IFA 010 [1].
NOTE 8: When feasibilityCheckReserve is set to FEASIBILITY_CHECK_ONLY, only feasibility check is performed for the operation, no resources are reserved.
NOTE 9: When feasibilityCheckReserve is set to FEASIBILITY_CHECK_WITH_OPERATION, feasibility check is performed and based on the feasibility check result, the actual NS instantiation operation is performed at the time indicated by the startTime parameter, if provided. Resources might be reserved as triggered by means outside the present operation request (e.g. configuration or policies), e.g. if it is a schedule NS instantiation operation. See clause A.2.3 in ETSI GS NFV-IFA 010 [1].
NOTE 10: When feasibilityCheckReserve is set to FEASIBILITY_CHECK_WITH_RESERVATION_AND_OPERATION, feasibility check is performed and based on the feasibility check result, resources are reserved as part of the feasibility check, and the actual NS instantiation operation is performed using the resources reserved during the feasibility check, and at the time indicated by the startTime parameter, if provided.
NOTE 11: The target size for NS instantiation may be specified in either nsInstantiationLevelId or targetNsScaleLevelInfo, but not both. If none of the two parameters (nsInstantiationLevelId or targetNsScaleLevelInfo) are present, the default NS instantiation level as declared in the NSD shall be used.

Parameter	Qualifier	Cardinality	Content	Description
NOTE 12: If targetNsScaleLevelInfo is specified, information provided in targetNsScaleLevelInfo shall be used to calculate the number of instances of the VNFs/nested NSs referred in the NSD. For VNFs or nested NSs that are not scalable based on targetNsScaleLevelInfo, the default NS instantiation level as declared in the NSD shall be used to calculate the number of instances those VNFs/nested NSs.				

7.3.3.3 Output parameters

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

Table 7.3.3.3-1: Instantiate NS operation output parameters

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

7.3.3.4 Operation results

The NFVO shall return a lifecycleOperationOccurrenceId that identifies the LCM operation. The LCM operation shall trigger the sending of the "start" NS LCM Operation Occurrence Notification (see clause 8.3.2.2) before additional notifications as part of this operation are triggered, or operations towards the VNFM or VIM are invoked.

If only feasibility check is required (feasibilityCheckReserve parameter value is "FEASIBILITY_CHECK_ONLY"), instantiation of the NS is not performed. In case of success, the feasibility check has been performed and the NFVO shall send the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2). In case of failure, appropriate error information is provided in the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2).

If the actual NS instantiation operation is requested to be performed after the feasibility check (feasibilityCheckReserve parameter value is "FEASIBILITY_CHECK_WITH_OPERATION" or "FEASIBILITY_CHECK_WITH_RESERVATION_AND_OPERATION"), after successful feasibility check, the NFVO shall send the feasibility check result as a "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2) and proceeds with the instantiation of NS. In case of failure of the feasibility check, appropriate error information is provided in the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2) and the NFVO will not proceed with the NS instantiation.

In case of successful completion of the NS instantiation operation, the NS has been instantiated. In case of failure during NS instantiation, appropriate error information is provided in the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2).

On the successful as well as the unsuccessful completion of the operation, the NFVO shall send the "result" NS LCM Operation Occurrence Notification.

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

7.3.4 Scale NS operation

7.3.4.1 Description

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

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

If the NS LCM operation is a scheduled operation, it shall be possible to modify the schedule time, whereas it shall not be possible to change any other operational parameters. Specification of handling the change of the schedule time is part of the protocol design.

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

Table 7.3.4.1-1: Scale NS operation

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

7.3.4.2 Input parameters

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

Table 7.3.4.2-1: Scale NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the instance of the NS.
scaleType	M	1	Enum	Indicates the type of scaling to be performed. VALUES: • SCALE_NS • SCALE_VNF
scaleNsData	M	0..1	ScaleNsData	Provides the necessary information to scale the referenced NS instance. It shall be present when scaleType = SCALE_NS. See note.
scaleVnfData	M	0..N	ScaleVnfData	Provides the information to scale a given VNF instance that is part of the referenced NS instance. Shall be present when scaleType = SCALE_VNF. See note.
scaleTime	M	0..1	DateTime	Indicates when the NS will be scaled. Cardinality "0" indicates the NS scaling takes place immediately.
NOTE: Either scaleNsData or scaleVnfData, but not both, shall be present.				

7.3.4.3 Output parameters

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

Table 7.3.4.3-1: Scale NS operation output parameters

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

7.3.4.4 Operation results

In case of success, the NS instance has been scaled according to the request. In case of failure, appropriate error information is provided in the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2).

The NFVO shall return a lifecycleOperationOccurrenceId that identifies the LCM operation. The LCM operation shall trigger the sending of the "start" NS LCM Operation Occurrence Notification (see clause 8.3.2.2) before additional notifications as part of this operation are triggered, or operations towards the VNFM or VIM are invoked.

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

On the successful as well as the unsuccessful completion of the operation, the NFVO shall send the "result" NS LCM Operation Occurrence Notification.

7.3.5 Update NS operation

7.3.5.1 Introduction

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

- Adding existing VNF instances to the NS instance.
- Removing VNF instances from the NS instance.
- Instantiating new VNF instances and adding them to the NS instance.
- Changing the DF of VNF instances belonging to the NS instance.
- Changing the operational state of a VNF instance belonging to the NS instance.
- Modifying information data and/or the configurable properties of a VNF instance belonging to the NS instance.
- Changing the external connectivity of a VNF instance belonging to the NS instance.
- Adding SAPs to the NS instance.
- Removing SAPs from the NS instance.
- Adding existing NS instances to the NS instance.
- Removing nested NS instances from the NS instance.
- Associating a new NSD version to the NS instance.
- Moving VNF instances from one NS instance to another NS instance.
- Adding VNFFGs to the NS instance.
- Removing VNFFGs from the NS instance.
- Updating VNFFGs of the NS instance.
- Changing the DF of the NS instance.
- Adding PNFs to the NS instance.
- Modifying PNFs in the NS instance.
- Removing PNFs from the NS instance.
- Creating VNF Snapshots of VNF instances belonging to the NS instance.
- Reverting to VNF Snapshots of VNF instances belonging to the NS instance.
- Deleting available VNF Snapshot information for VNF instances belonging to the NS instance.
- Associating a PNF with a new or updated PnfProfile.
- Associating a VNF instance with a new or updated VnfProfile.
- Changing current VNF package of a VNF instance belonging to the NS instance.
- Creating a new NsVirtualLink instance.
- Deleting an existing NsVirtualLink instance.

- Modifying WAN connectivity information.

Only one type of update shall be allowed per operation.

If the NS LCM operation is a scheduled operation, it shall be possible to modify the schedule time, whereas it shall not be possible to change any other operational parameters. Specification of handling the change of the schedule time is part of the protocol design.

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

Table 7.3.5.1-1: Update NS operation

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

7.3.5.2 Input parameters

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

Table 7.3.5.2-1: Update NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the NS instance being updated.
updateType	M	1	Not specified	Specifies the type of update. This parameter determines also which one of the following parameters is present in the operation. Possible values are: • AddVnf (adding existing VNF instance(s)); • RemoveVnf (removing VNF instance(s)); • InstantiateVnf (instantiating new VNF(s)); • ChangeVnfDf (Changing VNF DF); • OperateVnf (changing VNF state); • ModifyVnfInformation (modifying VNF information and/or the configurable properties of VNF instance(s)); • ChangeExtVnfConnectivity (changing the external connectivity of VNF instance(s)); • AddSap (adding SAP(s)); • RemoveSap (removing SAP(s)); • AddNestedNs (adding existing NS instance(s) as nested NS(s)); • RemoveNestedNs (removing existing nested NS instance(s)); • AssocNewNsdVersion (associating a new NSD version to the NS instance), • MoveVnf (moving VNF instance(s) from one origin NS instance to another target NS instance); • AddVnffg (adding VNFFG(s)); • RemoveVnffg (removing VNFFG(s)); • UpdateVnffg (updating VNFFG(s)); • ChangeNsDf (changing NS DF); • AddPnf (adding PNF); • ModifyPnf(modify PNF); • RemovePnf (removing PNF); • CreateSnapshot (creating VNF Snapshot). See note 3;

Parameter	Qualifier	Cardinality	Content	Description
				• RevertToSnapshot (reverting to VNF Snapshot). See notes 3 and 4; • DeleteSnapshotInfo (deleting available VNF Snapshot information). See note 3; • AssocPnfWithPnfProfile (associating a PNF with a new or updated PnfProfile); • AssocVnfWithVnfProfile (associating a VNF instance with a new or updated VnfProfile); • ChangeVnfPkg (Changing current VNF package). See note 14; • CreateNsVirtualLink (Creating a VL instance); • DeleteNsVirtualLink (Deleting a VL instance); • ModifyWanConnectionInfo (modifying WAN connection info).
addVnfInstance	M	0..N	VnfInstanceData	Specify an existing VNF instance to be added to the NS instance. This parameter shall be present only if updateType=AddVnf.
removeVnfInstanceId	M	0..N	Identifier	Specify an existing VNF instance to be removed from the NS instance. The parameter contains the identifier(s) of the VNF instances to be removed. This parameter shall be present only if updateType=RemoveVnf. See note 1.
instantiateVnfData	M	0..N	InstantiateVnfData	Specify the new VNF to be instantiated. This parameter can be used e.g. for the bottom-up NS creation. This parameter shall be present only if updateType=InstantiateVnf.
terminateVnfData	M	0..N	TerminateVnfData	Provides the information needed to terminate VNF instance(s). This parameter shall be present only if updateType= RemoveVnf and if the VNF instance(s) is(are) to be terminated as part of Update NS operation. See notes 1 and 13.
changeVnfFlavourData	M	0..N	ChangeVnfFlavourData	Specify the new DF of the VNF instance to be changed to. This parameter shall be present only if updateType=ChangeVnfDf.
operateVnfData	M	0..N	OperateVnfData	Specify the state of the VNF instance to be changed. This parameter shall be present only if updateType=OperateVnf.
modifyVnfInfoData	M	0..N	ModifyVnfInfoData	Specify the VNF Information parameters and/or the configurable properties of VNF instance to be modified. This parameter shall be present only if updateType=ModifyVnfInformation.
changeExtVnfConnectivityData	M	0..N	ChangeExtVnfConnectivityData	Specify the new external connectivity data of the VNF instance to be changed. This parameter shall be present only if updateType= ChangeExtVnfConnectivity.
addSap	M	0..N	SapData	Specify a new SAP to be added to the NS instance. This parameter shall be present only if updateType=AddSap.
removeSapId	M	0..N	Identifier	Specify an existing SAP to be removed from the NS instance. The parameter shall be present only if updateType=RemoveSap.
addNestedNsData	M	0..N	NestedNsInstanceData	Specify an existing nested NS instance to be added to (nested within) the NS instance. This parameter shall be present only if updateType=AddNestedNs.

Parameter	Qualifier	Cardinality	Content	Description
removeNestedNsId	M	0..N	Identifier	Specify an existing nested NS instance to be removed from the NS instance. The parameter shall be present only if updateType=RemoveVnfNestedNs.
assocNewNsdVersionData	M	0..1	AssocNewNsdVersionData	Specify the new NSD to be used for the NS instance. This parameter shall be present only if updateType=AssocNewNsdVersion.
moveVnfInstanceData	M	0..N	MoveVnfInstanceData	Specify existing VNF instance to be moved from one NS instance to another NS instance. This parameter shall be present only if updateType=MoveVnf.
addVnffg	M	0..N	AddVnffgData	Specify the new VNFFG to be created to the NS Instance. This parameter shall be present only if updateType=AddVnffg.
removeVnffgId	M	0..N	Identifier	Identifier of an existing VNFFG to be removed from the NS Instance. This parameter shall be present only if updateType=RemoveVnffg.
updateVnffg	M	0..N	UpdateVnffgData	Specify the new VNFFG Information data to be updated for a VNFFG of the NS Instance. This parameter shall be present only if updateType=UpdateVnffg.
changeNsFlavourData	M	0..1	ChangeNsFlavourData	Specifies the new DF to be applied to the NS instance. It shall be present only if updateType=ChangeNsDf.
updateTime	M	0..1	DateTime	Indicates when the NS will be updated. Cardinality "0" indicates the NS update takes place immediately. See note 8.
addPnfData	M	0..N	AddPnfData	Information of the PNF(s) that are being added into the NS instance. This parameter shall be present only if updateType=AddPnf.
modifyPnfData	M	0..N	ModifyPnfData	Information on the PNF(s) that are being modified in this NS instance. This parameter shall be present only if updateType=ModifyPnf. See note 2.
removePnfId	M	0..N	Identifier	Identifier of the PNF(s) that are part of this NS instance and that should be deleted from it. This parameter shall be present only if updateType=RemovePnf.
createSnapshotData	M	0..1	CreateSnapshotData	Specify the VNF instance to be snapshotted. This parameter shall be present only if updateType=CreateSnapshot.
revertToSnapshotData	M	0..1	RevertToSnapshotData	Specify the VNF instance to be reverted and the VNF Snapshot to be reverted to. This parameter shall be present only if updateType=RevertToSnapshot.
deleteSnapshotData	M	0..1	DeleteSnapshotData	Specify the VNF Snapshot info to be deleted and the related VNF instance. This parameter shall be present only if updateType=DeleteSnapshotInfo.
associatePnfWithPnfProfile	M	0..N	PnfProfileData	Specify the data needed for associating PNF with the new or updated PnfProfile. This parameter shall be present only if updateType=AssocPnfWithPnfProfile. See note 5.
associateVnfWithVnfProfile	M	0..N	VnfProfileData	Specify the data needed for associating VNF instance with the new or updated VnfProfile. This parameter shall be present only if updateType=AssocVnfWithVnfProfile. See note 6.
changeVnfPkgData	M	0..N	ChangeVnfPackageData	Specify the information for changing the current VNF package of the VNF instance. This parameter shall be present only if updateType=ChangeVnfPkg.
nsVirtualLinkProfile	M	0..N	Identifier (Reference to VirtualLinkProfile)	Reference the virtual link profile to be used to create a new NsVirtualLink instance. This parameter shall be present only if updateType=CreateNsVirtualLink. See note 7.

Parameter	Qualifier	Cardinality	Content	Description
deleteNsVirtualLinkId	M	0..N	Identifier	Identify an existing NsVirtualLink instance to be deleted. The parameter shall be present only if updateType=DeleteNsVirtualLink.
modifyWanConnectionInfoData	M	0..N	ModifyWanConnectionInfoData	Specifies the data to modify about WAN related connectivity information. The parameter shall be present only if updateType=ModifyWanConnectionInfo.
feasibilityCheckReserve	M	0..1	Enum	Specifies the feasibility check and reserve option for the Update NS instantiation operation. VALUES: • NO_FEASIBILITY_CHECK • FEASIBILITY_CHECK_ONLY • FEASIBILITY_CHECK_WITH_OPERATION • FEASIBILITY_CHECK_WITH_RESERVATION_AND_OPERATION Default option is NO_FEASIBILITY_CHECK. If not present, default value applies. See notes 8, 9, 10, 11 and 12.

NOTE 1: If a VNF instance is removed from an NS and this NS was the last one for which this VNF instance was a part, the VNF instance is terminated by the NFVO.
NOTE 2: New CP addresses should be contained in the element, if PNF CPs need to be changed.
NOTE 3: It depends on the VNF capabilities, and is declared in the VNFD (refer to the "supportedOperations" attribute in the VnfDf information element; see clause 7.1.8.2 in ETSI GS NFV-IFA 011 [2]), whether the operation is supported for a particular VNF.
NOTE 4: The operation might be service-disruptive.
NOTE 5: A new version of NSD with the new or updated PnfProfile needs to be uploaded, and associated with the NS prior to invoking the NS update operation.
NOTE 6: A new version of NSD with the new or updated VnfProfile needs to be uploaded, and associated with the NS prior to invoking the NS update operation.
NOTE 7: All NsVirtualLink instances of a particular NS DF based on a specific NsVirtualLinkDesc have the same characteristics as they use the same VirtualLinkProfile.
NOTE 8: When feasibilityCheckReserve is set to FEASIBILITY_CHECK_ONLY, the updateTime parameter shall be ignored.
NOTE 9: When feasibilityCheckReserve is set to NO_FEASIBILITY_CHECK, feasibility check is not performed. Resources might be reserved as triggered by means outside of the present operation request (e.g. configuration or policies), e.g. if it is a schedule Update NS operation. See clause A.2.3 in ETSI GS NFV-IFA 010 [1].
NOTE 10: When feasibilityCheckReserve is set to FEASIBILITY_CHECK_ONLY, only feasibility check is performed for the operation, no resources are reserved.
NOTE 11: When feasibilityCheckReserve is set to FEASIBILITY_CHECK_WITH_OPERATION, feasibility check is performed and based on the feasibility check result, the actual Update NS operation is performed at the time indicated by the updateTime parameter, if provided. Resources might be reserved as triggered by means outside of the present operation request (e.g. configuration or policies), e.g. if it is a schedule Update NS operation. See clause A.2.3 in ETSI GS NFV-IFA 010 [1].
NOTE 12: When feasibilityCheckReserve is set to FEASIBILITY_CHECK_WITH_RESERVATION_AND_OPERATION, feasibility check is performed and based on the feasibility check result, resources are reserved as part of the feasibility check, and the actual Update NS operation is performed using the resources reserved during the feasibility check, and at the time indicated by the updateTime parameter, if provided.
NOTE 13: For each of the referred vnfInstanceId in the terminateVnfData, there shall be a corresponding value in the removeVnfInstanceId.
NOTE 14: A change of VNFD version of a constituent VNF where the source and target VNFD versions have the same vnfdExtInvariantId can also be performed without a change in the NSD by a sequence of two Update NS operations, the first one of type RemoveVnf, the second one of type AddVnf or of type InstantiateVnf; or with the Scale NS operation, removing the existing VNF instance and adding a VNF instance with the target VNFD.

7.3.5.3 Output parameters

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

Table 7.3.5.3-1: Update NS operation output parameters

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

7.3.5.4 Operation results

The NFVO shall return a lifecycleOperationOccurrenceId that identifies the LCM operation. The LCM operation shall trigger the sending of the "start" NS LCM Operation Occurrence Notification (see clause 8.3.2.2) before additional notifications as part of this operation are triggered, or operations towards the VNFM or VIM are invoked.

If only feasibility check is required (feasibilityCheckReserve parameter value is "FEASIBILITY_CHECK_ONLY"), update of the NS is not performed. In case of success, the feasibility check has been performed and the NFVO shall send the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2). In case of failure, appropriate error information is provided in the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2).

If the actual Update NS operation is requested to be performed after the feasibility check (feasibilityCheckReserve parameter value is "FEASIBILITY_CHECK_WITH_OPERATION" or "FEASIBILITY_CHECK_WITH_RESERVATION_AND_OPERATION"), after successful feasibility check, the NFVO shall send the feasibility check result as a "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2) and proceeds with the update operation. In case of failure of the feasibility check, appropriate error information is provided in the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2) and the NFVO will not proceed with the Update NS operation.

In case of successful completion of the Update NS operation, the NS has been updated according to the request. In case of failure during Update NS, appropriate error information is provided in the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2).

On successful as well as unsuccessful completion of the operation, the NFVO shall send the "result" NS LCM Operation Occurrence Notification.

7.3.6 Query NS operation

7.3.6.1 Description

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

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

Table 7.3.6.1-1: Query NS operation

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

7.3.6.2 Input parameters

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

Table 7.3.6.2-1: Query NS operation input parameters

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

7.3.6.3 Output parameters

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

Table 7.3.6.3-1: Query NS operation output parameters

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

7.3.6.4 Operation results

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

7.3.7 Terminate NS operation

7.3.7.1 Description

This operation will terminate an NS. This can also be used to terminate an NS instance that is scheduled to be instantiated (i.e. a startTime parameter had been provided in the Instantiate NS operation).

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

If the NS LCM operation is a scheduled operation, it shall be possible to modify the schedule time, whereas it shall not be possible to change any other operational parameters. Specification of handling the change of the schedule time is part of the protocol design.

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

Table 7.3.7.1-1: Terminate NS operation

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

7.3.7.2 Input parameters

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

Table 7.3.7.2-1: Terminate NS operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the NS instance to terminate.
terminateTime	M	0..1	DateTime	Indicates the end time of the NS, i.e. the NS will be terminated automatically at this time. Cardinality "0" indicates the NS termination takes place immediately.
terminateNsData	M	0..1	Terminate NsData	Provides additional parameter(s) to the termination process at the NS level. See note 1.
terminateVnfData	M	0..N	Terminate VnfData	Provides the information needed to terminate VNF instance(s). See notes 1 and 2.
NOTE 1: Information needed for terminating specific VNF instances shall only be specified in the "terminateVnfData", and not in the "terminateNsData". NOTE 2: VNF instance(s) part of this NS instance is(are) terminated as part of Terminate NS operation only if the instance(s) is(are) not used by any other NS instance.				

7.3.7.3 Output parameters

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

Table 7.3.7.3-1: Terminate NS output parameters

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

7.3.7.4 Operation results

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

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

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

In case of failure, appropriate error information is provided in the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2).

The NFVO shall return a lifecycleOperationOccurrenceId that identifies the LCM operation. The LCM operation shall trigger the sending of the "start" NS LCM Operation Occurrence Notification (see clause 8.3.2.2) before additional notifications as part of this operation are triggered, or operations towards the VNFM or VIM are invoked.

On successful as well as unsuccessful completion of the operation, the NFVO shall send the "result" NS LCM Operation Occurrence Notification.

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

7.3.8 Delete NS Identifier operation

7.3.8.1 Description

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

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

Table 7.3.8.1-1: Delete NS Identifier operation

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

7.3.8.2 Input parameters

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

Table 7.3.8.2-1: Delete NS Identifier operation input parameters

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

7.3.8.3 Output parameters

No output parameter.

7.3.8.4 Operation results

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

In case of failure, appropriate error information is returned.

7.3.9 Heal NS operation

7.3.9.1 Description

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

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

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

Table 7.3.9.1-1: Heal NS operation

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

7.3.9.2 Input parameters

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

Table 7.3.9.2-1: Heal NS operation input parameters

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

7.3.9.3 Output parameters

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

Table 7.3.9.3-1: Heal NS output parameters

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

7.3.9.4 Operation results

In case of success, the NS has been healed, that means complete or partial healing as requested. In case of failure, appropriate error information is provided in the "result" NS LCM Operation Occurrence Notification (see clause 8.3.2.2).

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

The NFVO shall return a lifecycleOperationOccurrenceId that identifies the LCM operation. The LCM operation shall trigger the sending of the "start" NS LCM Operation Occurrence Notification (see clause 8.3.2.2) before additional notifications as part of this operation are triggered, or operations towards the VNFM or VIM are invoked.

On the successful as well as the unsuccessful completion of the operation, the NFVO shall send the "result" NS LCM Operation Occurrence Notification.

7.3.10 Get Operation Status operation

7.3.10.1 Description

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

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

Table 7.3.10.1-1: Get Operation Status operation

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

7.3.10.2 Input parameters

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

Table 7.3.10.2-1: Get Operation Status operation input parameters

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

7.3.10.3 Output Parameters

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

Table 7.3.10.3-1: Get Operation Status operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
operationStatus	M	1	Not specified	Indicates the operation status (which includes, for example: Processing, Successfully done, Failed, but can also include operation-specific states).

7.3.10.4 Operation results

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

7.3.11 Subscribe operation

7.3.11.1 Description

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

NOTE: Specification of the filtering mechanism is part of the protocol design.

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

Table 7.3.11.1-1: Subscribe operation

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

7.3.11.2 Input parameters

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

Table 7.3.11.2-1: Subscribe operation input parameters

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

7.3.11.3 Output parameters

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

Table 7.3.11.3-1: Subscribe operation output parameters

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

7.3.11.4 Operation results

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

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

7.3.12 Notify operation

7.3.12.1 Description

This operation notifies a subscriber about events related to notifications about lifecycle operation occurrences on NS instance, lifecycle operation occurrences impacting NS components, as well as the creation/deletion of NS instance identifiers and the associated NsInfo information element instances.

This operation distributes notifications to subscribers. It is a one-way operation issued by the producer (NFVO) that cannot be invoked as an operation by the consumer (OSS/BSS). In order to receive notifications, the consumer (OSS/BSS) has to perform an explicit Subscribe operation beforehand.

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

Table 7.3.12.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

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

- NsLcmOperationOccurrenceNotification. See clause 8.3.2.2.
- NsChangeNotification. See clause 8.3.2.11.
- NsIdentifierCreationNotification. See clause 8.3.2.9.
- NsIdentifierDeletionNotification. See clause 8.3.2.10.
- NsLcmCapacityShortageNotification. See clause 8.3.5.

7.3.13 Terminate Subscription operation

7.3.13.1 Description

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

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

Table 7.3.13.1-1: Terminate Subscription operation

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

7.3.13.2 Input parameters

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

Table 7.3.13.2-1: Terminate Subscription operation input parameters

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

7.3.13.3 Output parameters

No output parameter.

7.3.13.4 Operation results

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

7.3.14 Query Subscription Info operation

7.3.14.1 Description

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

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

Table 7.3.14.1-1: Query Subscription Info operation

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

7.3.14.2 Input parameters

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

Table 7.3.14.2-1: Query Subscription Info operation input parameters

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

7.3.14.3 Output parameters

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

Table 7.3.14.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query.

7.3.14.4 Operation results

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

7.4 Void

7.5 NS Performance Management interface

7.5.1 Description

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

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

Performance information on a given NS related measured object instance (see note) results from either collected performance information of the virtualised resources impacting the connectivity of this NS related measured object instance or VNF performance information, resulting from virtualised resource performance information, issued by the VNFM related to the VNFs that are part of this NS instance.

NOTE: The NS related measured object instance is the instance of one of the measured object type(s) for which the performance measurements applicable to Os-Ma-nfvo reference point are defined in clause 7.3 of ETSI GS NFV-IFA 027 [5].

When new performance information is available, the consumer is notified using the notification PerformanceInformationAvailableNotification (see clause 8.4.8). The details of the performance measurements are provided using the PerformanceReport information element (see clause 8.4.5). Delivery mechanism for the performance reports is not specified in the present document.

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

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

7.5.2 Create PM Job operation

7.5.2.1 Description

This operation will create a PM job, enabling an OSS/BSS to specify one or more NS related measured object(s), that the NFVO is managing, for which it wants to receive performance information. This will allow the requesting OSS/BSS to specify its performance information requirements with the NFVO.

The OSS/BSS needs to be subscribed to receive PerformanceInformationAvailable notifications in order to know when new collected performance information is available.

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

Table 7.5.2.1-1: Create PM Job operation

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

7.5.2.2 Input parameters

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

Table 7.5.2.2-1: Create PM Job operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
objectSelector	M	1	ObjectSelection	Defines the NS related measured object(s) for which performance information is to be collected.
performanceMetric	M	0..N	String	This defines the type of performance metric(s) for the specified measured object(s). At least one of the two attributes (performance metric or metricGroup) shall be present.
performanceMetricGroup	M	0..N	String	Group of performance metrics. A metric group is a pre-defined list of metrics, known to the producer that it can decompose to individual metrics. At least one of the two attributes (performance metric or metricGroup) shall be present.
collectionPeriod	M	1	Not specified	Specifies the periodicity at which the NFVO will collect performance information. See note.

Parameter	Qualifier	Cardinality	Content	Description
reportingPeriod	M	1	Not specified	Specifies the periodicity at which the NFVO will report to the OSS/BSS about performance information. See note.
reportingBoundary	O	0..1	Not specified	Identifies a boundary after which the reporting will stop. The boundary shall allow a single reporting as well as periodic reporting up to the boundary.
NOTE: At the end of each reportingPeriod, the NFVO informs OSS/BSS about availability of the performance data collected for each completed collection period during this reportingPeriod. While the exact definition of the types for collectionPeriod and reportingPeriod is part of the protocol design, it is recommended that the reportingPeriod be equal to or a multiple of the collectionPeriod. In the latter case, the performance data for the collection periods within one reporting period would be reported together.				

7.5.2.3 Output parameters

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

Table 7.5.2.3-1: Create PM Job operation output parameters

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

7.5.2.4 Operation results

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

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

7.5.3 Delete PM Jobs operation

7.5.3.1 Description

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

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

Table 7.5.3.1-1: Delete PM Jobs operation

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

7.5.3.2 Input parameters

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

Table 7.5.3.2-1: Delete PM Jobs operation input parameters

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

7.5.3.3 Output parameters

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

Table 7.5.3.3-1: Delete PM Jobs operation output parameters

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

7.5.3.4 Operation results

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

7.5.4 Subscribe operation

7.5.4.1 Description

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

NOTE 1: Specification of the filtering mechanism is part of the protocol design.

NOTE 2: It is part of the protocol design whether subscribing is represented as a separate "Subscribe" operation or whether subscription-related information is managed as part of managing PM jobs and Thresholds.

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

Table 7.5.4.1-1: Subscribe operation

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

7.5.4.2 Input parameters

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

Table 7.5.4.2-1: Subscribe operation input parameters

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

7.5.4.3 Output parameters

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

Table 7.5.4.3-1: Subscribe operation output parameters

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

7.5.4.4 Operation results

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

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

7.5.5 Notify operation

7.5.5.1 Description

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

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

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

Table 7.5.5.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

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

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

7.5.6 Query PM Job operation

7.5.6.1 Description

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

This operation does not return performance reports.

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

Table 7.5.6.1-1: Query PM Job operation

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

7.5.6.2 Input parameters

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

Table 7.5.6.2-1: Query PM Job operation input parameters

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

7.5.6.3 Output parameters

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

Table 7.5.6.3-1: Query PM Job operation output parameters

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

7.5.6.4 Operation results

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

7.5.7 Create Threshold operation

7.5.7.1 Description

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

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

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

Table 7.5.7.1-1: Create Threshold operation

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

7.5.7.2 Input parameters

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

Table 7.5.7.2-1: Create Threshold operation input parameters

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

7.5.7.3 Output parameters

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

Table 7.5.7.3-1: Create Threshold operation output parameters

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

7.5.7.4 Operation results

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

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

7.5.8 Delete Thresholds operation

7.5.8.1 Description

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

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

Table 7.5.8.1-1: Delete Thresholds operation

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

7.5.8.2 Input parameters

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

Table 7.5.8.2-1: Delete Thresholds operation input parameters

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

7.5.8.3 Output parameters

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

Table 7.5.8.3-1: Delete Thresholds operation output parameters

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

7.5.8.4 Operation results

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

7.5.9 Query Threshold operation

7.5.9.1 Description

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

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

Table 7.5.9.1-1: QueryThreshold operation

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

7.5.9.2 Input parameters

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

Table 7.5.9.2-1: QueryThreshold operation input parameters

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

7.5.9.3 Output parameters

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

Table 7.5.9.3-1: QueryThreshold operation output parameters

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

7.5.9.4 Operation results

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

7.6 NS Fault Management interface

7.6.1 Description

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

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

The fault management interface shall support the following operations:

- Subscribe operation: Subscription of OSS/BSSs with the NFVO for the notifications related to the alarms.
- Notify operation: Notifications of alarms or alarm state change from NFVO to OSS/BSS.
- Get alarm list operation: Accessing active alarms from the NFVO.
- Terminate Subscription operation: Terminating a particular subscription in the NFVO.
- Query Subscription Info operation: Querying subscription information from the NFVO.
- Acknowledge Alarms operation: Acknowledging alarms by the OSS/BSS.

7.6.2 Subscribe operation

7.6.2.1 Description

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

NOTE: Specification of the filtering mechanism is part of the protocol design.

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

Table 7.6.2.1-1: Subscribe operation

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

7.6.2.2 Input parameters

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

Table 7.6.2.2-1: Subscribe operation input parameters

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

7.6.2.3 Output parameters

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

Table 7.6.2.3-1: Subscribe operation output parameters

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

7.6.2.4 Operation results

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

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

7.6.3 Notify operation

7.6.3.1 Description

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

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

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

Table 7.6.3.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

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

- AlarmNotification. See clause 8.5.2.
- AlarmClearedNotification. See clause 8.5.3.
- AlarmListRebuiltNotification. See clause 8.5.5.

7.6.4 Get Alarm List operation

7.6.4.1 Description

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

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

Table 7.6.4.1-1: Get Alarm List operation

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

7.6.4.2 Input parameters

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

Table 7.6.4.2-1: Get Alarm List operation input parameters

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

7.6.4.3 Output parameters

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

Table 7.6.4.3-1: Get Alarm List operation output parameters

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

7.6.4.4 Operation results

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

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

7.6.5 Terminate Subscription operation

7.6.5.1 Description

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

Table 7.6.5.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.6.5.1-1: Terminate Subscription operation

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

7.6.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.6.5.2-1.

Table 7.6.5.2-1: Terminate Subscription operation input parameters

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

7.6.5.3 Output parameters

No output parameter.

7.6.5.4 Operation results

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

7.6.6 Query Subscription Info operation

7.6.6.1 Description

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

Table 7.6.6.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.6.6.1-1: Query Subscription operation

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

7.6.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.6.6.2-1.

Table 7.6.6.2-1: Query Subscription Info operation input parameters

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

7.6.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.6.6.3-1.

Table 7.6.6.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query.

7.6.6.4 Operation results

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

7.6.7 Acknowledge Alarms operation

7.6.7.1 Description

This operation enables the OSS/BSS to acknowledge alarms at NFVO.

Table 7.6.7.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.6.7.1-1: Acknowledge alarms operation

Message	Requirement	Direction
AcknowledgeAlarmsRequest	Mandatory	OSS/BSS → NFVO
AcknowledgeAlarmsResponse	Mandatory	NFVO → OSS/BSS

7.6.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.6.7.2-1.

Table 7.6.7.2-1: Acknowledge alarms operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
alarmId	M	1..N	Identifier (Reference to Alarm)	Identifier of an individual alarm to be acknowledged, or multiple identifiers of the alarms to be acknowledged. See note.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to acknowledge multiple alarms in one request, or as a series of requests that acknowledge one alarm at a time.				

7.6.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.6.7.3-1.

Table 7.6.7.3-1: Acknowledge alarms operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
acknowledgedAlarmId	M	1..N	Identifier (Reference to Alarm)	Identifier of an individual alarm that is acknowledged, or multiple identifiers of the alarms that are acknowledged. See note.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to acknowledge multiple alarms in one request, or as a series of requests that acknowledge one alarm at a time.				

7.6.7.4 Operation results

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

7.7 VNF Package management interface

7.7.1 Description

This interface allows for the management of VNF Packages.

The following operations are defined for this interface:

- Create VNF Package Info.
- Upload VNF Package.
- Update VNF Package Info.
- Delete VNF Package.
- Query VNF Package Info.
- Fetch VNF Package.
- Fetch VNF Package Artifacts.
- Subscribe to new notifications.
- Notify of on-boarding of new VNF Package or of changes of VNF Packages.
- Terminate Subscription.
- Query Subscription Info.

In the present document, the on-boarding of a VNF Package includes:

1) Creating a VNF Package information object.
2) Uploading the VNF Package.
3) Processing, including validation, the VNF Package inside the NFVO.

The VNF Package is referred as "on-boarded" only after these three procedures are successfully accomplished.

7.7.2 Upload VNF Package operation

7.7.2.1 Description

This operation will upload a VNF Package to the NFVO. A VNF Package information object shall be created a priori via the Create VNF Package Info operation. Only one VNF Package is allowed per VNF Package information object.

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

Table 7.7.2.1-1: Upload VNF Package operation

Message	Requirement	Direction
UploadVnfPackageRequest	Mandatory	OSS/BSS → NFVO
UploadVnfPackageResponse	Mandatory	NFVO → OSS/BSS

7.7.2.2 Input parameters

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

Table 7.7.2.2-1: Upload VNF Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfPkgInfoId	M	1	Identifier	Identifier of the VNF Package information object associated with the VNF Package to be uploaded.
vnfPackage	M	0..1	Binary	VNF Package to be uploaded. This attribute shall be supported when the VNF package is uploaded as a local file. See notes 2 and 3.
vnfPackagePath	M	0..1	URL	Address information based on which the VNF Package can be obtained. See note 1. This attribute shall be supported when the VNF package is uploaded from a remote server. See notes 2 and 3.
NOTE 1: This Structure can be the address information related to an FTP server when the VNF Package is stored, or be a URL where the NFVO can download the VNF Package. NOTE 2: Either the parameter vnfPackage or the parameter vnfPackagePath, but not both shall be provided. NOTE 3: For the onboarding procedure, the VNF Package contents may be split into more than one file, e.g. software image files separate from the VNF Package file containing the VNFD and other artifacts. The mechanism to support the upload of the one or more files comprising the VNF Package is part of the protocol design.				

7.7.2.3 Output parameters

No output parameter.

7.7.2.4 Operation results

The result of the operation indicates if the uploading and processing, including validation, of the VNF Package has been successful or not with a standard success/error result.

After a successful result, the VNF Package is known to and validated by the NFVO. The associated VNF Package information object is updated with the information populated from the validated VNF Package. The VNF Package is on-boarded and in "ENABLED,NOT_IN_USE" state, allowing its use for VNF lifecycle management. See the VNF Package state model in clause D.1.

7.7.3 Void

7.7.4 Void

7.7.5 Delete VNF Package operation

7.7.5.1 Description

This operation will delete a VNF Package. The associated VNF Package information objects will be deleted as well.

Note that a VNF Package can only be deleted once there are no VNFs using it and there are no NSDs referencing to it and the operational state is DISABLED.

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

Table 7.7.5.1-1: Delete VNF Package operation

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

7.7.5.2 Input parameters

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

Table 7.7.5.2-1: Delete VNF Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
VnfPkgInfoId	M	1	Identifier	Identifier of the VNF Package information object and associated VNF Package, which is to be deleted. This identifier was allocated by the NFVO.

7.7.5.3 Output parameters

No output parameter.

7.7.5.4 Operation results

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

7.7.6 Query VNF Package Info operation

7.7.6.1 Description

This operation will enable the OSS/BSS to query from the NFVO for details of one or more VNF Package information objects.

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

Table 7.7.6.1-1: Query VNF Package Info operation

Message	Requirement	Direction
QueryVnfPkgInfoRequest	Mandatory	OSS/BSS → NFVO
QueryVnfPkgInfoResponse	Mandatory	NFVO → OSS/BSS

7.7.6.2 Input parameters

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

Table 7.7.6.2-1: Query VNF Package Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filter defining the VNF Packages on which the query applies, based on attributes of VnfPkgInfo. It can also be used to specify one or more VNF Package information objects to be queried by providing their vnfdId or vnfPkgInfoId. See note.
attributeSelector	M	0..N	String	It provides a list of attribute names of VnfPkgInfo. If present, only these attributes are returned for VnfPkgInfo matching the filter. If absent, the complete VnfPkgInfo are returned.

Parameter	Qualifier	Cardinality	Content	Description
NOTE: The vnfdId, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2. The vnfPkgInfoId identifies the information related to the onboarding of a VNF package into the NFVO, which implies that it also identifies an onboarded VNF package.				

7.7.6.3 Output parameters

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

Table 7.7.6.3-1: Query VNF Package Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	VnfPkgInfo	Details of the VNF Package information objects matching the input filter. If attributeSelector is present, only the attributes listed in attributeSelector are returned for the selected entities.

7.7.6.4 Operation results

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

7.7.7 Subscribe operation

7.7.7.1 Description

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

NOTE: Specification of the filtering mechanism is part of the protocol design.

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

Table 7.7.7.1-1: Subscribe operation

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

7.7.7.2 Input parameters

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

Table 7.7.7.2-1: Subscribe operation input parameters

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

7.7.7.3 Output parameters

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

Table 7.7.7.3-1: Subscribe operation output parameters

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

7.7.7.4 Operation results

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

7.7.8 Notify operation

7.7.8.1 Description

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

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

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

Table 7.7.8.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

The following notification is sent by this operation:

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

7.7.9 Void

7.7.10 Fetch VNF Package operation

7.7.10.1 Description

This operation enables the OSS to fetch a whole VNF Package from the NFVO. The package is addressed using an identifier of the VNF Package information object associated with the VNF Package to be fetched. This identifier is contained within the VnfPackageOnBoardingNotification or is returned as a result of the Create VNF Package Info or Query VNF Package Info operations.

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

Table 7.7.10.1-1: Fetch VNF Package operation

Message	Requirement	Direction
FetchVnfPackageRequest	Mandatory	OSS → NFVO
FetchVnfPackageResponse	Mandatory	NFVO → OSS

7.7.10.2 Input parameters

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

Table 7.7.10.2-1: Fetch VNF Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
VnfPkgInfoId	M	1	Identifier	Identifier of the VNF Package information object associated with the VNF Package to be fetched. This identifier was allocated by the NFVO.

7.7.10.3 Output parameters

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

Table 7.7.10.3-1: Fetch VNF Package operation output parameters

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

7.7.10.4 Operation results

The result of the operation indicates whether the fetching of the VNF Package has been successful or not in the NFVO with a standard success/error result. After successful operation, the NFVO has provided to the OSS a copy of the requested VNF package.

7.7.11 Fetch VNF Package Artifacts operation

7.7.11.1 Description

This operation enables the OSS/BSS to fetch selected artifacts contained in a VNF package. Artifacts are addressed using selector information that can be obtained using the Query VNF Package Info operation.

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

Table 7.7.11.1-1: Fetch VNF PackageArtifacts operation

Message	Requirement	Direction
FetchPackageArtifactsRequest	Mandatory	OSS → NFVO
FetchVnfPackageArtifactsResponse	Mandatory	NFVO → OSS

7.7.11.2 Input parameters

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

Table 7.7.11.2-1: Fetch VNF Package Artifacts operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
VnfPkgInfoId	M	1	Identifier	Identifier of VNF Package information object associated with the VNF Package artifacts to be fetched. This identifier was allocated by the NFVO.
artifactSelector	M	1..N		Selector to address an individual VNF package artifact, or list of selectors to address multiple of those. See note.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to obtain multiple artifacts in one go, or as a series of operations that obtain one artifact at a time.				

7.7.11.3 Output parameters

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

Table 7.7.11.3-1: Fetch VNF Package Artifacts operation output parameters

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

7.7.11.4 Operation results

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

7.7.12 Void

7.7.13 Terminate Subscription operation

7.7.13.1 Description

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

Table 7.7.13.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.13.1-1: Terminate Subscription operation

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

7.7.13.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.13.2-1.

Table 7.7.13.2-1: Terminate Subscription operation input parameters

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

7.7.13.3 Output parameters

No output parameter.

7.7.13.4 Operation results

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

7.7.14 Query Subscription Info operation

7.7.14.1 Description

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

Table 7.7.14.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.14.1-1: Query Subscription operation

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

7.7.14.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.14.2-1.

Table 7.7.14.2-1: Query Subscription Info operation input parameters

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

7.7.14.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.7.14.3-1.

Table 7.7.14.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query.

7.7.14.4 Operation results

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

7.7.15 Create VNF Package Info operation

7.7.15.1 Description

This operation enables the OSS/BSS to create a VNF Package information object in the NFVO for the VNF Package to be uploaded.

Table 7.7.15.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.15.1-1: Create VNF Package Info operation

Message	Requirement	Direction
CreateVnfPackageInfoRequest	Mandatory	OSS/BSS → NFVO
CreateVnfPackageInfoResponse	Mandatory	NFVO → OSS/BSS

7.7.15.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.15.2-1.

Table 7.7.15.2-1: Create VNF Package Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
userDefinedData	O	0..N	KeyValuePair	User defined data for the VNF package to be uploaded.

7.7.15.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.7.15.3-1.

Table 7.7.15.3-1: Create VNF Package Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfPkgInfoId	M	1	Identifier	Identifier of the created VNF Package information object.

7.7.15.4 Operation results

The result of the operation indicates whether the creation of VNF Package information object has been successful or not with a standard success/error result.

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

7.7.16 Update VNF Package Info operation

7.7.16.1 Description

This operation enables the OSS/BSS to update the user defined data and/or the operational state of an existing VNF Package information object. The usage state shall not change as a result of the operation.

Table 7.7.16.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.7.16.1-1: Update VNF Package Info operation

Message	Requirement	Direction
UpdateVnfPackageInfoRequest	Mandatory	OSS/BSS → NFVO
UpdateVnfPackageInfoResponse	Mandatory	NFVO → OSS/BSS

7.7.16.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.16.2-1.

Table 7.7.16.2-1: Update VNF Package Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfPkgInfoId	M	1	Identifier	Identifier of the VNF Package information object to be updated.
operationalState	M	0..1	Enum	Operational state of the on-boarded VNF Package. VALUES: • ENABLED • DISABLED See note.
userDefinedData	O	0..N	KeyValuePair	User defined data to be updated. For existing keys, the value is replaced. See note.

Parameter	Qualifier	Cardinality	Content	Description
NOTE: At least one of the two parameters (operationalState and userDefinedData) shall be present. If VNF Package is not on-boarded, the operation is used only to update existing or add additional user defined data using the userDefinedData parameter.				

7.7.16.3 Output parameters

No output parameter.

7.7.16.4 Operation results

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

7.8 NFVI Capacity Information interface

7.8.1 Description

This interface allows providing of NFVI capacity information (including capacity shortage notifications) to the OSS/BSS.

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

- Query NFVI capacity operation.
- Subscribe operation.
- Terminate Subscription operation.
- Query Subscription Info operation.
- Notify operation.
- Create capacity threshold operation.
- Query capacity threshold operation.
- Delete capacity threshold operation.

The interface also provides the capabilities to create, query and delete capacity thresholds, which are used to set thresholds on specified NFVI capacity metrics. On the definition of a capacity threshold, a value for a given NFVI capacity metric is provided. When a threshold defined for a specific metric is crossed in up or down direction, an NFVI capacity information notification shall be raised to the consumer.

7.8.2 Query NFVI capacity operation

7.8.2.1 Description

This operation will enable the OSS/BSS to solicit from the VIM via the NFVO information about the available, reserved, allocated/used and total NFVI capacity.

NOTE: Specification of the filtering mechanism is part of the protocol design.

Table 7.8.2.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.8.2.1-1: Query NFVI capacity operation

Message	Requirement	Direction
QueryNfviCapacityRequest	Mandatory	OSS/BSS → NFVO
QueryNfviCapacityResponse	Mandatory	NFVO → OSS/BSS

7.8.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.2.2-1.

Table 7.8.2.2-1: Query NFVI capacity operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting information to query. For instance, based on the resource zone, VIM, time interval for which capacity is queried, and which capacity information (i.e. available, total, reserved and/or allocated/used capacity) is queried.

7.8.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.2.3-1.

Table 7.8.2.3-1: Query NFVI capacity operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
capacityResponse	M	0..1	Not specified	The capacity matching the query. Cardinality is 0 if no data is matching the input filter.

7.8.2.4 Operation results

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

7.8.3 Subscribe operation

7.8.3.1 Description

This operation enables the OSS/BSSs to subscribe with a filter for the notifications related to NFVI capacity shortage with the NFVO.

NOTE 1: Specification of the filtering mechanism and how to update the input parameters is part of the protocol design.

NOTE 2: It is part of the protocol design whether subscribing is represented as a separate "Subscribe" operation or whether subscription-related information is managed as part of managing Capacity Thresholds.

Table 7.8.3.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.8.3.1-1: Subscribe operation

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

7.8.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.3.2-1.

Table 7.8.3.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting notifications. The filter shall support: • Resource type(s): specifies the type(s) of resources (virtual compute, virtual network, virtual storage, and/or compute hosts) for which notifications shall be triggered. • Resource zone(s): specifies the resource zone(s) for which notifications shall be triggered. If no resource zone is provided, notifications for the whole VIM or NFVI shall be provided. • VIM(s): specifies the VIM(s) for which notifications shall be triggered. If no VIM is provided, notifications for the whole NFVI shall be provided.

7.8.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.3.3-1.

Table 7.8.3.3-1: Subscribe operation output parameters

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

7.8.3.4 Operation results

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

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

7.8.4 Terminate Subscription operation

7.8.4.1 Description

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

NOTE: It is part of the protocol design whether terminating a subscribing is represented as a separate "Terminate Subscription" operation or whether subscription-related information is managed as part of managing Capacity Thresholds. Table 7.8.4.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.8.4.1-1: Terminate Subscription operation

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

7.8.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.4.2-1.

Table 7.8.4.2-1: Terminate Subscription operation input parameters

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

7.8.4.3 Output parameters

No output parameter.

7.8.4.4 Operation results

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

7.8.5 Query Subscription Info operation

7.8.5.1 Description

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

NOTE: It is part of the protocol design whether querying information about subscriptions is represented as a separate "Query Subscription Info" operation or whether subscription-related information is managed as part of managing Capacity Thresholds.

Table 7.8.5.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.8.5.1-1: Query Subscription operation

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

7.8.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.5.2-1.

Table 7.8.5.2-1: Query Subscription Info operation input parameters

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

7.8.5.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.5.3-1.

Table 7.8.5.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query.

7.8.5.4 Operation results

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

7.8.6 Notify operation

7.8.6.1 Description

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

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

Table 7.8.6.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.8.6.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

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

- CapacityThresholdCrossedNotification. See clause 8.7.2.

7.8.7 Create Capacity Threshold operation

7.8.7.1 Description

This operation enables the OSS/BSS to create on the NFVO a threshold and specify threshold levels on a specified NFVI capacity metric. Notifications will be generated when crossed in up or down direction.

The OSS/BSS needs to be subscribed to receive CapacityThresholdCrossedNotification notifications.

Table 7.8.7.1-1 lists the information flow exchange between the OSS/BSS and the NFVO.

Table 7.8.7.1-1: Create Capacity Threshold operation

Message	Requirement	Direction
CreateCapacityThresholdRequest	Mandatory	OSS/BSS → NFVO
CreateCapacityThresholdResponse	Mandatory	NFVO → OSS/BSS

7.8.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.7.2-1.

Table 7.8.7.2-1: Create Capacity Threshold operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
thresholdType	M	1	Enum	Defines the type of threshold. The list of possible values is part of the protocol design and might include: single/multi valued threshold, static/dynamic threshold, template based threshold, etc. VALUES: • SIMPLE: Single-valued static threshold • etc.
thresholdDetails	M	1	Not specified.	Details of the threshold: value to be crossed, direction in which it is crossed, details on the notification to be generated, etc.

7.8.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.7.3-1.

Table 7.8.7.3-1: Create Capacity Threshold operation output parameters

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

7.8.7.4 Operation results

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

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

7.8.8 Delete Capacity Thresholds operation

7.8.8.1 Description

This operation enables the OSS/BSS to delete one or more existing NFVI capacity threshold(s) on the NFVO.

NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to delete multiple thresholds in one request, or as a series of requests that delete one threshold at a time.

Table 7.8.8.1-1 lists the information flow exchange between the OSS/BSS and the NFVO.

Table 7.8.8.1-1: Delete Capacity Thresholds operation

Message	Requirement	Direction
DeleteCapacityThresholdsRequest	Mandatory	OSS/BSS → NFVO
DeleteCapacityThresholdsResponse	Mandatory	NFVO → OSS/BSS

7.8.8.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.8.2-1.

Table 7.8.8.2-1: Delete Capacity Thresholds operation input parameters

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

7.8.8.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.8.3-1.

Table 7.8.8.3-1: Delete Capacity Thresholds operation output parameters

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

7.8.8.4 Operation results

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

7.8.9 Query Capacity Threshold operation

7.8.9.1 Description

This operation enables the OSS/BSS to query the details of one or more existing NFVI capacity thresholds on the NFVO.

Table 7.8.9.1-1 lists the information flow exchange between the OSS/BSS and the NFVO.

Table 7.8.9.1-1: Query Capacity Threshold operation

Message	Requirement	Direction
QueryCapacityThresholdRequest	Mandatory	OSS/BSS → NFVO
QueryCapacityThresholdResponse	Mandatory	NFVO → OSS/BSS

7.8.9.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.9.2-1.

Table 7.8.9.2-1: Query Capacity Threshold operation input parameters

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

7.8.9.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.9.3-1.

Table 7.8.9.3-1: Query Capacity Threshold operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
thresholdDetails	M	0..N	NfviCapacity Threshold	Details of thresholds matching the input filter.

7.8.9.4 Operation results

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

7.9 Policy Management interface

7.9.1 Description

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

The following policy management operations are defined for this interface:

- Transfer Policy.
- Delete Policy.
- Query Policy.
- Activate Policy.
- Deactivate Policy.

- Associate Policy.
- Disassociate Policy.

This interface allows the OSS/BSS to manage subscriptions to notifications sent by the NFVO which inform about changes of a policy and about any detected policy conflicts. It allows the NFVO to provide such notifications to the subscriber (i.e. OSS/BSS).

7.9.2 Transfer Policy operation

7.9.2.1 Description

This operation enables the OSS/BSS to transfer a NFV-MANO policy to the NFVO. Table 7.9.2.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.2.1-1: Transfer Policy operation

Message	Requirement	Direction
TransferPolicyRequest	Mandatory	OSS/BSS → NFVO
TransferPolicyResponse	Mandatory	NFVO → OSS/BSS

7.9.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.2.2-1.

Table 7.9.2.2-1: Transfer Policy operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
designer	M	1	String	Human readable name of designer of the policy.
name	M	1	String	Human readable name of the policy.
version	M	1	Version	Version of the policy.
pfId	M	0..1	Identifier	Identifier of the policy function (PF) which enforces the policy. The PF is either a VNFM or a VIM. Cardinality of zero indicates that the PF is the NFVO itself. See note 3.
policy	M	1	Not specified	Specifies the policy. See notes 1 and 2.
NOTE 1: An identifier for uniquely identifying the policy is included in the policy. NOTE 2: The OSS/BSS may use this operation to update an existing policy with a new version. Different policy versions share the same internal identifier of the policy but having different PolicyInfo instances. The design of different policy versions and their business logic is out of the scope of the present document. NOTE 3: The NFVO uses the pfId attribute to identify the policy function enforcing the policy. More information concerning the policy function can be found in ETSI GR NFV-IFA 023 [i.9].				

7.9.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.9.2.3-1.

Table 7.9.2.3-1: Transfer Policy operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
policyInfoId	M	1	Identifier	Identifier of the policy information created by the NFVO.

7.9.2.4 Operation results

In case of success, the NFV-MANO policy is transferred to the NFVO and corresponding policy information is created by the NFVO. In case of failure, appropriate error information is returned.

7.9.3 Delete Policy operation

7.9.3.1 Description

This operation enables the OSS/BSS to delete one or multiple NFV-MANO policy(ies) from the NFVO. Table 7.9.3.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.3.1-1: Delete Policy operation

Message	Requirement	Direction
DeletePolicyRequest	Mandatory	OSS/BSS → NFVO
DeletePolicyResponse	Mandatory	NFVO → OSS/BSS

7.9.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.3.2-1.

Table 7.9.3.2-1: Delete Policy operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
policyInfoId	M	1..N	Identifier (Reference to PolicyInfo)	Identifier(s) of policy information.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to delete multiple policies in one request, or as a series of requests that delete one policy at a time.				

7.9.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.9.3.3-1.

Table 7.9.3.3-1: Delete Policy operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deletedPolicyInfoId	M	0..N	Identifier (Reference to PolicyInfo)	Identifier(s) of the deleted NFV-MANO policy information.

7.9.3.4 Operation results

In case of success, the NFV-MANO policy(ies) is (are) deleted from the NFVO, and a success indicator is returned to the OSS/BSS. In case of failure, appropriate error information is returned.

7.9.4 Query Policy operation

7.9.4.1 Description

This operation enables the OSS/BSS to query the information from the NFVO on one or multiple NFV-MANO policy(ies). Table 7.9.4.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.4.1-1: Query Policy operation

Message	Requirement	Direction
QueryPolicyRequest	Mandatory	OSS/BSS → NFVO
QueryPolicyResponse	Mandatory	NFVO → OSS/BSS

7.9.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.4.2-1.

Table 7.9.4.2-1: Query Policy operation input parameters

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

7.9.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.9.4.3-1.

Table 7.9.4.3-1: Query Policy operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryNsPolicyInfoResult	M	0..N	PolicyInfo	NFV-MANO policy information matching the input filter. If attributeSelector is present, only the attributes listed in attributeSelector are returned for the selected policy information. See note.
NOTE: The lower cardinality is 0 since there may be no matches to the provided filter.				

7.9.4.4 Operation results

After success operation, the NFVO has queried the internal NFV-MANO policy information. The result of the operation indicates whether it has been successful or not with a standard success/error result. For a particular query, policy information that is matching the filter shall be returned.

7.9.5 Activate Policy operation

7.9.5.1 Description

This operation enables the OSS/BSS to activate one or multiple NFV-MANO policy(ies) in the NFVO. Table 7.9.5.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.5.1-1: Activate Policy operation

Message	Requirement	Direction
ActivatePolicyRequest	Mandatory	OSS/BSS → NFVO
ActivatePolicyResponse	Mandatory	NFVO → OSS/BSS

7.9.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.5.2-1.

Table 7.9.5.2-1: Activate Policy operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
policyInfoId	M	1..N	Identifier (Reference to PolicyInfo)	Identifier(s) of policy information. See note.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to activate multiple policies in one request, or as a series of requests that activate one policy at a time.				

7.9.5.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.9.5.3-1.

Table 7.9.5.3-1: Activate Policy operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
activatedPolicyInfoId	M	0..N	Identifier (Reference to PolicyInfo)	Identifier(s) of the activated NFV-MANO policy(ies).

7.9.5.4 Operation results

In case of success, the NFV-MANO policy(ies) are activated in the NFVO, and a success indicator is returned to the OSS/BSS. In case of failure, appropriate error information is returned.

7.9.6 Deactivate Policy operation

7.9.6.1 Description

This operation enables the OSS/BSS to deactivate one or multiple NFV-MANO policy(ies) in the NFVO.
Table 7.9.6.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.6.1-1: Deactivate Policy operation

Message	Requirement	Direction
DeactivatePolicyRequest	Mandatory	OSS/BSS → NFVO
DeactivatePolicyResponse	Mandatory	NFVO → OSS/BSS

7.9.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.6.2-1.

Table 7.9.6.2-1: Deactivate Policy operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
policyInfoId	M	1..N	Identifier (Reference to PolicyInfo)	Identifier(s) of policy information. See note.
NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to deactivate multiple policies in one request, or as a series of requests that deactivate one policy at a time.				

7.9.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.9.6.3-1.

Table 7.9.6.3-1: Deactivate Policy operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
deactivatedPolicyInfoId	M	0..N	Identifier (Reference to PolicyInfo)	Identifier(s) of the deactivated NFV-MANO policy(ies).

7.9.6.4 Operation results

In case of success, the NFV-MANO policy(ies) are deactivated in the NFVO, and a success indicator is returned to the OSS/BSS. In case of failure, appropriate error information is returned.

7.9.7 Subscribe operation

7.9.7.1 Description

This operation enables the OSS/BSS to subscribe with a filter for the notifications sent by the NFVO which are related to changes of a policy and any detected policy conflicts. Changes of a policy are related to operations of transferring policy, deleting policy, activating policy, deactivating policy, associate policy and disassociate policy.

Table 7.9.7.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.7.1-1: Subscribe operation

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

7.9.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.7.2-1.

Table 7.9.7.2-1: Subscribe operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Input filter for selecting the notifications. This filter can contain information about specific types of notifications to subscribe to, or attributes of the PolicyInfo. Details are part of the protocol design.

7.9.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.9.7.3-1.

Table 7.9.7.3-1: Subscribe operation output parameters

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

7.9.7.4 Operation results

After successful subscription, the consumer (OSS/BSS) is registered to receive notifications about events related to changes of a policy and any detected policy conflicts.

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

7.9.8 Notify operation

7.9.8.1 Description

This operation notifies a subscriber about events related to notifications about changes of a policy and any detected policy conflicts.

This operation distributes notifications to subscribers. It is a one-way operation issued by the producer (NFVO) that cannot be invoked as an operation by the consumer (OSS/BSS). In order to receive notifications, the consumer (OSS/BSS) has to perform an explicit Subscribe operation beforehand.

Table 7.9.8.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.8.1-1: Notify operation

Message	Requirement	Direction
Notify	Mandatory	NFVO → OSS/BSS

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

- PolicyChangeNotification. See clause 8.8.3.
- PolicyConflictNotification. See clause 8.8.4.

7.9.9 Terminate Subscription operation

7.9.9.1 Description

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

Table 7.9.9.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.9.1-1: Terminate Subscription operation

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

7.9.9.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.9.2-1.

Table 7.9.9.2-1: Terminate Subscription operation input parameters

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

7.9.9.3 Output parameters

No output parameter.

7.9.9.4 Operation results

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

7.9.10 Query Subscription Info operation

7.9.10.1 Description

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

Table 7.9.10.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.10.1-1: Query Subscription Info operation

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

7.9.10.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.10.2-1.

Table 7.9.10.2-1: Query Subscription Info operation input parameters

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

7.9.10.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.9.10.3-1.

Table 7.9.10.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	Not specified	Information about the subscription(s) matching the query.

7.9.10.4 Operation results

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

7.9.11 Associate Policy operation

7.9.11.1 Description

This operation enables the OSS/BSS to associate an NFV-MANO policy to one or multiple NS instances in the NFVO.

Table 7.9.11.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.11.1-1: Associate Policy operation

Message	Requirement	Direction
AssociatePolicyRequest	Mandatory	OSS/BSS → NFVO
AssociatePolicyResponse	Mandatory	NFVO → OSS/BSS

7.9.11.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.11.2-1.

Table 7.9.11.2-1: Associate Policy operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
policyInfoId	M	1	Identifier (Reference to PolicyInfo	Identifier of policy information.
nsInstanceId	M	1..N	Identifier	Identifier(s) of the NS instance(s) to associate policy to. See note.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to associate a policy to multiple NS instances in one request, or as a series of requests that associate the policy to one NS instance at a time.				

7.9.11.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.9.11.3-1.

Table 7.9.11.3-1: Associate Policy operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	0..N	Identifier	Identifier(s) of the NS instance(s) to which the policy has been associated.

7.9.11.4 Operation results

After successful operation, the NFVO has associated the MANO policy to the NS instance(s), and a success indicator is returned to the OSS/BSS. In case of failure, appropriate error information is returned.

7.9.12 Disassociate Policy operation

7.9.12.1 Description

This operation enables the OSS/BSS to disassociate a MANO policy from one or multiple NS instances in the NFVO.

Table 7.9.12.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.9.12.1-1: Disassociate Policy operation

Message	Requirement	Direction
DisassociatePolicyRequest	Mandatory	OSS/BSS → NFVO
DisassociatePolicyResponse	Mandatory	NFVO → OSS/BSS

7.9.12.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.9.12.2-1.

Table 7.9.12.2-1: Query Subscription Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
policyInfoId	M	1	Identifier (Reference to PolicyInfo	Identifier of policy information.
nsInstanceId	M	1..N	Identifier	Identifier(s) of the NS instance(s) to disassociate policy from. See note.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to disassociate a policy from multiple NS instances in one request, or as a series of requests that disassociate the policy from one NS instance at a time.				

7.9.12.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.9.12.3-1.

Table 7.9.12.3-1: Query Subscription Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
nsInstanceId	M	0..N	Identifier	Identifier(s) of the NS instance(s) from which the policy has been disassociated.

7.9.12.4 Operation results

After successful operation, the NFVO has disassociated the MANO policy from the NS instance(s), and a success indicator is returned to the OSS/BSS. In case of failure, appropriate error information is returned.

7.10 VNF Snapshot Package Management interface

7.10.1 Description

This interface allows the OSS/BSS to access the VNF Snapshot Package information and to fetch, create, upload, extract, delete VNF Snapshot packages. The create VNF Snapshot Package operation is designed as a 2-step operation, whereby first a VNF Snapshot Package information object is created based on the VNF Snapshot metadata. Then, second, either a new VNF Snapshot Package is built or an existing VNF Snapshot Package is uploaded.

7.10.2 Create VNF Snapshot Package Info operation

7.10.2.1 Description

This operation enables the OSS/BSS to request to the NFVO the creation of a VNF Snapshot Package identifier and associated instance of a VNF Snapshot Package information element, identified by that identifier.

Table 7.10.2.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.10.2.1-1: Create VNF Snapshot Package Info operation

Message	Requirement	Direction
CreateVnfSnapshotPackageInfoRequest	Mandatory	OSS/BSS → NFVO
CreateVnfSnapshotPackageInfoResponse	Mandatory	NFVO → OSS/BSS

7.10.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.10.2.2-1.

Table 7.10.2.2-1: Create VNF Snapshot Package Info operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
name	M	1	String	Human-readable name of the VNF Snapshot Package.
userDefinedData	O	0..N	KeyValuePair	User defined data for the VNF Package to be created.

7.10.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.10.2.3-1.

Table 7.10.2.3-1: Create VNF Snapshot Package Info operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfSnapshotPkgInfoId	M	1	Identifier (Reference to VnfSnapshotPkgInfo)	References the information held by the NFVO about the created VNF Snapshot Package.

7.10.2.4 Operation results

The result of the operation indicates if the creation of the VNF Snapshot Package information object has been successful or not with a standard success/error result.

After successful operation, the NFVO has created a VNF Snapshot Package information object and the state of this VnfSnapshotPkgInfo is "CREATED". Once created, the VNF Snapshot Package is known to the NFVO. It is enabled to be queried for its associated information and is enabled for building/uploading the VNF Snapshot package.

7.10.3 Build VNF Snapshot Package operation

7.10.3.1 Description

This operation enables the OSS/BSS to request to the NFVO to populate the VNF Snapshot Package information object with the information from the VnfSnapshotInfo, the VnfcSnapshotImageInfo object(s), and SnapshotPkgArtifactInformation object(s). The VNF Snapshot to be packaged is addressed using an identifier of information known to the NFVO about a specific VNF Snapshot.

Table 7.10.3.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.10.3.1-1: Build VNF Snapshot Package operation

Message	Requirement	Direction
BuildVnfSnapshotPackageRequest	Mandatory	OSS/BSS → NFVO
BuildVnfSnapshotPackageResponse	Mandatory	NFVO → OSS/BSS

7.10.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.10.3.2-1.

Table 7.10.3.2-1: Build VNF Snapshot Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfSnapshotPkgInfoId	M	1	Identifier (Reference to VnfSnapshotPkgInfo)	References the information held by the NFVO about the specific VNF Snapshot Package to which the VNF Snapshot image(s) are to be added.
vnfSnapshotInfoId	M	1	Identifier (Reference to VnfSnapshotInfo)	References the information about a specific VNF Snapshot to be added to the VNF Snapshot Package. This identifier was allocated by the VNFM and is assumed to be known to the NFVO.

7.10.3.3 Output parameters

No output parameter.

7.10.3.4 Operation results

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

After successful operation, the NFVO has populated the VNF Snapshot Package Info object with the information from the VnfSnapshotInfo, the VnfcSnapshotImageInfo object(s), and SnapshotPkgArtifactInformation object(s). It is enabled to be queried for its associated information, its content is enabled to be fetched, and it is enabled to be extracted.

The state of the VnfSnapshotPkgInfo is changed to "BUILDING" during the build operation, is changed to "PROCESSING" once the build is completed, and is changed to "AVAILABLE" once the validation is completed. Also, a globally unique vnfSnapshotPkgId is created that remains the same for the lifetime of the package.

7.10.4 Upload VNF Snapshot Package operation

7.10.4.1 Description

This operation enables the OSS/BSS to request to the NFVO to upload an external VNF Snapshot Package from an external location into the NFVO. A new VNF Snapshot Package information element shall be created a priori via the Create VNF Snapshot Package Info operation. Only one VNF Snapshot Package is allowed per VNF Snapshot Package information object.

NOTE: The NFVO may utilize remote storage capabilities to store the package.

Table 7.10.4.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.10.4.1-1: Upload VNF Snapshot Package operation

Message	Requirement	Direction
UploadVnfSnapshotPackageRequest	Mandatory	OSS/BSS → NFVO
UploadVnfSnapshotPackageResponse	Mandatory	NFVO → OSS/BSS

7.10.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.10.4.2-1.

Table 7.10.4.2-1: Upload VNF Snapshot Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfSnapshotPkgInfoId	M	1	Identifier (Reference to VnfSnapshotPkgInfo)	References the information held by the NFVO about the specific VNF Snapshot Package info which the external VNF Snapshot Package shall be uploaded to.
vnfSnapshotPkg	M	0..1	Binary	VNF Snapshot Package to be uploaded. This attribute shall be supported when the VNF Snapshot Package is uploaded as a local file. See note 2.
vnfSnapshotPkgPath	M	0..1	Not specified	Address information based on which the VNF Snapshot Package can be obtained. See note 1. This attribute shall be supported when the VNF Snapshot Package is uploaded from a remote server. See note 2.
NOTE 1: This structure can be the address information related to an FTP server where the VNF Snapshot Package is located, or be a URL where the NFVO can download the VNF Snapshot Package. NOTE 2: Either the parameter vnfSnapshotPkg or the parameter vnfSnapshotPkgPath, but not both shall be provided. NOTE 3: The vnfdId shall match the vnfdId in the VNF Snapshot Package (if this parameter exists inside the VNF Snapshot Package).				

7.10.4.3 Output parameters

No output parameter.

7.10.4.4 Operation results

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

After successful operation, the VNF Snapshot Package is known to and validated by the NFVO. The associated VNF Snapshot Package information object was updated with the information populated from the validated VNF Snapshot Package, e.g. the globally unique vnfSnapshotPkgId was obtained from the VNF Snapshot Package. It is enabled to be queried for its associated information, its content is enabled to be fetched, and it is enabled to be extracted.

The state of the VnfSnapshotPkgInfo is changed to "UPLOADING" during the upload operation, is changed to "PROCESSING" once the upload is completed, and is changed to "AVAILABLE" once the validation is completed.

7.10.5 Extract VNF Snapshot Package operation

7.10.5.1 Description

This operation enables the OSS/BSS to request to the NFVO the extraction of a VNF Snapshot Package, i.e. it requests the NFVO to extract the VNF Snapshot Package and store the included VNFC Snapshot information object(s) and VNF Snapshot information. The VNF Snapshot Package to be extracted is addressed using an identifier of information held by the NFVO about a specific VNF Snapshot Package.

Table 7.10.5.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.10.5.1-1: Extract VNF Snapshot Package operation

Message	Requirement	Direction
ExtractVnfSnapshotPackageRequest	Mandatory	OSS/BSS → NFVO
ExtractVnfSnapshotPackageResponse	Mandatory	NFVO → OSS/BSS

7.10.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.10.5.2-1.

Table 7.10.5.2-1: Extract VNF Snapshot Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfSnapshotPkgInfoId	M	1	Identifier (Reference to VnfSnapshotPkgInfo)	References the information held by the NFVO about a specific VNF Snapshot Package to be extracted.
vnfSnapshotInfoId	M	0..1	Identifier (Reference to VnfSnapshotInfo)	Identifier held by the NFVO about an "Individual VNF snapshot" managed by the VNFM to which the content of the VNF snapshot package will be extracted to. See note.
vnfInstanceId	M	0..1	Identifier (Reference to VnfInfo)	Identifier of the VNF instance to which the content and extraction of the VNF snapshot package is to be associated. See note.
NOTE: Either the parameter vnfSnapshotInfoId or vnfInstanceId, but not both, shall be provided.				

7.10.5.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.10.5.3-1.

Table 7.10.5.3-1: Extract VNF Snapshot Package operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfSnapshotInfoId	M	1	Identifier (Reference to VnfSnapshotInfo)	References the information held by the NFVO about the extracted VNF Snapshot.

7.10.5.4 Operation results

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

After successful operation, the NFVO has extracted a VNF Snapshot from the specified VNF Snapshot Package and has created information associated with this VNF Snapshot. Once extracted, the VNF Snapshot is known to the NFVO. It is enabled to be queried for its associated information, and it is enabled to be reverted to.

The state of the VnfSnapshotPkgInfo is changed to "EXTRACTING" during the extract operation and is changed to "AVAILABLE" once the extraction is completed.

7.10.6 Fetch VNF Snapshot Package operation

7.10.6.1 Description

This operation enables the OSS/BSS to fetch a whole VNF Snapshot Package from the NFVO. The package is addressed using an identifier of information held by the NFVO about the specific VNF Snapshot Package.

Table 7.10.6.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.10.6.1-1: Fetch VNF Snapshot Package operation

Message	Requirement	Direction
FetchVnfSnapshotPackageRequest	Mandatory	OSS/BSS → NFVO
FetchVnfSnapshotPackageResponse	Mandatory	NFVO → OSS/BSS

7.10.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.10.6.2-1.

Table 7.10.6.2-1: Fetch VNF Snapshot Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfSnapshotPkgInfoId	M	1	Identifier (Reference to VnfSnapshotPkgInfo)	References the information held by the NFVO about the VNF Snapshot Package to be fetched.

7.10.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.10.6.3-1.

Table 7.10.6.3-1: Fetch VNF Snapshot Package operation output parameters

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

7.10.6.4 Operation results

After successful operation, the NFVO has provided to the OSS/BSS a copy of the requested VNF Snapshot Package.

7.10.7 Fetch VNF Snapshot Package Artifacts operation

7.10.7.1 Description

This operation enables the OSS/BSS to fetch selected artifacts contained in an VNF Snapshot Package. Artifacts are addressed using selector information that can be obtained using the QueryVnfSnapshotPkgInfo operation.

Table 7.10.7.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.10.7.1-1: Fetch VNF Snapshot Package Artifacts operation

Message	Requirement	Direction
FetchVnfSnapshotPackageArtifactsRequest	Mandatory	OSS/BSS → NFVO
FetchVnfSnapshotPackageArtifactsResponse	Mandatory	NFVO → OSS/BSS

7.10.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.10.7.2-1.

Table 7.10.7.2-1: Fetch VNF Snapshot Package Artifacts operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfSnapshotPkgInfoId	M	1	Identifier (Reference to VnfSnapshotPkgInfo)	References the information held by the NFVO about the VNF Snapshot Package.
artifactSelector	M	1..N	Not specified	Selector to address an individual VNF Snapshot Package artifact, or list of selectors to address multiple of those. See note.
NOTE: It is part of the protocol design whether this operation is modelled as a "bulk" operation that allows to obtain multiple artifacts in one go, or as a series of operations that obtain one artifact at a time.				

7.10.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.10.7.3-1.

Table 7.10.7.3-1: Fetch VNF Snapshot Package Artifacts operation output parameters

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

7.10.7.4 Operation results

After successful operation, the NFVO has provided to the OSS/BSS a copy/copies of the requested artifact(s) contained in the VNF Snapshot Package.

7.10.8 Query VNF Snapshot Package Information operation

7.10.8.1 Description

When a VNF Snapshot Package information element and the VNF Snapshot Package is built by or uploaded to the NFVO, the NFVO creates and stores information associated with this VNF Snapshot Package. It maintains this information during the VNF Snapshot Package's operational lifecycle. This operation will enable the OSS/BSS to query the NFVO for information it has stored about one or more VNF Snapshot Packages. Table 7.10.8.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

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

NOTE: The vnfSnapshotInfoId is an attribute of the VnfSnapshotPkgInfo.

Table 7.10.8.1-1: Query VNF Snapshot Package Information operation

Message	Requirement	Direction
QueryVnfSnapshotPkgInfoRequest	Mandatory	OSS/BSS → NFVO
QueryVnfSnapshotPkgInfoResponse	Mandatory	NFVO → OSS/BSS

7.10.8.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.10.8.2-1.

Table 7.10.8.2-1: Query VNF Snapshot Package Information operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
filter	M	1	Filter	Filter defining the VNF Snapshot Packages on which the query applies, based on attributes of the VnfSnapshotPkgInfo. It can also be used to specify one or more VNF Snapshot Packages to be queried by providing their vnfSnapshotInfoId or vnfSnapshotPkgInfoId. See note.
attributeSelector	M	0..N	String	It provides a list of attribute names of VnfSnapshotPkgInfo. If present, only these attributes are returned for the VnfSnapshotPkgInfo matching the filter. If absent, the complete VnfSnapshotPkgInfo is returned.
NOTE: The vnfSnapshotInfoId, assigned by the VNFM at VNF Snapshot creation or at VNF Snapshot Package extraction, identifies the information related to a VNF Snapshot. It is assumed that this information is known to the NFVO. The vnfSnapshotPkgInfoId identifies the information related to the creation or storage of a VNF Snapshot Package in the NFVO, which implies that it also identifies an available VNF Snapshot Package.				

7.10.8.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.10.8.3-1.

Table 7.10.8.3-1: Query available VNF Snapshot Package Information operation output parameters

Parameter	Qualifier	Cardinality	Content	Description
queryResult	M	0..N	VnfSnapshotPkgInfo	Details of the VNF Snapshot Packages available to the NFVO matching the input filter. If attributeSelector is present, only the attributes listed in attributeSelector are returned for the selected entities. Cardinality is 0 if no data is matching the input filter.

7.10.8.4 Operation results

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

7.10.9 Delete VNF Snapshot Package operation

7.10.9.1 Description

This operation enables the OSS/BSS to request to the NFVO the deletion of a VNF Snapshot Package. The VNF Snapshot Package to be deleted is addressed using an identifier of information held by the NFVO about a specific VNF Snapshot Package.

Table 7.10.9.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.10.9.1-1: Delete VNF Snapshot Package operation

Message	Requirement	Direction
DeleteVnfSnapshotPackageRequest	Mandatory	OSS/BSS → NFVO
DeleteVnfSnapshotPackageResponse	Mandatory	NFVO → OSS/BSS

7.10.9.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.10.9.2-1.

Table 7.10.9.2-1: Delete VNF Snapshot Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfSnapshotPkgInfoId	M	1	Identifier (Reference to VnfSnapshotPkgInfo)	References the information held by the NFVO about a specific VNF Snapshot Package to be deleted.

7.10.9.3 Output parameters

No output parameter.

7.10.9.4 Operation results

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

After successful operation, the NFVO has deleted the referenced artifacts and the held information associated to the specified VNF Snapshot Package.

7.10.10 Update VNF Snapshot Package operation

7.10.10.1 Description

This operation enables the OSS/BSS to request to the NFVO the update of selected attributes of a VNF Snapshot Package. The VNF Snapshot Package to be updated is addressed using an identifier of information held by the NFVO about a specific VNF Snapshot Package.

Table 7.10.10.1-1 lists the information flow exchanged between the OSS/BSS and the NFVO.

Table 7.10.10.1-1: Update VNF Snapshot Package operation

Message	Requirement	Direction
UpdateVnfSnapshotPackageRequest	Mandatory	OSS/BSS → NFVO
UpdateVnfSnapshotPackageResponse	Mandatory	NFVO → OSS/BSS

7.10.10.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.10.10.2-1.

Table 7.10.10.2-1: Update VNF Snapshot Package operation input parameters

Parameter	Qualifier	Cardinality	Content	Description
vnfSnapshotPkgInfoId	M	1	Identifier (Reference to VnfSnapshotPkgInfo)	References the information held by the NFVO about a specific VNF Snapshot Package to be updated.
name	M	0..1	String	If present, specified the new value of the human-readable name of the VNF Snapshot Package. See note
userDefinedData	O	0..N	KeyValuePair	If present, specifies the user defined data for the VNF Snapshot Package to be updated. For existing keys, the value is replaced. See note.
NOTE: At least one of the two parameters name and userDefinedData shall be present. If the VNF snapshot package is not uploaded or built, the operation is used only to update existing or add additional user defined data using the userDefinedData attribute.				

7.10.10.3 Output parameters

No output parameter.

7.10.10.4 Operation results

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

After successful operation, the NFVO has updated the specified VNF Snapshot Package.

8 Information elements exchanged

8.1 Introduction

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

The specification of the following information elements is part of the protocol design:

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

8.2 Information elements related to NSD Management

8.2.1 Introduction

The clauses below define information elements related to NSD management.

8.2.2 NsdInfo information element

8.2.2.1 Description

This information element provides the details of an NsdInfo information element.

8.2.2.2 Attributes

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

Table 8.2.2.2-1: Attributes of the NsdInfo information element

Attribute	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier	Identifier of the NSD information object.
nsdId	M	0..1	Identifier	Identifier of the on-boarded NSD. See note 1.
name	M	0..1	String	Name of the on-boarded NSD. See note 1.
version	M	0..1	Version	Version of the on-boarded NSD. See note 1.
designer	M	0..1	String	Designer of the on-boarded NSD. See note 1.
nsd	M	0..1	Identifier (Reference to Nsd)	Reference to the on-boarded NSD details, e.g. URL to the on-boarded NSD. See note 1.
vnfPkgInfoId	M	0..N	Identifier (Reference to VnfPkgInfo)	Identifies the VnfPkgInfo objects for the VNFD referenced by the on-boarded NSD. See note 1.
pnfdInfoId	M	0..N	Identifier (Reference to PnfdInfo)	Identifies the PNFD information object for the PNFD referenced by the on-boarded NSD. See note 1.
nestedNsdInfoId	M	0..N	Identifier (Reference to NsdInfo)	Identifies the NSD information object for the nested NSD referenced by the on-boarded NSD. See note 1.
artifacts	M	0..N	NsdArchiveArtifactInformation	Information about artifacts contained in the NSD archive. See note 2.
onboardingState	M	1	Enum	On-boarding state of the NSD. VALUES: • CREATED • UPLOADING • PROCESSING • ONBOARDED
operationalState	M	1	Enum	Operational state of the NSD. VALUES: • ENABLED • DISABLED
usageState	M	1	Enum	Usage state of the NSD. VALUES: • IN_USE • NOT_IN_USE
userDefinedData	O	0..N	KeyValuePair	User defined data for the NSD.
NOTE 1: These attributes shall be present after the NSD is on-boarded. NOTE 2: The attribute may be present after the NSD archive is on-boarded and shall be absent otherwise.				

8.2.3 Pnfd information element

8.2.3.1 Description

This information element provides the details of the PNFD.

8.2.3.2 Attributes

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

8.2.4 PnfdInfo information element

8.2.4.1 Description

This information element provides the details of a PNFD.

8.2.4.2 Attributes

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

Table 8.2.4.2-1: Attributes of the PnfdInfo information element

Attribute	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier	Identifier of the on-boarded instance of the PNFD.
pnfdId	M	0..1	Identifier	Identifier of the on-boarded PNFD. See note 1.
name	M	0..1	String	Name of the on-boarded PNFD. See note 1.
version	M	0..1	Version	Version of the on-boarded PNFD. See note 1.
provider	M	0..1	String	Provider of the on-boarded PNFD. See note 1.
pnfdInvariantId	M	0..1	Identifier	Identifies a PNFD in a version independent manner. This attribute is invariant across versions of PNFD. See note 1.
pnfd	M	0..1	Identifier (Reference to Pnfd)	Reference to the on-boarded PNFD, e.g. URL to the on-boarded PNFD. See note 1.
artifacts	M	0..N	PnfdArchiveArtifactInformation	Information about artifacts contained in the PNFD archive. See note 2.
onboardingState	M	1	Enum	On-boarding state of the PNFD. VALUES: • CREATED • UPLOADING • PROCESSING • ONBOARDED
usageState	M	1	Enum	Usage state of the PNFD. VALUES: • IN_USE • NOT_IN_USE
userDefinedData	O	0..N	KeyValuePair	User defined data for the PNFD.
NOTE 1: These attributes shall be present after the PNFD is on-boarded. NOTE 2: The attribute may be present after the PNFD archive is on-boarded and shall be absent otherwise.				

8.2.5 Nsd information element

8.2.5.1 Description

This information element provides the details of the NSD.

8.2.5.2 Attributes

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

8.2.6 NsdOnBoardingNotification

8.2.6.1 Description

This notification indicates that a new NSD is on-boarded, after all the on-boarding steps (e.g. uploading and processing) are done. A change in on-boarding state before the NSD is on-boarded is not reported.

Support of this notification is mandatory.

8.2.6.2 Trigger Conditions

The notification is produced when:

- New NSD is on-boarded.

8.2.6.3 Attributes

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

Table 8.2.6.3-1: Attributes of the NsdOnBoardingNotification

Attribute	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier (Reference to NsdInfo)	Identifier of the NSD information object.
nsdId	M	1	Identifier (Reference to Nsd)	Identifies the on-boarded NSD.

8.2.7 NsdChangeNotification

8.2.7.1 Description

This notification indicates a change of state in an on-boarded NSD. Only a change in operational state will be reported. A change in usage state is not reported.

Support of this notification is mandatory.

8.2.7.2 Trigger Conditions

The notification is produced when:

- Change of the operational state of an on-boarded NSD.

8.2.7.3 Attributes

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

Table 8.2.7.3-1: Attributes of the NsdChangeNotification

Attribute	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier (Reference to NsdInfo)	Identifier of the NSD information object.
nsdId	M	1	Identifier (Reference to Nsd)	Identifies the on-boarded NSD.
operationalState	M	0..1	Enum	New operational state of the on-boarded NSD. VALUES: • ENABLED • DISABLED

8.2.8 NsdDeletionNotification

8.2.8.1 Description

This notification indicates an on-boarded NSD is deleted. Support of this notification is mandatory.

8.2.8.2 Trigger Conditions

The notification is produced when:

- An on-boarded NSD is deleted.

8.2.8.3 Attributes

The attributes of the NsdOnBoardingNotification shall follow the indications provided in table 8.2.8.3-1.

Table 8.2.8.3-1: Attributes of the NsdDeletionNotification

Attribute	Qualifier	Cardinality	Content	Description
nsdInfoId	M	1	Identifier (Reference to NsdInfo)	Identifier of the deleted NSD information object.
nsdId	M	1	Identifier (Reference to Nsd)	Identifies the deleted NSD.

8.2.9 PnfdOnBoardingNotification

8.2.9.1 Description

This notification indicates that a new PNFD is on-boarded, after all the on-boarding steps (e.g. uploading and processing) are done. A change in on-boarding state before the PNFD is on-boarded is not reported.

Support of this notification is mandatory.

8.2.9.2 Trigger Conditions

The notification is produced when:

- New PNFD is on-boarded.

8.2.9.3 Attributes

The attributes of the PnfdOnBoardingNotification shall follow the indications provided in table 8.2.9.3-1.

Table 8.2.9.3-1: Attributes of the PnfdOnBoardingNotification

Attribute	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier (Reference to PnfdInfo)	Identifier of the PNFD information object.
pnfdId	M	1	Identifier (Reference to Pnfd)	Identifies the on-boarded PNFD.

8.2.10 PnfdDeletionNotification

8.2.10.1 Description

This notification indicates an on-boarded PNFD is deleted. Support of this notification is mandatory.

8.2.10.2 Trigger Conditions

The notification is produced when:

- An on-boarded PNFD is deleted.

8.2.10.3 Attributes

The attributes of the PnfdDeletionNotification shall follow the indications provided in table 8.2.10.3-1.

Table 8.2.10.3-1: Attributes of the PnfdDeletionNotification

Attribute	Qualifier	Cardinality	Content	Description
pnfdInfoId	M	1	Identifier (Reference to PnfdInfo)	Identifier of the deleted PNFD information object.
pnfdId	M	1	Identifier (Reference to Pnfd)	Identifies the deleted PNFD.

8.2.11 NsdArchiveArtifactInformation information element

8.2.11.1 Description

This information element provides identification information for an artifact which is contained in the NSD archive.

8.2.11.2 Attributes

The NsdArchiveArtifactInformation information element shall follow the indications provided in table 8.2.11.2-1.

Table 8.2.11.2-1: Attributes of the NsdArchiveArtifactInformation information element

Attribute	Qualifier	Cardinality	Content	Description
selector	M	1	Not specified	Information (such as a path) that identifies this artifact in the NSD archive.
metadata	M	1	Not specified	The metadata of the artifact that are available in the NSD archive, such as Content type, size, creation date, etc.

8.2.12 PnfdArchiveArtifactInformation information element

8.2.12.1 Description

This information element provides identification information for an artifact which is contained in the PNFD archive.

8.2.12.2 Attributes

The PnfdArchiveArtifactInformation information element shall follow the indications provided in table 8.2.12.2-1.

Table 8.2.12.2-1: Attributes of the PnfdArchiveArtifactInformation information element

Attribute	Qualifier	Cardinality	Content	Description
selector	M	1	Not specified	Information (such as a path) that identifies this artifact in the PNFD archive.
metadata	M	1	Not specified	The metadata of the artifact that are available in the PNFD archive, such as Content type, size, creation date, etc.

8.3 Information elements and notifications related to NS Lifecycle Management

8.3.1 Introduction

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

8.3.2 Information elements and notifications related to NS Lifecycle Changes

8.3.2.1 Introduction

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

8.3.2.2 NsLcmOperationOccurrenceNotification

8.3.2.2.1 Description

This notification informs the receiver of changes in the NS lifecycle caused by NS lifecycle management operation occurrences, which may be manually triggered by the OSS/BSS or automatically triggered by the NFVO. The automatic trigger inside the NFVO includes auto-scaling, auto-healing and impact on the nested NS instances triggered by the NS lifecycle operation on its composite NS. The support of the notification is mandatory.

8.3.2.2.2 Trigger conditions

This notification is produced when there is a change in the NS lifecycle caused by NS lifecycle management operation occurrences, including:

- Instantiation of the NS (start and result, including feasibility check).
- Scaling of the NS (start and result, including the auto-scaling).
- Update of the NS (start and result, including feasibility check).
- Termination of the NS (start and result).
- Healing of the NS (start and result, including the auto-healing).
- Impact on the nested NS instances triggered by the NS lifecycle operation on its composite NS.

If this is a notification about the start of an LCM operation occurrence, the notification shall be sent before any action is taken, however, after acknowledging the LCM operation request to the consumer.

If this is a notification about the result of an LCM operation, the notification shall be sent after all other actions of the LCM operation have been executed.

If this is a notification about the result of an unsuccessful LCM operation occurrence and the cause is a resource shortage, the notification shall include appropriate information about the resource shortage.

If this is a notification where a pre-emption occurred due to e.g. a higher priority LCM operation during resource shortage, the notification shall include appropriate information about the pre-emption.

If this is a notification about the result of successful feasibility check of an NS LCM operation, the notification shall include "feasibilityCheckResult".

If this is a notification about the result of failure(s) observed during the feasibility check of an NS LCM operation, the notification shall include appropriate information about the failures experienced in performing the NS LCM operation as part of the feasibility check. It is up to the protocol design stage to determine on how the failure information is sent in the notification.

8.3.2.2.3 Attributes

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

Table 8.3.2.2.3-1: Attributes of the NsLcmOperationOccurrenceNotification

Attribute	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the NS instance affected.
lifecycleOperationOccurrenceId	M	1	Identifier	Identifier of the NS lifecycle operation occurrence associated to the notification.
operation	M	1	Not specified	The lifecycle operation.
notificationStatus	M	1	Enum	Indicates whether this notification reports about the start of a lifecycle operation occurrence or the result of a lifecycle operation occurrence. VALUES: • START: Informs about the start of the NS LCM operation occurrence. • RESULT: Informs about the final or intermediate result of the NS LCM operation occurrence.
operationStatus	M	1	Not specified	Indicates the operation status. See note 1.
isAutomaticInvocation	M	1	Boolean	Set to true if the NS lifecycle operation occurrence has been automatically triggered by the NFVO. The automatic trigger inside the NFVO includes auto-scaling, auto-healing and impact on the nested NS instances triggered by the NS lifecycle operation on its composite NS. Set to false otherwise.
affectedVnf	M	0..N	AffectedVnf	Information about the VNF instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.

Attribute	Qualifier	Cardinality	Content	Description
affectedPnf	M	0..N	AffectedPnf	Information about the PNF instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
affectedVl	M	0..N	AffectedVirtualLink	Information about the VL instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
affectedVnffg	M	0..N	AffectedVnffg	Information about the VNFFG instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
affectedNs	M	0..N	AffectedNs	Information about the nested NS instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
affectedSap	M	0..N	AffectedSap	Information about the SAP instances that were affected during the lifecycle operation, if this notification represents the result of a lifecycle operation. See note 2.
feasibilityCheckResult	M	0..1	Enum	Information about the feasibility check result. The parameter shall be provided if the notification is sent to indicate the result of feasibility check. VALUES: • FEASIBILITY_CHECK_DONE • FEASIBILITY_CHECK_DONE_WITH_RESERVATION • etc.
NOTE 1: If this notification represents the result of a lifecycle operation that was not successful, the notification shall contain appropriate error information. NOTE 2: If the notification represents the successful result of a lifecycle operation, at least an affectedVnf, or affectedPnf, or affectedVl, or affectedVnffg or affectedNs, or affectedSap shall be present.				

8.3.2.3 AffectedVnf information element

8.3.2.3.1 Description

This information element provides information about affected VNF instances.

8.3.2.3.2 Attributes

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

Table 8.3.2.3.2-1: Attributes of the AffectedVnf information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.
vnfdId	M	1	Identifier (Reference to Vnfd)	Identifier of the VNFD of the VNF instance.
vnfProfileId	M	1	Identifier (Reference to VnfProfile)	Identifier of the VNF profile of the NSD.
vnfName	M	1	String	Name of the VNF instance.
changeType	M	1	Enum	Signals the type of lifecycle change. VALUES: • ADD • REMOVE • INSTANTIATE • TERMINATE • SCALE • CHANGE_FLAVOUR • HEAL • OPERATE • MODIFY_INFORMATION • CHANGE_EXT_VNF_CONNECTIVITY • REVERT_TO_VNF_SNAPSHOT • CHANGE_CURRENT_VNF_PKG • ASSOCIATE_WITH_VNF_PROFILE • etc.
changeResult	M	1	Enum	Signal the result of lifecycle change. VALUES: • COMPLETED • FAILED • etc.
changedInfo	M	0..1	Not specified	Information about the changed VNF instance information, including VNF configurable properties, if applicable.

8.3.2.4 AffectedPnf information element

8.3.2.4.1 Description

This information element provides information about affected PNFs from an NS.

8.3.2.4.2 Attributes

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

Table 8.3.2.4.2-1: Attributes of the AffectedPnf information element

Attribute	Qualifier	Cardinality	Content	Description
pnfId	M	1	Identifier	Identifier of the PNF. Assigned by OSS.
pnfName	M	1	String	Human readable name of the PNF.
pnfdId	M	1	Identifier (Reference to Pnfd)	Identifier of the PNFD.
pnfProfileId	M	1	Identifier (Reference to PnfProfile)	Identifier of the PNF profile of the NSD.
cpInstanceId	M	1..N	Identifier (Reference to PnfExtCpInfo)	Identifier of the affected PNF external CP instance.
changeType	M	1	Enum	Signals the type of lifecycle change. VALUES: • ADD • MODIFY • REMOVE

Attribute	Qualifier	Cardinality	Content	Description
changeResult	M	1	Enum	Signal the result of lifecycle change. VALUES: • COMPLETED • FAILED • etc.

8.3.2.5 AffectedVirtualLink information element

8.3.2.5.1 Description

This information element provides information about affected VLs of an NS.

8.3.2.5.2 Attributes

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

Table 8.3.2.5.2-1: Attributes of the AffectedVirtualLink information element

Attribute	Qualifier	Cardinality	Content	Description
nsVirtualLinkId	M	1	Identifier	Identifier of the VL instance.
nsVirtualLinkDescId	M	1	Identifier (Reference to NsVirtualLinkDesc)	Identifier of the VLD in the NSD for this VL.
virtualLinkProfileId	M	1	Identifier (Reference to VirtualLinkProfile)	Identifier of the VL profile of the NSD.
changeType	M	1	Enum	Signals the type of lifecycle change. VALUES: • ADD • DELETE • MODIFY • ADD_LINK_PORT • REMOVE_LINK_PORT
linkPortId	M	0..N	Identifier (Reference to NsLinkPortInfo)	Identifiers of the link ports of the affected VL related to the change. Shall be set when changeType is equal to "ADD_LINK_PORT" or "REMOVE_LINK_PORT", and the related link ports are present (case "add") or have been present (case "remove") in the NS VL represented by the "virtualLinkInfo" attribute in the "NsInfo".
changeResult	M	1	Enum	Signal the result of lifecycle change. VALUES: • COMPLETED • FAILED • etc.

8.3.2.6 AffectedVnffg information element

8.3.2.6.1 Description

This information element provides information about affected VNFFG instances.

8.3.2.6.2 Attributes

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

Table 8.3.2.6.2-1: Attributes of the AffectedVnffg information element

Attribute	Qualifier	Cardinality	Content	Description
vnffgId	M	1	Identifier	Identifier of the VNFFG instance.
vnffgdId	M	1	Identifier (Reference to Vnffgd)	Identifier of the VNFFGD of the VNFFG instance.
changeType	M	1	Enum	Signals the type of lifecycle change. VALUES: • ADD • REMOVE • MODIFY See note.
changeResult	M	1	Enum	Signal the result of lifecycle change. VALUES: • COMPLETED • FAILED • etc.
NOTE: CP or NFP information might be modified for the VNFFG.				

8.3.2.7 AffectedNs information element

8.3.2.7.1 Description

This information element provides information about affected nested NSs.

8.3.2.7.2 Attributes

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

Table 8.3.2.7.2-1: Attributes of the AffectedNs information element

Attribute	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the nested NS instance.
nsdId	M	1	Identifier (Reference to Nsd)	Identifier of the NSD of the nested NS instance.
changeType	M	1	Enum	Signals the type of lifecycle change. VALUES: • ADD • REMOVE • INSTANTIATE • TERMINATE • SCALE • HEAL • UPDATE
changeResult	M	1	Enum	Signal the result of lifecycle change. VALUES: • COMPLETED • FAILED • etc.

8.3.2.8 AffectedSap information element

8.3.2.8.1 Description

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

8.3.2.8.2 Attributes

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

Table 8.3.2.8.2-1: Attributes of the AffectedSap information element

Attribute	Qualifier	Cardinality	Content	Description
sapInstanceId	M	1	Identifier	Identifier of this SapInfo information element, identifying the SAP instance.
sapdId	M	1	Identifier (Reference to Sapd)	Reference to the SAPD for this SAP.
sapName	M	1	String	Human readable name for the SAP.
changeType	M	1	Enum	Signals the type of lifecycle change. VALUES: • ADD • REMOVE • MODIFY
changeResult	M	1	Enum	Signal the result of lifecycle change. VALUES: • COMPLETED • FAILED • etc.

8.3.2.9 NsIdentifierCreationNotification

8.3.2.9.1 Description

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

8.3.2.9.2 Trigger conditions

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

8.3.2.9.3 Attributes

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

Table 8.3.2.9.3-1: Attributes of the NsIdentifierCreationNotification

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

8.3.2.10 NsIdentifierDeletionNotification

8.3.2.10.1 Description

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

8.3.2.10.2 Trigger conditions

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

8.3.2.10.3 Attributes

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

Table 8.3.2.10.3-1: Attributes of the NsIdentifierDeletionNotification

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

8.3.2.11 NsChangeNotification

8.3.2.11.1 Description

This notification informs the receiver of changes on the NS instance caused by the LCM operation occurrence, which directly or indirectly impacts its NS component and is triggered without any context of this NS instance. This notification is different from the NsLcmOperationOccurrenceNotification (see clause 8.3.2.2), which is triggered by the LCM operation occurrence on the NS instance itself. The support of the notification is mandatory.

8.3.2.11.2 Trigger conditions

The trigger conditions include:

- LCM operation occurrence which directly or indirectly impacts the NS component (start and result).

If this is a notification about the start of an LCM operation occurrence impacting the NS component, the notification shall be provided as soon as the impact on the NS component is identified.

If this is a notification about the result of an LCM operation occurrence impacting the NS component, the notification shall be provided after the impact on the NS component has been executed.

8.3.2.11.3 Attributes

The attributes of the NsChangeNotification notification shall follow the indications provided in table 8.3.2.11.3-1.

Table 8.3.2.11.3-1: Attributes of the NsChangeNotification

Attribute	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of the NS instance affected.
nsComponentType	M	1	Enum	Indicates the affected NS component type. VALUES: • VNF • PNF • NS: Indicates a nested NS
nsComponentId	M	1	Identifier	Identifier of the affected NS component instance.
lcmOpOccIdImpactingNsComponent	M	1	Identifier	Identifier of the lifecycle management operation occurrence impacting the NS component and associated to this notification.
lcmOpOccNameImpactingNsComponent	M	1	String	Name of the lifecycle management operation occurrence impacting the NS component and associated to this notification.
lcmOpOccStatusImpactingNsComponent	M	1	Not specified	Status of the lifecycle management operation occurrence impacting the NS component and associated to this notification. Indicates whether this notification reports about the start of a lifecycle operation occurrence or the final result of a lifecycle operation occurrence, e.g. start, completed, failed, etc.

8.3.3 Information elements related to NsInfo

8.3.3.1 Introduction

The clauses below define information elements related to NsInfo.

8.3.3.2 NsInfo information element

8.3.3.2.1 Description

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

8.3.3.2.2 Attributes

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

Table 8.3.3.2.2-1: Attributes of the NsInfo information element

Attribute	Qualifier	Cardinality	Content	Description
nsInstanceId	M	1	Identifier	Identifier of this NsInfo information element, identifying the NS instance.
nsName	M	1	String	Human readable name of the NS instance.
description	M	1	String	Human readable description of the NS instance.
nsdId	M	1	Identifier (Reference to Nsd)	Reference to the NSD associated with this NS. This is the NSD used to instantiate this NS or an NSD explicitly associated after instantiation.
nsdInfoId	M	1	Identifier (Reference to NsdInfo)	Reference to the NSD information object associated with the NS. This identifier was allocated by the NFVO.
flavourId	M	0..1	Identifier (Reference to NsDf)	Reference to the flavour of the NSD used to instantiate this NS. See notes 1 and 2.
vnfInfo	M	0..N	VnfInfo	Information on constituent VNFs of this NS.
pnfInfo	M	0..N	PnfInfo	Information on the PNF(s) that are part of this NS.
virtualLinkInfo	M	0..N	NsVirtualLinkInfo	Information on the VLs of this NS.
vnffgInfo	M	0..N	VnffgInfo	Information on the VNFFGs of this NS.
sapInfo	M	0..N	SapInfo	Information on the SAPs of this NS.
nestedNsInfoId	M	0..N	Identifier (Reference to NsInfo)	Reference to information on nested NSs of this NS.
vnfSnapshotInfo	M	0..N	VnfSnapshotInfo	Information on Snapshots of VNFs that are part of this NS. See note 4.
nsState	M	1	Enum	The state of the NS. VALUES: • NOT_INSTANTIATED: the NS instance is not instantiated or terminated • INSTANTIATED: the NS instance is instantiated

Attribute	Qualifier	Cardinality	Content	Description
monitoringParameter	M	0..N	Not specified	Performance metrics tracked by NFVO (e.g. for auto-scaling purposes). See note 3.
nsScaleStatus	M	0..N	NsScaleInfo	Represents for each NS scaling aspect declared in the applicable DF, how "big" the NS instance has been scaled w.r.t. that aspect.
additionalAffinityOrAntiAffinityRule	M	0..N	AffinityOrAntiAffinityRule	Information on the additional affinity or anti-affinity rule from NS instantiation operation. Shall not conflict with rules already specified in the NSD. See clause 8.3.4.26.
wanConnectionInfo	M	0..N	WanConnectionInfo	Information about WAN related connectivity enabling multi-site VLs.

NOTE 1: The NsDf information element is defined in ETSI GS NFV-IFA 014 [3], clause 6.3.2.
NOTE 2: Cardinality of zero is only valid for a non-instantiated NS.
NOTE 3: The monitoring parameters to be tracked by NFVO are identified by NSD designer in the NSD.
NOTE 4: The NFVO shall keep information about the VNF snapshots even if they are not associated to an NS instance, as it shall still be possible to query, delete, and package VNF snapshots after the termination of the NS instance.

8.3.3.3 VnfInfo information element

8.3.3.3.1 Description

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

8.3.3.3.2 Attributes

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

Table 8.3.3.3.2-1: Attributes of the VnfInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance that is represented by this VnfInfo information element.
vnfInstanceName	M	0..1	String	VNF instance name. See note 1.
vnfInstanceDescription	M	0..1	String	Human-readable description of the VNF instance. See note 1.
vnfdId	M	1	Identifier (Reference to Vnfd)	Identifier of the VNFD on which the VNF instance is based. See notes 1, 2 and 4.
vnfProvider	M	1	String	Provider of the VNF and the VNFD. See note 3.
vnfProductName	M	1	String	Name to identify the VNF Product. See note 3.
vnfSoftwareVersion	M	1	Version	Software version of the VNF. See note 3.
vnfdVersion	M	1	Version	Identifies the version of the VNFD. See note 3.

Attribute	Qualifier	Cardinality	Content	Description
vnfConfigurableProperty	M	0..N	KeyValuePair	Additional VNF-specific attributes that provide the current values of the configurable properties of the VNF instance. These attributes represent values that are stored persistently in the VnfInfo information element and that correspond to configuration parameters of the VNF instance. Modifying the values of these attributes directly affects the configuration of the VNF instance if it exists. Configurable properties referred in this attribute are declared in the VNFD (see clause 7.1.12 in ETSI GS NFV-IFA 011 [2]). See notes 1 and 5.
instantiationState	M	1	Enum	The instantiation state of the VNF. VALUES: • NOT_INSTANTIATED: VNF instance is terminated or not instantiated, and the identifier of the VNF instance exists • INSTANTIATED: VNF is instantiated
instantiatedVnfInfo	M	0..1	InstantiatedVnfInfo	Information specific to an instantiated VNF instance. Shall be present if the VNF is in INSTANTIATED instantiation state.
metadata	M	0..N	KeyValuePair	Additional VNF-specific attributes that provide metadata describing the VNF instance. Metadata that are writeable are declared in the VNFD (see clause 7.1.14.2 in ETSI GS NFV-IFA 011 [2]). See note 1. These attributes represent values that are stored persistently in the VnfInfo information element for consumption by functional blocks that invoke the VNF lifecycle management interface. They are not consumed by the VNFM or the lifecycle management scripts. Modifying the values of these attributes has no effect on the VNF instance, it only affects the information represented in VnfInfo.

Attribute	Qualifier	Cardinality	Content	Description
extension	M	0..N	KeyValuePair	Additional VNF-specific attributes that affect the lifecycle management of this VNF instance. These attributes represent values that are stored persistently in the VnfInfo information element for consumption by the VNFM, or the lifecycle management scripts during the execution of VNF lifecycle management operations. Modifying the values of these attributes has no direct effect on the VNF instance; however, the modified attribute values can be considered during subsequent VNF lifecycle management operations, which means that the modified values can indirectly affect the configuration of the VNF instance. Extensions that are writeable are declared in the VNFD (see clause 7.1.14.2 in ETSI GS NFV-IFA 011 [2]). See note 1.

NOTE 1: This attribute in the VnfInfo shall be writable through the modifyVnfInfoData attribute of the Update NS operation (refer to clause 7.3.5.2).
NOTE 2: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way.
NOTE 3: See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package which was used to instantiate the VNF instance.
NOTE 4: Modifying the value of this attribute can be performed when no conflicts exist between the previous and the newly referred VNF Package, e.g. when the new VNFD is not changed with respect to the previous VNFD apart from referencing to other VNF software image(s). In order to avoid misalignment of the VnfInfo with the current VNF's on-boarded VNF Package, the values copied from the VNFD of the on-boarded VNF Package (see note 3) need to be kept in sync.
NOTE 5: VNF configurable properties are sometimes also referred to as configuration parameters applicable to a VNF. Some of these are set prior to instantiation and cannot be modified if the VNF is instantiated, some are set prior to instantiation (are part of initial configuration) and can be modified later, and others can be set only after instantiation. The applicability of certain configuration may depend on the VNF and the required operation of the VNF at a certain point in time.

8.3.3.4 InstantiatedVnfInfo information element

8.3.3.4.1 Description

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

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

8.3.3.4.2 Attributes

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

Table 8.3.3.4.2-1: Attributes of the InstantiatedVnfInfo information element

Attribute	Qualifier	Cardinality	Content	Description
flavourId	M	1	Identifier (Reference to VnfDf)	Identifier of the VNF DF applied to this VNF instance. See note 1.
vnfState	M	1	Enum	The state of the VNF instance. VALUES: • STARTED • STOPPED
scaleStatus	M	0..N	ScaleInfo	Scale status of the VNF, one entry per aspect. Shall be present if the VNF supports scaling. Represents for every scaling aspect how "big" the VNF has been scaled w.r.t. that aspect. See note 2.
maxScaleLevel	M	0..N	ScaleInfo	Maximum allowed scale levels of the VNF, one entry per aspect, as defined in the VNFD. This attribute shall be present if the VNF supports scaling. Represents for every scaling aspect how "big" the VNF can be scaled w.r.t. that aspect. See note 2.
extCpInfo	M	1..N	VnfExtCpInfo	External CPs exposed by the VNF instance.
vipCpInfo	M	1..N	VipCpInfo	VIP CPs that are part of the VNF instance. Shall be present when that particular VIP CP of the VNFC instance is associated to an external CP of the VNF instance. May be present otherwise.
extVirtualLinkInfo	M	0..N	ExtVirtualLinkInfo	External VLs the VNF instance is connected to.
extManagedVirtualLinkInfo	M	0..N	ExtManagedVirtualLinkInfo	Externally-managed internal VLs of the VNF instance. See note 4.
monitoringParameter	M	0..N	Not specified	Performance metrics tracked by VNFM (e.g. for auto-scaling purposes). See note 3.
localizationLanguage	M	0..1	Not specified	Information about localization language of the VNF (includes e.g. strings in the VNFD). The localization languages supported by a VNF can be declared in the VNFD, and localization language selection can take place at instantiation time.
vimId	M	0..N	Identifier	Identifier of a VIM that manages resources for the VNF instance.
vnfcResourceInfo	M	0..N	VnfcResourceInfo	Information on the virtualised compute and storage resource(s) used by the VNFCs of the VNF instance.
vnfVirtualLinkResourceInfo	M	0..N	VnfVirtualLinkResourceInfo	Information on the virtualised network resource(s) used by the VLs of the VNF instance.
virtualStorageResourceInfo	M	0..N	VirtualStorageResourceInfo	Information on the virtualised storage resource(s) used as storage for the VNF instance.
mcioInfo	M	0..N	McioInfo	Information on the MCIO(s) representing VNFC instance(s) realized by one or a set of OS containers and created from the same VDU for the VNF instance.

NOTE 1: The VnfDf information element is defined in ETSI GS NFV-IFA 011 [2], clause 7.1.8.2.

NOTE 2: For every scaling aspect, the information provided by the "scaleStatus" and "maxScaleLevel" attributes allows an external entity to derive how many scaling steps are possible for scaling in or scaling out a VNF instance. Per aspect, the number of steps possible to scale in corresponds to the "scaleLevel" attribute for that aspect in the "scaleStatus" information element, and the possible number of steps to scale out corresponds to the difference between "maxScaleLevel" for that aspect, and the "scaleLevel" attribute for that aspect in the "scaleStatus" information element.

NOTE 3: The monitoring parameters to be tracked by VNFM are identified by VNF provider in the VNFD. The VNFM collects the values of identified performance metrics using one or more locally initiated PM Jobs.

NOTE 4: It is possible to have several ExtManagedVirtualLinkInfo for the same VNF internal VL in case of a multi-site VNF spanning several VIMs. The set of ExtManagedVirtualLinkInfo corresponding to the same VNF internal VL shall indicate so by referencing to the same VnfVirtualLinkDesc and externally-managed multi-site VL instance (refer to clause 8.3.3.19).

8.3.3.5 VnfcResourceInfo information element

8.3.3.5.1 Description

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

8.3.3.5.2 Attributes

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

Table 8.3.3.5.2-1: Attributes of the VnfcResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfcInstanceId	M	1	Identifier	Identifier of this VNFC instance.
vduId	M	1	Identifier (Reference to Vdu)	Reference to the applicable Vdu information element in the VNFD.
vnfdId	M	0..1	Identifier (Reference to Vnfd)	Identifier of the VNFD. Shall be present in case the value differs from the vnfdId attribute of the VNF instance (e.g. during a "Change current VNF package" operation or due to its final failure).
computeResource	M	1	ResourceHandle	Reference to the VirtualCompute resource or reference to a Compute MCIO.
storageResourceId	M	0..N	Identifier (Reference to VirtualStorageResourceInfo)	Reference(s) to the VirtualStorage resource(s) or reference(s) to Storage MCIO(s).
reservationId	M	0..1	Identifier	The reservation identifier applicable to the resource. It shall be present when an applicable reservation exists.
vnfcCpInfo	M	0..N	VnfcCpInfo	CP(s) of the VNFC instance. Shall be present when that particular CP of the VNFC instance is associated to an external CP of the VNF instance. May be present otherwise.
metadata	M	0..N	KeyValuePair	Metadata about this resource.
trunkPortsInfo	M	0..N	TrunkPortsInfo	Collections of CPs of the VNFC instance in trunk(s). Shall be present when the VNFC has CPs working in trunk mode, as parent port of a trunk, and other CPs working as subports of the same trunk, and the referred CP instances are also present in the vnfcCpInfo attribute.

8.3.3.6 VnfVirtualLinkResourceInfo information element

8.3.3.6.1 Description

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

8.3.3.6.2 Attributes

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

Table 8.3.3.6.2-1: Attributes of the VnfVirtualLinkResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
virtualLinkInstanceId	M	1	Identifier	Identifier of this VL instance.
vnfVirtualLinkDescId	M	1	Identifier (Reference to VnfVirtualLinkDesc)	Identifier of the VNF VLD in the VNFD.
vnfdId	M	0..1	Identifier (Reference to Vnfd)	Identifier of the VNFD. Shall be present in case the value differs from the vnfdId attribute of the VNF instance (e.g. during a "Change current VNF package" operation or due to its final failure).
networkResource	M	1	ResourceHandle	Reference to the VirtualNetwork resource or reference to a Network MCIO.
reservationId	M	0..1	Identifier	The reservation identifier applicable to the resource. It shall be present when an applicable reservation exists.
vnfLinkPort	M	0..N	VnfLinkPortInfo	Links ports of this VL. Shall be present when the linkPort is used for external connectivity by the VNF (refer to VnfLinkPortInfo in clause 8.3.3.20). May be present otherwise.
metadata	M	0..N	KeyValuePair	Metadata about this resource.

8.3.3.7 VirtualStorageResourceInfo information element

8.3.3.7.1 Description

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

8.3.3.7.2 Attributes

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

Table 8.3.3.7.2-1: Attributes of the VirtualStorageResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
virtualStorageInstanceId	M	1	Identifier	Identifier of this virtual storage resource instance.
virtualStorageDescId	M	1	Identifier (Reference to VirtualStorageDesc)	Identifier of the VirtualStorageDesc in the VNFD.
vnfdId	M	0..1	Identifier (Reference to Vnfd)	Identifier of the VNFD. Shall be present in case the value differs from the vnfdId attribute of the VNF instance (e.g. during a "Change current VNF package" operation or due to its final failure).
storageResource	M	1	ResourceHandle	Reference to the VirtualStorage resource(s) or reference to a Storage MCIO.
reservationId	M	0..1	Identifier	The reservation identifier applicable to the resource. It shall be present when an applicable reservation exists.
metadata	M	0..N	KeyValuePair	Metadata about this resource.

8.3.3.8 ResourceHandle information element

8.3.3.8.1 Description

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

8.3.3.8.2 Attributes

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

Table 8.3.3.8.2-1: Attributes of the ResourceHandle information element

Attribute	Qualifier	Cardinality	Content	Description
vimId	CM	0..1	Identifier	Identifier of the VIM or WIM or CISM under whose control this resource is placed. CONDITION: This attribute shall be supported when VNF-related Resource Management in direct mode is applicable. It shall also be supported for resources that are part of an NS instance such as virtual link resources.
resourceProviderId	CM	0..1	Identifier	Identifies the entity responsible for the management of the virtualised resource. CONDITION: This attribute shall be supported when VNF-related Resource Management in indirect mode is applicable.
resourceId	M	1	Identifier	Identifier of the resource in the scope of the VIM, the WIM, the CISM or the resource provider.
vimLevelResourceType	M	0..1	Not specified	Type of the resource in the scope of the VIM, the WIM, the CISM or the resource provider. See note 1.
vimLevelAdditionalResourceInfo	M	0..1	Not specified	Additional resource information which is specific to this resource and its type, and which is available from the VIM, the WIM, the CISM or the resource provider. See note 2.
NOTE 1: The value set of the "vimLevelResourceType" attribute is within the scope of the VIM, the WIM or the resource provider and can be used as information that complements the ResourceHandle. NOTE 2: Which structure and content of the resource information to be expected depends on the type of resource and its provider. The information shall be limited to properties directly owned by the resource referenced in this ResourceHandle.				

8.3.3.9 PnfInfo information element

8.3.3.9.1 Description

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

8.3.3.9.2 Attributes

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

Table 8.3.3.9.2-1: Attributes of the PnfInfo information element

Attribute	Qualifier	Cardinality	Content	Description
pnfId	M	1	Identifier	Identifier of the PNF. Assigned by OSS and provided to NFVO.
pnfName	M	1	String	Human readable name of the PNF.
pnfdId	M	1	Identifier (Reference to Pnfd)	Identifier of the PNFD.
pnfdInfoId	M	1	Identifier (Reference to PnfdInfo)	Identifier of (reference to) the PNFD information related to this PNF.
pnfProfileId	M	1	Identifier (Reference to PnfProfile)	Identifier of (reference to) the PNF Profile to be used for this PNF.
cpInfo	M	1..N	PnfExtCpInfo	Information on the external CP of the PNF.

8.3.3.10 NsVirtualLinkInfo information element

8.3.3.10.1 Description

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

As an NS can include NFs deployed in NFVI-PoPs under the control of several different VIMs, therefore deploying an NS VL can involve several VIMs, each allocating different virtualised network resources, as well as WIMs handling the connectivity in between the NFVI-PoPs in the form of multi-site connectivity services.

When this NsVirtualLinkInfo is provided as an ExtVirtualLinkInfo as input of a VNF LCM operation, the id of the ExtVirtualLinkInfo shall be the one of the corresponding NsVirtualLinkInfo.

8.3.3.10.2 Attributes

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

Table 8.3.3.10.2-1: Attributes of the NsVirtualLinkInfo information element

Attribute	Qualifier	Cardinality	Content	Description
nsVirtualLinkInstanceId	M	1	Identifier	Identifier of this NsVirtualLinkInfo information element, identifying the NS VL instance.
nsVirtualLinkDescId	M	1	Identifier (Reference to NsVirtualLinkDesc)	Identifier of the VLD in the NSD for this VL.
virtualLinkProfileId	M	1	Identifier (Reference to VirtualLinkProfile)	Identifier of the VL profile in the NSD for this VL.
resourceHandle	M	1..N	ResourceHandle	Identifier(s) of the virtualised network resource(s) and/or multi-site connectivity service(s) realizing this VL.
linkPort	M	0..N	NsLinkPortInfo	Link ports of this VL. Cardinality of zero indicates that no port has yet been created for this VL.

8.3.3.11 NsLinkPortInfo information element

8.3.3.11.1 Description

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

When the NsVirtualLinkInfo, from which the present NsLinkPortInfo is part of, is provided as an ExtVirtualLinkInfo as input of a VNF LCM operation, the id of the ExtLinkPortInfo shall be identical to the one of the corresponding NsLinkPortInfo.

8.3.3.11.2 Attributes

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

Table 8.3.3.11.2-1: Attributes of the NsLinkPortInfo information element

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

8.3.3.12 SapInfo information element

8.3.3.12.1 Description

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

8.3.3.12.2 Attributes

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

Table 8.3.3.12.2-1: Attributes of the SapInfo information element

Attribute	Qualifier	Cardinality	Content	Description
sapInstanceId	M	1	Identifier	Identifier of this SapInfo information element, identifying the SAP instance.
sapdId	M	1	Identifier (Reference to Sapd)	Reference to the SAPD for this SAP.
sapName	M	1	String	Human readable name for the SAP.
description	M	1	String	Human readable description for the SAP.
cpProtocolInfo	M	1..N	CpProtocolInfo	Protocol information for this SAP. There shall be one cpProtocolInfo for each layer protocol supported.

8.3.3.13 VnffgInfo information element

8.3.3.13.1 Description

This information element contains information about a VNFFG instance.

8.3.3.13.2 Attributes

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

Table 8.3.3.13.2-1: Attributes of the VnffgInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnffgId	M	1	Identifier	Identifier of the Vnffg information element.
vnffgdId	M	1	Identifier (Reference to Vnffgd)	Identifier of the VNFFGD used to instantiate this VNFFG.
vnfId	M	1..N	Identifier (Reference to VnfInfo)	Identifier(s) of the constituent VNF instance(s) of the VNFFG.
pnfId	M	0..N	Identifier (Reference to PnfInfo)	Identifier(s) of the constituent PNF instance(s) of the VNFFG.
virtualLinkId	M	1..N	Identifier (Reference to NsVirtualLinkInfo)	Identifier(s) of the constituent VL instance(s) of the VNFFG.
cpId	M	1..N	Identifier (Reference to VnfExtCpInfo or PnfExtCpInfo or SapInfo)	Identifiers of the CP instances attached to the constituent VNFs and PNFs or the sap instances of the VNFFG (see note).
nfpInfo	M	1..N	NfpInfo	Information on the NFPs of this VNFFG.
NOTE: It indicates an exhaustive list of all the CP instances and SAP instances of the VNFFG.				

8.3.3.14 PnfExtCpInfo information element

8.3.3.14.1 Description

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

8.3.3.14.2 Attributes

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

Table 8.3.3.14.2-1: Attributes of the PnfExtCpInfo information element

Attribute	Qualifier	Cardinality	Content	Description
cpInstanceId	M	1	Identifier	Identifier of this external CP instance and of this PnfExtCpInfo information element.
cpdId	M	1	Identifier (Reference to Cpd)	Identifier of (reference to) the Connection Point Descriptor (CPD) for this CP.
cpProtocolInfo	M	1..N	CpProtocolInfo	Protocol information for this CP. There shall be one cpProtocolInfo for each layer protocol supported.

8.3.3.15 NfpInfo information element

8.3.3.15.1 Description

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

8.3.3.15.2 Attributes

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

Table 8.3.3.15.2-1: Attributes of the NfpInfo information element

Attribute	Qualifier	Cardinality	Content	Description
nfpId	M	1	Identifier	Identifier of this Nfp information element.
nfpdId	M	0..1	Identifier (Reference to Nfpd)	Identifier of the NFPD used to instantiate this NFP.
nfpName	M	0..1	String	Human readable name for the NFP.
description	M	0..1	String	Human readable description for the NFP.
cpGroup	M	1..N	CpGroupInfo	Group(s) of CPs and/or SAPs which the NFP passes through. See note.
totalCp	O	0..1	Integer	Total number of CPs in this NFP.
nfpRule	M	1	NfpRule	NFP classification and selection rule.
nfpState	M	1	Enum	An indication of whether the NFP instance is enabled or disabled. VALUES: • ENABLED • DISABLED
NOTE: When multiple identifiers are included, the position of the identifier in the information element value specifies the position of the group in the path.				

8.3.3.16 NsScaleInfo information element

8.3.3.16.1 Description

This information element provides information about an NS scaling aspect.

8.3.3.16.2 Attributes

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

Table 8.3.3.16.2-1: Attributes of the NsScaleInfo information element

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

8.3.3.17 VnfExtCpInfo information element

8.3.3.17.1 Description

This information element provides information related to an external CP.

8.3.3.17.2 Attributes

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

Table 8.3.3.17.2-1: Attributes of the VnfExtCpInfo information element

Attribute	Qualifier	Cardinality	Content	Description
cpInstanceId	M	1	Identifier	Identifier of this external CP instance and of this VnfExtCpInfo information element.
cpdId	M	1	Identifier (Reference to VnfExtCpd)	Identifier of the external CPD, vnfExtCpd in the VNFD.
vnfdId	M	0..1	Identifier (Reference to Vnfd)	Identifier of the VNFD. Shall be present in case the value differs from the vnfdId attribute of the VNF instance (e.g. during a "Change current VNF package" operation or due to its final failure).
cpProtocolInfo	M	0..N	CpProtocolInfo	Protocol information for this CP. There shall be one cpProtocolInfo for each layer protocol supported.
associatedVnfcCpId	M	0..1	Identifier (Reference to VnfcCpInfo)	Identifier of the VnfcCp that is exposed as this VnfExtCp instance, either directly or via a floating IP address. Shall be present if the cpdId of this VnfExtCp has an intCpd attribute. See note 1.
associatedVipCpId	M	0..1	Identifier (Reference to VipCpInfo)	Identifier of the VIP CP that is exposed as this VnfExtCp instance, either directly or via a floating IP address. Shall be present if the cpdId of this VnfExtCp has a vipCpd attribute. See note 1.
associatedVnfVirtualLinkId	M	0..1	Identifier (Reference to VnfVirtualLinkResourceInfo)	Identifier of the Vnf VL that this VnfExtCP maps to. Shall be present if the cpdId of this VnfExtCp has an intVirtualLinkDesc attribute. See note 1.
extLinkPortId	M	0..1	Identifier (Reference to ExtLinkPortInfo)	Identifier of the "ExtLinkPortInfo" information element in the "ExtVirtualLinkInfo" information element. Shall be present if the CP is associated to a link port. See note 2.
metadata	M	0..N	KeyValuePair	Metadata about this external CP.
NOTE 1: The attributes associatedVnfcCpId, associatedVipCpId and associatedVnfVirtualLinkId are mutually exclusive. Exactly one shall be present. NOTE 2: An external CP is not associated to a link port in the cases indicated for the "extLinkPorts" attribute in clause 8.12.2.2 of ETSI GS NFV-IFA 007 [i.5].				

8.3.3.18 ExtVirtualLinkInfo information element

8.3.3.18.1 Description

This information element provides a reference to an external VL.

8.3.3.18.2 Attributes

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

Table 8.3.3.18.2-1: Attributes of the ExtVirtualLinkInfo information element

Attribute	Qualifier	Cardinality	Content	Description
extVirtualLinkId	M	1	Identifier	Identifier of this external VL. The identifier is assigned by the NFV-MANO entity that manages this VL instance.
resourceHandle	M	1	ResourceHandle	Reference to the resource realizing this VL.
extLinkPort	M	0..N	ExtLinkPortInfo	Link ports of this VL.

8.3.3.19 ExtManagedVirtualLinkInfo information element

8.3.3.19.1 Description

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

8.3.3.19.2 Attributes

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

Table 8.3.3.19.2-1: Attributes of the ExtManagedVirtualLinkInfo information element

Attribute	Qualifier	Cardinality	Content	Description
extManagedVirtualLinkId	M	1	Identifier	Identifier of this externally-managed internal VL. The identifier is assigned by the NFV-MANO entity that manages this VL instance.
vnfVirtualLinkDescId	M	1	Identifier (Reference to VnfVirtualLinkDesc)	Identifier of the VNF Virtual Link Descriptor (VLD) in the VNFD.
vnfdId	M	0..1	Identifier (Reference to Vnfd)	Identifier of the VNFD. Shall be present in case the value differs from the vnfdId attribute of the VNF instance (e.g. during a "Change current VNF package" operation or due to its final failure).
networkResource	M	1	ResourceHandle	Reference to the VirtualNetwork resource or multi-site connectivity service providing this VL.
vnfLinkPort	M	0..N	VnfLinkPortInfo	Link ports of this VL.
extManagedMultisiteVirtualLinkId	M	0..1	Identifier	Identifier of the externally-managed multi-site VL instance. The identifier is assigned by the NFV-MANO entity that manages the externally managed multi-site VL instance. It shall be present when the externally-managed internal VL is part of a multi-site VL, e.g. in support of multi-site VNF spanning several VIMs. All externally-managed internal VL instances corresponding to a an internal VL created based on the same virtualLinkDescId shall refer to the same extManagedMultisiteVirtualLinkId.

8.3.3.20 VnfLinkPortInfo information element

8.3.3.20.1 Description

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

8.3.3.20.2 Attributes

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

Table 8.3.3.20.2-1: Attributes of the VnfLinkPortInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfLinkPortId	M	1	Identifier	Identifier of this link port as provided by the entity that has created the link port.
resourceHandle	M	1	ResourceHandle	Reference to the virtualised resource realizing this link port.
associatedExtCpId	M	0..1	Identifier (Reference to VnfExtCpInfo)	External CP of the VNF associated to this link port. When an external CP is associated to an internal VL, this attribute reflects this association. Shall be present when the link port is used for external connectivity by the VNF. See notes 1, 2, 3 and 4.
vnfcCpInstanceId	M	0..1	Identifier (Reference to VnfcCpInfo)	VNFC CP of the VNF connected to this link port. May be present. See notes 1, 3 and 4.
vipCpInstanceId	M	0..1	Identifier (Reference to VipCpInfo)	VIP CP instance of the VNF connected to this link port. May be present. See notes 1, 3, 4 and 5.

NOTE 1: There shall be at most one link port associated with any external connection point instance or internal connection point (i.e. VNFC CP) instance or VIP CP instance.
NOTE 2: A VnfLinkPort does not terminate on an external CP, as external CPs are connected to external VLs.
NOTE 3: Either associatedExtCpId or any combination of vnfcCpInstanceId and vipCpInstanceId (i.e. one or both of them) shall be present for a VnfLinkPortInfo. In case both vnfcCpInstanceId and vipCpInstanceId are present, the two different CP instances share the linkport.
NOTE 4: The attributes "associatedExtCpId" and "vnfcCpInstanceId" model two separate associations in the information model. It is part of the protocol design to define the representation of these associations.
NOTE 5: Clause A.4 of ETSI GS NFV-IFA 007 [i.5] provides examples for configurations where both vipCpInstanceId and vnfcCpInstanceId are present (UC#5 and UC#5-b), only vnfcCpInstanceId is present (UC#2), or only vipCpInstanceId is present (UC6 and UC#6-b).

8.3.3.21 ScaleInfo information element

8.3.3.21.1 Description

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

8.3.3.21.2 Attributes

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

Table 8.3.3.21.2-1: Attributes of the ScaleInfo information element

Attribute	Qualifier	Cardinality	Content	Description
aspectId	M	1	Identifier (Reference to ScalingAspect)	Reference to the scaling aspect.
vnfdId	M	0..1	Identifier (Reference to Vnfd)	Identifier of the VNFD. Shall be present in case the value differs from the vnfdId attribute of the VNF instance (e.g. during a "Change current VNF package" operation or due to its final failure).
scaleLevel	M	1	Integer	The scale level for that aspect. Minimum value 0, maximum value maxScaleLevel as declared in the VNFD (see ETSI GS NFV-IFA 011 [2], clause 7.1.10.2.2).

8.3.3.22 ExtLinkPortInfo information element

8.3.3.22.1 Description

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

8.3.3.22.2 Attributes

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

Table 8.3.3.22.2-1: Attributes of the ExtLinkPortInfo information element

Attribute	Qualifier	Cardinality	Content	Description
extLinkPortId	M	1	Identifier	Identifier of this link port as provided by the entity that has created the link port.
resourceHandle	M	1	ResourceHandle	Reference to the virtualised resource realizing this link port.
cpInstanceId	M	0..1	Identifier (Reference to VnfExtCpInfo)	External CP of the VNF connected to this link port. See note 1.
secondaryCpInstanceId	M	0..1	Identifier (Reference to VnfExtCpInfo)	Additional external CP of the VNF connected to this link port. If present, this attribute shall refer to a "secondary" ExtCpInfo item in the VNF instance that exposes a virtual IP CP instance which shares this linkport with the external CP instance referenced by the "cpInstanceId" attribute. See note 1 and note 2.
NOTE 1: There shall be at most one link port associated with any external connection point instance. NOTE 2: The use cases UC#4 and UC#5 in clause A.4 of ETSI GS NFV-IFA 007 [i.5] provide examples for such a configuration.				

8.3.3.23 VnfcCpInfo information element

8.3.3.23.1 Description

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

8.3.3.23.2 Attributes

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

Table 8.3.3.23.2-1: Attributes of the VnfcCpInfo information element

Attribute	Qualifier	Cardinality	Content	Description
cpInstanceId	M	1	Identifier	Identifier of this VnfcCpInfo information element.
cpdId	M	1	Identifier (Reference to VduCpd)	Identifier of the VDU CPD, cpdId, in the VNFD.
vnfExtCpId	M	0..1	Identifier (Reference to VnfExtCpInfo)	When the VNFC CP is exposed as external CP of the VNF, the identifier of this external VNF CP.
cpProtocolInfo	M	0..N	CpProtocolInfo	Protocol information for this CP. There shall be one cpProtocolInfo for each layer protocol supported.
vnfLinkPortId	M	0..1	Identifier (Reference to VnfLinkPortInfo)	Identifier of the "VnfLinkPortInfo" information element in the "VnfVirtualLinkResourceInfo" information element. Shall be present if the CP is associated to a link port.
metadata	M	0..N	KeyValuePair	Metadata about this VNFC CP.

8.3.3.24 CpProtocolInfo information element

8.3.3.24.1 Description

This information element describes and associates the protocol layer that a CP uses together with other protocol-related information, like addresses.

8.3.3.24.2 Attributes

The CpProtocolInfo information element shall follow the indications provided in table 8.3.3.24.2-1.

Table 8.3.3.24.2-1: Attributes of the CpProtocolInfo information element

Attribute	Qualifier	Cardinality	Content	Description
layerProtocol	M	1	Enum	Identifies which protocol the CP uses for connectivity purposes. See note 1. VALUES: • IP_OVER_ETHERNET • etc.
address	M	1..N	Not specified.	For a SAP or a PnfExtCp, address of this CP. For a VnfExtCp or a VnfcCp, list of network addresses that have been configured (statically or dynamically) on the link port that connects the CP to a VL. See notes 2 and 3.
NOTE 1: The layerProtocol values shall be compatible with the ones defined in the CPD. NOTE 2: The address information shall be compatible with the layerProtocol attribute. NOTE 3: For a SAP or a PnfExtCp, only a single address shall be provided. For a SAP, in some cases, the NFVO provides the address.				

8.3.3.25 CpGroupInfo information element

8.3.3.25.1 Description

This information element describes a group of CPs and/or SAPs pairs associated to the same position in an NFP.

8.3.3.25.2 Attributes

The attributes of the CpGroupInfo information element shall follow the indications provided in table 8.3.3.25.2-1.

Table 8.3.3.25.2-1: Attributes of the CpGroupInfo information element

Attribute	Qualifier	Cardinality	Content	Description
cpPairInfo	M	1..N	CpPairInfo	One or more pair(s) of ingress and egress CPs or SAPs which the NFP passes by. See note.
forwardingBehaviour	M	0..1	Enum	Identifies a rule to apply to forward traffic to the ingress CPs or SAPs of the group. VALUES: • ALL: Traffic flows shall be forwarded simultaneously to all CPs or SAPs of the group. • LB: Traffic flows shall be forwarded to one CP or SAP of the group selected based on a load-balancing algorithm. • Etc.
forwardingBehaviourInputParameters	M	0..1	Not specified	Provides input parameters to configure the forwarding behaviour (e.g. identifies a load balancing algorithm and criteria).
NOTE: All CP or SAP pairs in a group shall be instantiated from connection point descriptors or service access point descriptors referenced in the corresponding NfpPositionDesc (see ETSI GS NFV-IFA 014 [3]).				

8.3.3.26 CpPairInfo information element

8.3.3.26.1 Description

This information element describes a pair of ingress and egress CPs or SAPs which the NFP passes by.

8.3.3.26.2 Attributes

The attributes of the CpPairInfo information element shall follow the indications provided in table 8.3.3.26.2-1.

Table 8.3.3.26.2-1: Attributes of the CpPairInfo information element

Attribute	Qualifier	Cardinality	Content	Description
cpInfo	M	1..2	Identifier (Reference to VnfExtCpInfo or PnfExtCpInfo or SapInfo)	Identifier(s) of the CPs and/or SAPs which form the pair. See note.
NOTE: The presence of a single cpInfo occurrence indicates that the CP or SAP is used both as an ingress and egress port at a particular NFP position.				

8.3.3.27 VnfSnapshotInfo information element

8.3.3.27.1 Description

This information element provides the details of a VNF Snapshot of a VNF instance which is part of an NS.

8.3.3.27.2 Attributes

The VnfSnapshotInfo information element shall follow the indications provided in table 8.3.3.27.2-1.

Table 8.3.3.27.2-1: Attributes of the VnfSnapshotInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfSnapshotInfoId	M	1	Identifier	Identifier of information held by the VNFM about a specific VNF Snapshot. This identifier was allocated by the VNFM.
triggeredAt	M	1	DateTime	Timestamp indicating when the VNF Snapshot creation has been started.
createdAt	M	0..1	DateTime	Timestamp indicating when the VNF Snapshot creation has been completed. Cardinality is 0 when the VNF Snapshot creation has not yet completed and shall be 1 afterwards. See note 1.
vnfInstanceId	M	1	Identifier	Identifier of the snapshotted VNF instance.
vnfdId	M	1	Identifier (Reference to Vnfd)	References the VNFD in use at the time the snapshot of the VNF instance has been created. See notes 2 and 3.
vnfInfo	M	1	VnfInfo	VnfInfo of the snapshotted VNF instance.
vnfcSnapshotInfo	M	1..N	VnfcSnapshotInfo	Information about VNFC Snapshots constituting this VNF Snapshot.
userDefinedData	O	0..N	KeyValuePair	User defined data for the VNF Snapshot.

NOTE 1: On the Os-Ma-nfvo reference point, the cardinality 0 is not used.
NOTE 2: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way.
NOTE 3: This information is copied from the VNFD of the on-boarded VNF Package which was used to instantiate the VNF instance.

8.3.3.28 VnfcSnapshotInfo information element

8.3.3.28.1 Description

This information element provides the details of a VNFC Snapshot of a VNFC instance which is part of VNF instance of an NS.

8.3.3.28.2 Attributes

The VnfcSnapshotInfo information element shall follow the indications provided in table 8.3.3.28.2-1.

Table 8.3.3.28.2-1: Attributes of the VnfcSnapshotInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfcSnapshotInfoId	M	1	Identifier	Identifier of information held by the VNFM about a specific VNFC Snapshot. This identifier was allocated by the VNFM.
triggeredAt	M	1	DateTime	Timestamp indicating when the VNF Snapshot creation has been started.
createdAt	M	0..1	DateTime	Timestamp indicating when the VNFC Snapshot creation has been completed. Cardinality is 0 when the VNF Snapshot creation has not yet completed and shall be 1 afterwards. See note 2.
vnfcInstanceId	M	1	Identifier	Identifier of the snapshotted VNFC instance.
vnfcInfoId	M	1	Identifier (Reference to VnfcResourceInfo)	Reference to the information about the snapshotted VNFC instance.
computeSnapshotResource	M	0..1	ResourceHandle	Reference to a compute snapshot resource. See note 1.
storageSnapshotResource	M	0..N	StorageSnapshotResource	Mapping of the storage resources associated to the VNFC with the storage snapshot resources.

Attribute	Qualifier	Cardinality	Content	Description
userDefinedData	O	0..N	KeyValuePair	User defined data for the VNFC Snapshot.
NOTE 1: The identifier of the compute snapshot resource is assigned during creation of a VNFC Snapshot being returned from the VIM as output data in the response message of the individual resource operations. This attribute shall only be present for a VNFC snapshot that has been newly created by the VNFM as a result of the "Create Snapshot operation". NOTE 2: On the Os-Ma-nfvo reference point, the cardinality 0 is not used.				

8.3.3.29 WanConnectionInfo information element

8.3.3.29.1 Description

This information element provides information about the connectivity to the WAN of network resources realizing a VL, e.g. when the VL is deployed on several sites across a WAN.

8.3.3.29.2 Attributes

The WanConnectionInfo information element shall follow the indications provided in table 8.3.3.29.2-1.

Table 8.3.3.29.2-1: Attributes of the WanConnectionInfo information element

Attribute	Qualifier	Cardinality	Content	Description
wanConnectionInfoId	M	1	Identifier	Identifier of this WAN connection information.
virtualLinkInstanceId	M	0..1	Identifier (Reference to NsVirtualLinkInfo or VnfVirtualLinkResourceInfo or ExtManagedVirtualLinkInfo)	References the VL instance to which the connection information is associated. Shall be present if the corresponding VL instance has been created.
protocolData	M	0..1	Not specified	Protocol specific information for connecting to the WAN.

8.3.3.30 StorageSnapshotResource information element

8.3.3.30.1 Description

This information element provides a mapping of the storage resources associated to the VNFC with the storage snapshot resources.

8.3.3.30.2 Attributes

The StorageSnapshotResource information element shall follow the indications provided in table 8.3.3.30.2-1.

Table 8.3.3.30.2-1: Attributes of the StorageSnapshotResource information element

Attribute	Qualifier	Cardinality	Content	Description
storageResourceId	M	1	Identifier (Reference to VirtualStorageResourceInfo)	Reference to a virtual storage resource.
storageSnapshotResource	M	0..1	ResourceHandle	Reference to a storage snapshot resource. See note.
NOTE: The identifier of the storage snapshot resource is assigned during creation of a VNFC Snapshot being returned from the VIM as output data in the response message of the individual resource operations. This attribute shall only be present for a VNFC snapshot with associated storage resources and that has been newly created by the VNFM as a result of the "Create Snapshot operation".				

8.3.3.31 TrunkPortsInfo information element

8.3.3.31.1 Description

The information element provides runtime information of a collection of CPs of the VNFC instance which has one CP working in trunk mode, as parent port of a trunk, and other CPs working as subports of the same trunk.

8.3.3.31.2 Attributes

The attributes of the TrunkPortsInfo information element shall follow the indications provided in table 8.3.3.31.2-1.

Table 8.3.3.31.2-1: Attributes of the TrunkPortsInfo information element

Attribute	Qualifier	Cardinality	Content	Description
parentPort	M	1	Identifier (Reference to VnfcCpInfo)	Reference to the CP instance which is used as parent port in the trunk.
subportList	M	0..N	Identifier (Reference to VnfcCpInfo)	Reference to the CP instance(s) working as subport(s) in the trunk.

8.3.3.32 VipCpInfo information element

8.3.3.32.1 Description

This information element provides information related to VIP CP.

8.3.3.32.2 Attributes

The VipCpInfo information element shall follow the indications provided in table 8.3.3.32.2-1.

Table 8.3.3.32.2-1: Attributes of the VipCpInfo information element

Attribute	Qualifier	Cardinality	Content	Description
cpInstanceId	M	1	Identifier	Identifier of this VIP CP instance and of this VipCpInfo information element.
cpdId	M	1	Identifier (Reference to VipCpd)	Identifier of the VIP Connection Point Descriptor, VipCpd, in the VNFD.
vnfExtCpId	M	0..1	Identifier (Reference to VnfExtCpInfo)	When the VIP CP is exposed as external CP of the VNF, the identifier of this external VNF CP instance.
cpProtocolInfo	M	0..N	CpProtocolInfo	Protocol information for this CP. There shall be one cpProtocolInfo for layer 3. There may be one cpProtocolInfo for layer 2.
associatedVnfcCpId	M	0..N	Identifier (Reference to VnfcCpInfo)	Identifiers of the VnfcCps that share the virtual IP address allocated to the VIP CP instance. See note.
vnfLinkPortId	M	0..1	Identifier (Reference to VnfLinkPortInfo)	Identifier of the "VnfLinkPortInfo" information element in the "VnfVirtualLinkResourceInfo" information element. Shall be present if the CP is associated to a link port in an internal VL.
metadata	M	0..N	KeyValuePair	Metadata about this VIP CP.
NOTE: It is possible that there is no associated VnfcCp because the VIP CP is available but not associated yet.				

8.3.3.33 McioInfo information element

8.3.3.33.1 Description

This information element provides information about MCIOs representing VNFC instances realized by one or a set of OS containers which have been created based on the same VDU.

Within the CISM, MCIO controllers monitor the actual state of MCIOs representing VNFC instances realized by one or a set of OS containers and compare it to the desired state as specified in the respective declarative descriptor. They trigger actions toward the CIS to align the actual to the desired state. Monitoring the actual state includes monitoring the number of MCIO instances available at any specific point in time. In addition, MCIO controllers maintain properties and runtime information on the MCIOs which have been created based on the same VDU. The McioInfo information element provides the runtime information on the MCIOs obtained from the respective MCIO controllers.

Runtime information of the set of OS containers realizing an individual VNFC instances is not part of the McioInfo information element; such runtime information is provided in the ResourceHandle information element referenced from the VnfcResourceInfo. The McioInfo does not provide runtime information of a constituent VNFC instance created based on a specific VDU.

8.3.3.33.2 Attributes

The McioInfo information element shall follow the indications provided in table 8.3.3.33.2-1.

Table 8.3.3.33.2-1: Attributes of the McioInfo information element

Attribute	Qualifier	Cardinality	Content	Description
mcioId	M	1	Identifier	Identifier of this MCIO, created by the CISM.
mcioName	M	1	String	Human readable name of this MCIO.
vduId	M	1	Identifier (Reference to Vdu)	Reference to the applicable Vdu information element in the VNFD.
cismId	M	1	Identifier	Identifier of the CISM managing this MCIO.
mcioType	M	1	Not specified	The type of MCIO. See note 1.
desiredInstances	M	1	Integer	Number of desired MCIO instances.
availableInstances	M	1	Integer	Number of available MCIO instances.
additionalInfo	M	0..1	Not Specified	Additional information which is specific to the MCIO, its type, and which is available from the CISM. See note 2.
NOTE 1: The type of MCIO as specified in the declarative descriptor of the MCIO, and that can be read from the CISM. EXAMPLE: In case of MCIOs managed by Kubernetes®, the type of MCIO corresponds to the "kind" property of the declarative descriptor. NOTE 2: If the attribute additionalInfo is present, it may contain runtime information on the actual and desired state of the MCIO(s).				

8.3.4 Information elements related to NS Lifecycle Management operations

8.3.4.1 Introduction

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

8.3.4.2 SapData information element

8.3.4.2.1 Description

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

8.3.4.2.2 Attributes

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

Table 8.3.4.2.2-1: Attributes of the SapData information element

Attribute	Qualifier	Cardinality	Content	Description
sapdId	M	1	Identifier (Reference to Sapd)	Reference to the SAPD for this SAP.
sapName	M	1	String	Human readable name for the SAP.
description	M	1	String	Human readable description for the SAP.
address	M	0..N	Not specified	Address for this SAP, including the information on applicable layer protocol(s). In some cases, the NFVO provides the address (refer to attribute sapAddressAssignment of Sapd information element in ETSI GS NFV-IFA 014 [3], clause 6.2.3.2). See note.
NOTE: The address information shall be compatible with the layerProtocol values defined in the CPD.				

8.3.4.3 VnfInstanceData information element

8.3.4.3.1 Description

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

8.3.4.3.2 Attributes

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

Table 8.3.4.3.2-1: Attributes of the VnfInstanceData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the existing VNF instance to be used in the NS.
vnfProfileId	M	0..1	Identifier (Reference to VnfProfile)	Identifier of (Reference to) a vnfProfile defined in the NSD which the existing VNF instance shall be matched with. If not present, the NFVO will select the VnfProfile matching the information in the VNF instance. See note 1.
overridingVnfdId	M	0..1	Identifier (Reference to a Vnfd)	It replaces the vnfdId indicated in the vnfProfileId. The VNFD referenced by this attribute shall have the same vnfdExtInvariantId as the one indicated in the VnfProfile. If the VnfProfile does not contain a vndExtInvariantId this attribute shall be ignored. If the VNFD of the existing VNF instance referenced by vnfInstanceId does not match the VNFD indicated by this attribute the VNF instance shall not be added to the NS instance. See notes 1 and 2.
NOTE 1: If the overridingVnfdId attribute is present the vnfProfileId attribute shall also be present. NOTE 2: This attribute allows to use an existing VNF instance based on a different VNFD to the one specified in the NSD with vnfProfileId, provided both have the same vnfdExtInvariantId.				

8.3.4.4 VnfLocationConstraint information element

8.3.4.4.1 Description

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

8.3.4.4.2 Attributes

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

Table 8.3.4.4.2-1: Attributes of the VnfLocationConstraint information element

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

8.3.4.5 ParamsForVnf information element

8.3.4.5.1 Description

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

8.3.4.5.2 Attributes

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

Table 8.3.4.5.2-1: Attributes of the ParamsForVnf information element

Attribute	Qualifier	Cardinality	Content	Description
vnfProfileId	M	1	Identifier (Reference to VnfProfile)	Identifier of (reference to) a vnfProfile to which the additional parameters apply.
additionalParam	M	0..N	KeyValuePair	Additional parameters that are to be applied per VNF instance.
overridingVnfdId	M	0..1	Identifier (Reference to a Vnfd)	If present it replaces the vnfdId indicated in the referenced VnfProfile at the VNF instantiation. The VNFD referenced by this attribute shall have the same vnfdExtInvariantId as the one indicated in the VnfProfile. If the VnfProfile does not contain a vnfdExtInvariantId this attribute shall be ignored. See note.
NOTE: This attribute allows at VNF instantiation the use of a VNFD different from the one specified in the NSD with vnfProfileId provided the two VNFDs refer to the same vnfdExtInvariantId.				

8.3.4.6 ScaleNsData information element

8.3.4.6.1 Description

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

8.3.4.6.2 Attributes

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

Table 8.3.4.6.2-1: Attributes of the ScaleNsData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceToBeAdded	M	0..N	VnfInstanceData	Specifies an existing VNF instance to be added to the NS instance as part of the scaling operation. If needed, the VNF Profile to be used for this VNF instance is also provided. See notes 1, 2 and 3.
vnfInstanceToBeRemoved	M	0..N	Identifier	Specifies a VNF instance to be removed from the NS instance as part of the scaling operation. See notes 1 and 4.
scaleNsBystepsData	M	0..1	ScaleNsBystepsData	Specifies the information needed to scale an NS instance by one or more scaling steps. See note 1.
scaleNsToLevelData	M	0..1	ScaleNsToLevelData	Specifies the information needed to scale an NS instance to a target size. See note 1.
additionalParamForNs	M	0..N	KeyValuePair	Allows the OSS/BSS to provide additional parameter(s) at the NS level necessary for the NS scaling (as opposed to the VNF level, which is covered in additionalParamForVnf).
additionalParamForVnf	M	0..N	ParamsForVnf	Allows the OSS/BSS to provide additional parameter(s) per VNF instance (as opposed to the NS level, which is covered in additionalParamforNs). This is for VNFs that are to be created by the NFVO as part of the NS scaling and not for existing VNF that are covered by the scaleVnfData.
locationConstraints	M	0..N	VnfLocationConstraint	Defines the location constraints for the VNF to be instantiated as part of the NS scaling. An example can be a constraint for the VNF to be in a specific geographic location.
nestedNsLocationConstraints	M	0..N	NestedNsLocationConstraint	Defines the location constraints for the nested NS to be instantiated as part of the NS instantiation. An example can be a constraint for the nested NS to be in a specific geographic location.

NOTE 1: No more than two attributes between vnfInstanceToBeAdded, vnfInstanceToBeRemoved, scaleNsBystepsData and scaleNsToLevelData shall be present. In case of two, the attributes shall be vnfInstanceToBeAdded and vnfInstanceToBeRemoved.
NOTE 2: The DF of the VNF instance shall match the VNF DF present in the associated VNF Profile of the new NS flavour.
NOTE 3: This functionality is the same as the one provided by the Update NS operation when the AddVnf update type is selected (see clause 7.3.5).
NOTE 4: This functionality is the same as the one provided by the Update NS operation when the RemoveVnf update type is selected (see clause 7.3.5).

8.3.4.7 ScaleNsBystepsData information element

8.3.4.7.1 Description

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

8.3.4.7.2 Attributes

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

Table 8.3.4.7.2-1: Attributes of the ScaleNsByStepsData information element

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

8.3.4.8 ScaleNsToLevelData information element

8.3.4.8.1 Description

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

8.3.4.8.2 Attributes

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

Table 8.3.4.8.2-1: Attributes of the ScaleNsToLevelData information element

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

8.3.4.9 ScaleVnfData information element

8.3.4.9.1 Description

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

8.3.4.9.2 Attributes

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

Table 8.3.4.9.2-1: Attributes of the ScaleVnfData information element

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

8.3.4.10 ScaleToLevelData information element

8.3.4.10.1 Description

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

8.3.4.10.2 Attributes

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

Table 8.3.4.10.2-1: Attributes of the ScaleToLevelData information element

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

8.3.4.11 ScaleByStepData information element

8.3.4.11.1 Description

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

8.3.4.11.2 Attributes

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

Table 8.3.4.11.2-1: Attributes of the ScaleByStepData information element

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

8.3.4.12 InstantiateVnfData information element

8.3.4.12.1 Description

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

8.3.4.12.2 Attributes

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

Table 8.3.4.12.2-1: Attributes of the InstantiateVnfData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfdId	M	0..1	Identifier (Reference to Vnfd)	Information sufficient to identify the VNFD which defines the VNF to be instantiated. See notes 4 and 5.
flavourId	M	0..1	Identifier (Reference to VnfDf)	Identifier of the VNF DF to be instantiated. See notes 1, 4 and 5.
instantiationLevelId	M	0..1	Identifier (Reference to InstantiationLevel)	Identifier of the instantiation level of the DF to be instantiated. See notes 5 and 6.

Attribute	Qualifier	Cardinality	Content	Description
targetScaleLevelInfo	M	0..N	VnfScaleInfo	This attribute is applicable if VNF supports target scale level instantiation. For each scaling aspect of the current deployment flavour, the attribute specifies the scale level of VNF constituents (e.g. VDU level) to be instantiated. See notes 6 and 7.
vnfProfileId	M	0..1	Identifier (Reference to VnfProfile)	Identifier of (Reference to) a vnfProfile defined in the NSD which is used for instantiating the VNF. See notes 5 and 8.
vnfInstanceName	M	0..1	String	Human-readable name of the VNF instance to be created.
vnfInstanceDescription	M	0..1	String	Human-readable description of the VNF instance to be created.
extVirtualLink	M	0..N	ExtVirtualLinkData	Information about external VLs to connect the VNF to.
extManagedVirtualLink	M	0..N	ExtManagedVirtualLinkData	Information about internal VLs that are managed by other entities than the VNFM. See notes 2 and 3.
localizationLanguage	M	0..1	Not specified	Localization language of the VNF to be instantiated. The localization languages supported by a VNF can be declared in the VNFD. If this parameter is not provided and the "defaultLocalizationLanguage" attribute is declared in the VNFD, the "defaultLocalizationLanguage" shall be used to determine the localization language VNF to be instantiated.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the instantiation process, specific to the VNF being instantiated, as declared in the VNFD (see clause 7.1.5.3 in ETSI GS NFV-IFA 011 [2]).
locationConstraint	M	0..1	VnfLocationConstraint	Defines the location constraints for the VNF to be instantiated as part of the NS Update. An example can be a constraint for the VNF to be in a specific geographic location.
metadata	M	0..N	KeyValuePair	This parameter provides values for the "metadata" input parameter of the Create VNF Identifier operation for the VNF to be instantiated as part of the NS update.
extension	M	0..N	KeyValuePair	This parameter provides values for the "extension" input parameter of the Instantiate VNF operation.
overridingVnfdId	M	0..1	Identifier (Reference to Vnfd)	If present the vnfProfileId shall also be present. It replaces the vnfdId indicated in the vnfProfileId. The VNFD referenced by this attribute shall have the same vnfdExtInvariantId as the one indicated in the VnfProfile. If the VnfProfile does not contain a vndExtInvariantId this attribute shall be ignored. See notes 8 and 9.

Attribute	Qualifier	Cardinality	Content	Description
NOTE 1: The VnfDf information element is defined in ETSI GS NFV-IFA 011 [2], clause 7.1.8.2.				
NOTE 2: The indication of externally-managed internal VLs is needed in case networks have been pre-configured for use with certain VNFs, for instance to ensure that these networks have certain properties such as security or acceleration features, or to address particular network topologies. The present document assumes that externally-managed internal VLs are managed by the NFVO and created towards the VIM as supported by the virtualised network resource management interface specified in ETSI GS NFV-IFA 005 [i.4].				
NOTE 3: It is possible to have several ExtManagedVirtualLinkData for the same VNF internal VL in case of a multi-site VNF spanning several VIMs. The set of ExtManagedVirtualLinkData corresponding to the same VNF internal VL shall indicate so by referencing to the same VnfVirtualLinkDesc and externally-managed multi-site VL instance (refer to clause 8.3.4.28).				
NOTE 4: If vnfdId and flavourId (and InstantiationLevelId, if provided) are present, there should be only one vnfProfile that matches the vnfdId and flavourId (and InstantiationLevelId, if present) in the NS deployment flavour specified in the NSD associated to the NS instance to which the present operation is triggered. In the case there is more than one matching vnfProfile, the NFVO may select a matching vnfProfile based on other information, such as external VL.				
NOTE 5: Either the attribute triple "vnfdId, flavourId and instantiationLevelId (if provided)" or the attribute "vnfProfileId" shall be present, but not both.				
NOTE 6: The target size for VNF instantiation may be specified in either instantiationLevelId or targetScaleLevelInfo, but not both. If none of the two attributes (instantiationLevelId or targetScaleLevelInfo) are present, the default instantiation level as declared in the VNFD shall be used.				
NOTE 7: If targetScaleLevelInfo is specified, information provided in targetScaleLevelInfo shall be used for instantiating scalable constituents of the VNF (e.g. VDUs/VLs). For scaling aspects not specified in targetScaleLevelInfo or for the VNF constituents (e.g. VDUs/VLs) that are not scalable, the default instantiation level as declared in the VNFD shall be used for instantiation.				
NOTE 8: If the overridingVnfdId attribute is present the vnfProfileId attribute shall also be present.				
NOTE 9: This attribute allows at VNF instantiation the use of a VNFD different from the one specified in the NSD with vnfProfileId provided the two VNFDs have the same vnfdExtInvariantId.				

8.3.4.13 ExtVirtualLinkData information element

8.3.4.13.1 Description

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

8.3.4.13.2 Attributes

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

Table 8.3.4.13.2-1: Attributes of the ExtVirtualLinkData information element

Attribute	Qualifier	Cardinality	Content	Description
extVirtualLinkId	M	1	Identifier	Identifier of this external VL instance. The identifier is assigned by the NFV-MANO entity that manages this VL instance.
vimId	CM	0..1	Identifier	Identifier of the VIM that manages this resource. CONDITION: This attribute shall be supported and present if VNF-related resource management in direct mode is applicable.
resourceProviderId	CM	0..1	Identifier	Identifies the entity responsible for the management of the resource. CONDITION: This attribute shall be supported and present when VNF-related Resource Management in indirect mode is applicable.
resourceId	M	1	Identifier	Identifier of the resource in the scope of the VIM or the resource provider.
extCp	M	1..N	VnfExtCpData	External CPs of the VNF to be connected to this external VL.
extLinkPorts	M	0..N	ExtLinkPortData	Externally provided link ports to be used to connect external connection points to this external VL.

8.3.4.14 VnfExtCpData information element

8.3.4.14.1 Description

This information element provides input information related to one or more external CP instances created based on the same CPD.

8.3.4.14.2 Attributes

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

Table 8.3.4.14.2-1: Attributes of the VnfExtCpData information element

Attribute	Qualifier	Cardinality	Content	Description
cpdId	M	1	Identifier	Identifier of the CPD in the VNFD.
cpConfig	M	1..N	VnfExtCpConfig	List of instance data that need to be configured on the CP instances created from the respective CPD.

8.3.4.15 ChangeVnfFlavourData information element

8.3.4.15.1 Description

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

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

8.3.4.15.2 Attributes

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

Table 8.3.4.15.2-1: Attributes of the ChangeVnfFlavourData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance to be modified.
newFlavourId	M	1	Identifier (Reference to VnfDf)	Identifier of the new VNF DF to apply to this VNF instance. See note 3.
instantiationLevelId	M	0..1	Identifier (Reference to InstantiationLevel)	Identifier of the instantiation level of the DF to be used. See note 4.
targetScaleLevelInfo	M	0..N	VnfScaleInfo	This attribute is applicable if VNF supports target scale level instantiation. For each scaling aspect of the current deployment flavour, the attribute specifies the scale level of VNF constituents (e.g. VDU level) to be instantiated. See notes 4 and 5.
extVirtualLink	M	0..N	ExtVirtualLinkData	Information about external VLs to connect the VNF to.
extManagedVirtualLink	M	0..N	ExtManagedVirtualLinkData	Information about internal VLs that are managed by other entities than the VNFM. See notes 1 and 2.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the flavour change process, specific to the VNF being modified, as declared in the VNFD (see clause 7.1.5.9 in ETSI GS NFV-IFA 011 [2]).

Attribute	Qualifier	Cardinality	Content	Description
extension	M	0..N	KeyValuePair	This parameter provides values for the "extension" parameter of the ChangeVnfFlavour operation.
vnfConfigurableProperty	M	0..N	KeyValuePair	This parameter provides values for the "vnfConfigurableProperty" parameter of the ChangeVnfFlavour operation.

NOTE 1: The indication of externally-managed internal VLs is needed in case networks have been pre-configured for use with certain VNFs, for instance to ensure that these networks have certain properties such as security or acceleration features, or to address particular network topologies. The present document assumes that externally-managed internal VLs are managed by the NFVO and created towards the VIM as supported by the virtualised network resource management interface specified in ETSI GS NFV-IFA 005 [i.4].
NOTE 2: It is possible to have several ExtManagedVirtualLinkData for the same VNF internal VL in case of a multi-site VNF spanning several VIMs. The set of ExtManagedVirtualLinkData corresponding to the same VNF internal VL shall indicate so by referencing to the same VnfVirtualLinkDesc and externally-managed multi-site VL instance (refer to clause 8.3.4.28).
NOTE 3: The VnfDf information element is defined in ETSI GS NFV-IFA 011 [2], clause 7.1.8.2.
NOTE 4: The target size for VNF instantiation may be specified in either instantiationLevelId or targetScaleLevelInfo, but not both. If none of the two attributes (instantiationLevelId or targetScaleLevelInfo) are present, the default instantiation level as declared in the VNFD shall be used.
NOTE 5: If targetScaleLevelInfo is specified, information provided in targetScaleLevelInfo shall be used for instantiating scalable constituents of the VNF (e.g. VDUs/VLs). For scaling aspects not specified in targetScaleLevelInfo or for the VNF constituents (e.g. VDUs/VLs) that are not scalable, the default instantiation level as declared in the VNFD shall be used for instantiation.

8.3.4.16 OperateVnfData information element

8.3.4.16.1 Description

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

8.3.4.16.2 Attributes

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

Table 8.3.4.16.2-1: Attributes of the OperateVnfData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.
changeStateTo	M	1	Enum	The desired state to change the VNF to. VALUES: • STARTED • STOPPED
stopType	M	0..1	Enum	It signals whether forceful or graceful stop is requested. • VALUES: FORCEFUL: The VNF is stopped immediately. See note 2. • GRACEFUL: The VNFM first arranges to take the VNF out of service (by means out of scope of the present specification, e.g. involving interaction with EM, if required). Once this is successful, or after a timeout, the VNFM stops the VNF. Only applicable when changing state to stop.

Attribute	Qualifier	Cardinality	Content	Description
gracefulStopTimeout	M	0..1	TimeDuration	The time interval to wait for the VNF to be taken out of service during graceful stop, before stopping the VNF. If not given, it is expected that the VNFM waits for the successful taking out of service of the VNF, no matter how long it takes, before stopping the VNF. See note 1. Minimum timeout or timeout range are specified by the VNF vendor (e.g. defined in the VNFD or communicated by other means). The parameter is not relevant in case of forceful stop.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the Operate VNF operation, specific to the VNF being operated, as declared in the VNFD (see clause 7.1.5.8 in ETSI GS NFV-IFA 011 [2]).
NOTE 1: This implies that no VNF stop will be attempted if taking the VNF out of service fails or hangs. NOTE 2: If a VNF is stopped immediately and if the VNF is still in service, this may adversely impact network service. Therefore, operator policies apply to determine if forceful stop is allowed in the particular situation.				

8.3.4.17 ModifyVnfInfoData information element

8.3.4.17.1 Description

The ModifyVnfInfoData information element specifies for a VNF instance the information that is requested to be modified. The information to be modified shall comply with the associated NSD.

EXAMPLE: When the vnfdIdattribute value of VnfInfo needs to be updated, the value would need to match the identifier's value of a VnfPkgInfo whose vnfdId is present in the associated VNF type and profile of the NSD.

8.3.4.17.2 Attributes

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

Table 8.3.4.17.2-1: Attributes of the ModifyVnfInfoData information element

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

8.3.4.18 Void

8.3.4.19 AssocNewNsdVersionData information element

8.3.4.19.1 Description

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

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

8.3.4.19.2 Attributes

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

Table 8.3.4.19.2-1: Attributes of the AssocNewNsdVersionData information element

Attribute	Qualifier	Cardinality	Content	Description
newNsdId	M	1	Identifier (Reference to Nsd)	Identifier of the new NSD version that is to be associated to the NS instance.
sync	M	0..1	Boolean	Specify whether the NS instance should be automatically synchronized to the new NSD by the NFVO (in case of true value) or the NFVO should not do any action (in case of a false value) and wait for further guidance from OSS/BSS (i.e. waiting for OSS/BSS to issue NS lifecycle management operation to explicitly add/remove VNFs and modify information of VNF instances according to the new NSD). The synchronization to the new NSD means e.g. instantiating/adding those VNFs whose VNFD is referenced by the new NSD version but not referenced by the old one, terminating/removing those VNFs whose VNFD is referenced by the old NSD version but not referenced by the new NSD version, modifying information of VNF instances to the new applicable VNFD provided in the new NSD version (see note).
NOTE: The replacement of VNF instances, nested NS instances or PNF instances by instances whose descriptor is referenced by the new NSD is applicable even if the instance with a descriptor not referenced from the new NSD has the same VnfdExtInvariantId, NsdExtInvariantId or PnfdExtInvariantId as the new one, i.e. the synchronization procedure is based on descriptor identities of NSD constituents, and does not consider VnfdExtInvariantId, NsdExtInvariantId or PnfdExtInvariantId values.				

8.3.4.20 MoveVnfInstanceData information element

8.3.4.20.1 Description

The MoveVnfInstanceData specifies existing VNF instances that needs to be moved from one NS instance (source) to another NS instance (destination). The NS instance defined in the Update NS operation (refer to nsInstanceId in table 7.3.5.2-1) indicates the source NS instance and the destination NS instance is specified in the present information element (refer to targetNsInstanceId in table 8.3.4.20.2-1).

8.3.4.20.2 Attributes

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

Table 8.3.4.20.2-1: Attributes of the MoveVnfInstanceData information element

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

8.3.4.21 AddVnffgData information element

8.3.4.21.1 Description

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

8.3.4.21.2 Attributes

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

Table 8.3.4.21.2-1: Attributes of the AddVnffgData information element

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

8.3.4.22 UpdateVnffgData information element

8.3.4.22.1 Description

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

8.3.4.22.2 Attributes

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

Table 8.3.4.22.2-1: Attributes of the UpdateVnffgData information element

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

8.3.4.23 NfpData information element

8.3.4.23.1 Description

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

8.3.4.23.2 Attributes

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

Table 8.3.4.23.2-1: Attributes of the NfpData information element

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

8.3.4.24 HealNsData information element

8.3.4.24.1 Description

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

8.3.4.24.2 Attributes

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

Table 8.3.4.24.2-1: Attributes of the HealNsData information element

Attribute	Qualifier	Cardinality	Content	Description
degreeHealing	M	1	Enum	Indicates the degree of healing. VALUES: • HEAL_RESTORE: Complete the healing of the NS restoring the state of the NS before the failure occurred • HEAL_QOS: Complete the healing of the NS based on the newest QoS values • HEAL_RESET: Complete the healing of the NS resetting to the state original instantiation state of the NS • PARTIAL_HEALING
actionsHealing	M	0..N	String	Used to specify dedicated healing actions in a particular order (e.g. as a script). The actionsHealing can be used to provide a specific script whose content and actions might only be possible to be derived during runtime. See note.

Attribute	Qualifier	Cardinality	Content	Description
healScript	M	0..1	Identifier (Reference to LifeCycleManagementScript)	Reference to a script from the NSD that shall be used to execute dedicated healing actions in a particular order. The healScript, since it refers to a script in the NSD, can be used to execute healing actions which are defined during NS design time. See note.
additionalParamForNs	M	0..N	KeyValuePair	Allows the OSS/BSS to provide additional parameter(s) to the healing process at the NS level.
NOTE: Either the actionsHealing or healScript attribute shall be present, not both attributes.				

8.3.4.25 HealVnfData information element

8.3.4.25.1 Description

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

8.3.4.25.2 Attributes

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

Table 8.3.4.25.2-1: Attributes of the HealVnfData information element

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

8.3.4.26 AffinityOrAntiAffinityRule information element

8.3.4.26.1 Description

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

8.3.4.26.2 Attributes

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

Table 8.3.4.26.2-1: Attributes of the AffinityOrAntiAffinityRule information element

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

8.3.4.27 ChangeNsFlavourData information element

8.3.4.27.1 Description

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

8.3.4.27.2 Attributes

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

Table 8.3.4.27.2-1: ChangeNsFlavourData operation information element

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

8.3.4.28 ExtManagedVirtualLinkData information element

8.3.4.28.1 Description

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

8.3.4.28.2 Attributes

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

Table 8.3.4.28.2-1: Attributes of the ExtManagedVirtualLinkData information element

Attribute	Qualifier	Cardinality	Content	Description
extManagedVirtualLinkId	M	1	Identifier	Identifier of this externally-managed internal VL instance. The identifier is assigned by the NFV-MANO entity that manages this VL instance.
vnfVirtualLinkDescId	M	1	Identifier (Reference to VnfVirtualLinkDesc)	Identifier of the VLD in the VNFD for this VL.
vimId	CM	0..1	Identifier	Identifier of the VIM that manages this resource. CONDITION: This attribute shall be supported and present if VNF-related resource management in direct mode is applicable.
resourceProviderId	CM	0..1	Identifier	Identifies the entity responsible for the management of the resource. CONDITION: This attribute shall be supported and present when VNF-related Resource Management in indirect mode is applicable.
resourceId	M	1	Identifier	Identifier of the resource in the scope of the VIM or the resource provider.
vnfLinkPort	M	0..N	VnfLinkPortData	Externally provided link ports to be used to connect VNFC connection points to this externally-managed VL on this network resource. If this attribute is not present, the VNFM shall create the link ports on the externally-managed VL.
extManagedMultisiteVirtualLinkId	M	0..1	Identifier	Identifier of the externally-managed multi-site VL instance. The identifier is assigned by the NFV-MANO entity that manages the externally managed multi-site VL instance. It shall be present when the present externally-managed internal VL (indicated by extManagedVirtualLinkId) is part of a multi-site VL, e.g. in support of multi-site VNF spanning several VIMs. All externally-managed internal VL instances corresponding to an internal VL created based on the same virtualLinkDescId shall refer to the same extManagedMultisiteVirtualLinkId.

8.3.4.29 ChangeExtVnfConnectivityData information element

8.3.4.29.1 Description

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

- Disconnect external CPs that are connected to a particular external VL and connect them to a different external VL.
- Disconnect external CPs that are connected to a particular external VL.
- Disconnect and delete external CPs that are connected to a particular external VL and that represent sub-ports of a trunk port, i.e. CP instances that are created from external CPDs that have trunk mode configured according to clause 7.1.6.3 in ETSI GS NFV-IFA 011 [2]. If the parent port is exposed as an extCp, the VNFM shall ensure that the parent port is not deleted. If the parent port is exposed as an extCp and there are other subports connected, the VNFM shall ensure that the parent port is not disconnected.

- Change the connectivity parameters of the existing external CPs, including changing addresses.

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

- Connect CPs to a particular external VL.
- Create new CPs that represent sub-ports of a trunk port, i.e. CP instances that are created from external CPDs that have trunk mode configured according to clause 7.1.6.3 in ETSI GS NFV-IFA 011 [2], and connect them to a particular external VL. Creation of the parent port with this operation is not supported.

8.3.4.29.2 Attributes

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

Table 8.3.4.29.2-1: Attributes of the ChangeExtVnfConnectivityData information element

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

8.3.4.30 NfpRule Information element

8.3.4.30.1 Description

The NfpRule information element is an expression of the conditions that shall be met for the NFP to be applicable to the packet. The condition acts as a flow classifier and it is met only if all the values expressed in the condition are matched by those in the packet.

8.3.4.30.2 Attributes

The attributes of the NfpRule information element shall follow the indications provided in table 8.3.4.30.2-1.

Table 8.3.4.30.2-1: Attributes of the NfpRule information element

Attribute	Qualifier	Cardinality	Content	Description
etherType	M	0..1	Enum	Indicates the protocol carried over the Ethernet layer. VALUES: • IPV4. • IPV6. See note.
etherDestinationAddress	M	0..1	MacAddress	Indicates a destination Mac address See note.
etherSourceAddress	M	0..1	MacAddress	Indicates a source Mac address See note.
vlanTag	M	0..N	String	Indicates a VLAN identifier in an IEEE 802.1Q-2018 [i.8] tag. Multiple tags can be included for QinQ stacking. See note.
protocol	M	0..1	String	Indicates the L4 protocol, e.g. "TCP", "UDP", etc. See note.

Attribute	Qualifier	Cardinality	Content	Description
dscp	M	0..1	String	Differentiated services code point (DSCP) value. See note.
sourcePortRange	M	0..1	PortRange	Indicates a range of source ports. See note.
destinationPortRange	M	0..1	PortRange	Indicates a range of destination ports. See note.
sourceIPAddressPrefix	M	0..1	IpAddress	Indicates the source IP address or prefix. See note.
destinationIPAddressPrefix	M	0..1	IpAddress	Indicates the destination IP address or prefix. See note.
extendedCriteria	M	0..N	Not specified	Indicates values of specific bits in a frame. See note.
NOTE: The presence of at least one attribute is required. If multiple attributes are present, a logical "AND" operation shall be applied to those attributes when matching packets against the rule.				

8.3.4.31 PortRange Information element

8.3.4.31.1 Description

The PortRange information element provides the lower and upper bounds of a range of Internet ports.

8.3.4.31.2 Attributes

The attributes of the PortRange information element shall follow the indications provided in table 8.3.4.31.2-1.

Table 8.3.4.31.2-1: Attributes of the PortRange information element

Attribute	Qualifier	Cardinality	Content	Description
lowerPort	M	1	Integer	Identifies the lower bound of the port range.
upperPort	M	1	Integer	Identifies the upper bound of the port range.

8.3.4.32 AddPnfData information element

8.3.4.32.1 Description

This information element provides input information about PNF which needs to be added into an NS instance.

8.3.4.32.2 Attributes

The AddPnfData information element shall follow the indications provided in table 8.3.4.32.2-1.

Table 8.3.4.32.2-1: Attributes of the AddPnfData information element

Attribute	Qualifier	Cardinality	Content	Description
pnfId	M	1	Identifier	Identifier of the PNF. Assigned by OSS and provided to NFVO.
pnfName	M	1	String	Human readable name of the PNF.
pnfdId	M	1	Identifier (Reference to Pnfd)	Identifier of (reference to) the PNFD related to this PNF. If different to the one indicated in the referenced PnfProfile it replaces it when adding the PNF to an NS. The PNFD referenced by this attribute shall have the same pnfdExtInvariantId as the one indicated in the PnfProfile. If the PnfProfile does not contain a pnfdExtInvariantId this attribute shall be ignored (see note).

Attribute	Qualifier	Cardinality	Content	Description
pnfProfileId	M	1	Identifier (Reference to PnfProfile)	Identifier of (reference to) the PNF Profile to be used for this PNF.
cpData	M	0..N	PnfExtCpData	Information on the external CP of the PNF.
NOTE: This attribute allows the use of a PNFD different from the one specified in the NSD when adding the PNF to the NS, provided they have the same pnfdExtInvariantId.				

8.3.4.33 ModifyPnfData information element

8.3.4.33.1 Description

This information element provides input information about PNFs which need to be modified in an NS instance.

8.3.4.33.2 Attributes

The ModifyPnfData information element shall follow the indications provided in table 8.3.4.33.2-1.

Table 8.3.4.33.2-1: Attributes of the ModifyPnfData information element

Attribute	Qualifier	Cardinality	Content	Description
pnfId	M	1	Identifier	Identifier of the PNF. Assigned by OSS and provided to NFVO.
pnfName	M	0..1	String	Human readable name of the PNF. See note.
cpData	M	0..N	PnfExtCpData	Information on the external CP of the PNF. See note.
NOTE: At least one of the attributes shall be provided.				

8.3.4.34 PnfExtCpData information element

8.3.4.34.1 Description

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

8.3.4.34.2 Attributes

The attributes of the PnfExtCpData information element shall follow the indications provided in table 8.3.4.34.2-1.

Table 8.3.4.34.2-1: Attributes of the PnfExtCpData information element

Attribute	Qualifier	Cardinality	Content	Description
cpInstanceId	M	0..1	Identifier	Identifier of this external CP instance. Shall be present for existing CP.
cpdId	M	0..1	Identifier (Reference to Cpd)	Identifier of (reference to) the Connection Point Descriptor (CPD) for this CP. Shall be present for new CP.
address	M	1..N	Not specified	Address for this CP, including the information on applicable layer protocol(s). See note.
NOTE: The address information shall be compatible with the layerProtocol values defined in the CPD. In case of an IP address, a port number may be included.				

8.3.4.35 ExtLinkPortData information element

8.3.4.35.1 Description

This information element represents an externally provided link port to be used to connect an external connection point to an external VL.

8.3.4.35.2 Attributes

The ExtLinkPortData information element shall follow the indications provided in table 8.3.4.35.2-1.

Table 8.3.4.35.2-1: Attributes of the ExtLinkPortData information element

Attribute	Qualifier	Cardinality	Content	Description
extLinkPortId	M	1	Identifier	Identifier of this link port as provided by the entity that has created the link port.
resourceHandle	M	1	ResourceHandle	Resource handle of the virtualised resource that realizes the external link port.

8.3.4.36 VnfExtCpConfig information element

8.3.4.36.1 Description

This information element represents an externally provided link port or network address information per instance of a VNF external connection point. In case a link port is provided, the NFVO shall use that link port when connecting the VNF external CP to the external VL. In case no link port is provided, the NFVO or VNFM shall create a link port on the external VL, and use that link port to connect the VNF external CP to the external VL.

8.3.4.36.2 Attributes

The VnfExtCpConfig information element shall follow the indications provided in table 8.3.4.36.2-1.

Table 8.3.4.36.2-1: Attributes of the VnfExtCpConfig information element

Attribute	Qualifier	Cardinality	Content	Description
cpInstanceId	M	0..1	Identifier	Identifier of the external CP instance to which this set of configuration parameters is requested to be applied. Shall be present if this instance has already been created.
linkPortId	M	0..1	Identifier (Reference to ExtLinkPortData)	Identifier of a pre-configured link port to which the external CP will be associated. See note.
cpProtocolData	M	0..N	Not specified	Parameters for configuring fixed and dynamic network addresses for the CP, including the information on applicable layer protocol(s). For dynamic addresses, it should be possible to define per parameter set the number of network addresses to be assigned dynamically. Other parameters could be e.g. valid address ranges or subnets. See note.
NOTE: The following conditions apply to the attributes "linkPortId" and "cpProtocolData": 1) The "linkPortId" and "cpProtocolData" attributes shall both be absent for the deletion of an existing external CP instance addressed by cpInstanceId. 2) At least one of these attributes shall be present for a to-be-created external CP instance or an existing external CP instance.				

8.3.4.37 NestedNsInstanceData information element

8.3.4.37.1 Description

The NestedNsInstanceData specifies an existing nested NS instance to be used in the NS instance and if needed, the NsProfile to use for this nested NS instance.

8.3.4.37.2 Attributes

The attributes of the NestedNsInstanceData information element shall follow the indications provided in table 8.3.4.37.2-1.

Table 8.3.4.37.2-1: Attributes of the NestedNsInstanceData information element

Attribute	Qualifier	Cardinality	Content	Description
nestedNsInstanceId	M	1	Identifier (Reference to NsInfo)	Identifier of the existing nested NS instance to be used in the NS.
nsProfileId	M	0..1	Identifier (Reference to NsProfile)	Identifier of an NsProfile defined in the NSD which the existing nested NS instance shall be matched with. If not present, the NFVO will select the NsProfile matching the information in the nested NS instance. See note 1.
overridingNsdId	M	0..1	Identifier (Reference to an Nsd)	It replaces the nsdId indicated in the nsProfileId. The NSD referenced by this attribute shall have the same nsdExtInvariantId as the one indicated in the NsProfile. If the NsProfile does not contain an nsdExtInvariantId this attribute shall be ignored. If the NSD of the existing NS instance referenced by nestedNsInstanceId does not match the NSD indicated by this attribute the NS instance shall not be added as nested NS to the composite NS instance. See notes 1 and 2.
NOTE 1: If the overridingNsdId attribute is present the nsProfileId attribute shall also be present. NOTE 2: This attribute allows to use as nested NS an existing NS instance based on a different NSD to the one specified in the composite NSD with nsProfileId, provided they have the same nsdExtInvariantId.				

8.3.4.38 ParamsForNestedNs information element

8.3.4.38.1 Description

The ParamsForNestedNs specifies additional parameters on a per nested NS instance basis.

8.3.4.38.2 Attributes

The attributes of the ParamsForNestedNs information element shall follow the indications provided in table 8.3.4.38.2-1.

Table 8.3.4.38.2-1: Attributes of the ParamsForNestedNs information element

Attribute	Qualifier	Cardinality	Content	Description
nsProfileId	M	1	Identifier (Reference to NsProfile)	Identifier of an NsProfile to which the additional parameters apply.
additionalParam	M	0..N	KeyValuePair	Additional parameters that are to be applied per nested NS instance.
overridingNsdId	M	0..1	Identifier (Reference to an NSD)	If present it replaces the nsdId indicated in the referenced NsProfile at the instantiation of the nested NS. The NSD referenced by this attribute shall have the same nsdExtInvariantId as the one indicated in the NsProfile. If the NsProfile does not contain an nsdExtInvariantId this attribute shall be ignored. See note.
NOTE: This attribute allows for a nested NS at instantiation of the composite NS the use of an NSD different from the one specified in the NSD of the composite NS with nsProfileId, provided they have the same nsdExtInvariantId.				

8.3.4.39 RevertToSnapshotData information element

8.3.4.39.1 Description

The RevertToSnapshotData specifies an existing VNF instance of the NS instance to be reverted and the identifier of an existing VNF Snapshot to be reverted to.

8.3.4.39.2 Attributes

The attributes of the RevertToSnapshotData information element shall follow the indications provided in table 8.3.4.39.2-1.

Table 8.3.4.39.2-1: Attributes of the RevertToSnapshotData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance to be reverted.
vnfSnapshotInfoId	M	1	Identifier	Identifier of information held by the VNFM about the VNF Snapshot to be reverted to. This identifier was allocated by the VNFM.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS as input for the revert to VNF snapshot process, specific for the VNF being "reverted", as declared in the VNFD (see clause 7.1.5.12 in ETSI GS NFV-IFA 011 [2]).

8.3.4.39a CreateSnapshotData information element

8.3.4.39a.1 Description

The CreateSnapshotData specifies an existing VNF instance of the NS instance to be snapshotted.

8.3.4.39a.2 Attributes

The attributes of the CreateSnapshotData information element shall follow the indications provided in table 8.3.4.39a.2-1.

Table 8.3.4.39a.2-1: Attributes of the CreateSnapshotData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance to perform the snapshot from.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS as input for the VNF snapshot creation process, specific for the VNF being "snapshotted", as declared in the VNFD (see clause 7.1.5.11 in ETSI GS NFV-IFA 011 [2]).
userDefinedData	O	0..N	KeyValuePair	User defined data for the VNF snapshot.

8.3.4.40 DeleteSnapshotData information element

8.3.4.40.1 Description

The DeleteSnapshotData specifies the identifier of information of an available VNF Snapshot to be deleted and the identifier of the related VNF instance of the NS instance.

8.3.4.40.2 Attributes

The attributes of the DeleteSnapshotData information element shall follow the indications provided in table 8.3.4.40.2-1.

Table 8.3.4.40.2-1: Attributes of the DeleteSnapshotData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfSnapshotInfoId	M	1	Identifier	Identifier of information held by the VNFM about the VNF Snapshot to be deleted. This identifier was allocated by the VNFM.
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance to identify the VNFM holding the VNF Snapshot information to be deleted.

8.3.4.41 PnfProfileData information element

8.3.4.41.1 Description

The PnfProfileData information element specifies the information needed to associate a PNF with the PnfProfile. The types of association may include:

- Associate the PNF with a new PnfProfile that has been added to the NSD. If the PNF is currently associated with an existing PnfProfile, it should dissociate with such PnfProfile prior to association with the new PnfProfile.
- Associate the PNF with the PnfProfile that has been updated.

In the present document version, the type of changes permitted in the PnfProfile which allow the re-association of the PNF to the new PnfProfile are:

- To support the changes of connectivity of the PNF, i.e. changes in the pnfVirtualLinkConnectivity (refer to ETSI GS NFV-IFA 014 [3], clause 6.3.6).

The association may result in adding or changing connectivity for the PNF instance.

8.3.4.41.2 Attributes

The attributes of the PnfProfileData information element shall follow the indications provided in table 8.3.4.41.2-1.

Table 8.3.4.41.2-1: Attributes of the PnfProfileData information element

Attribute	Qualifier	Cardinality	Content	Description
pnfId	M	1	Identifier	Identifier of the PNF.
pnfProfileId	M	1	Identifier (Reference to PnfProfile)	Identifier of (Reference to) a PnfProfile. The PnfProfile can be an update of the existing PNF profile or a new PNF profile, which the PNF shall be associated with. See note.
NOTE: The PnfProfile may contain the new or changed NsVirtualLinkConnectivity for the PNF.				

8.3.4.42 VnfProfileData information element

8.3.4.42.1 Description

The VnfProfileData information element specifies the information needed to associate a VNF instance with the VnfProfile. The types of association may include:

- Associate the VNF instance with a new VnfProfile that has been added to the NSD. If the VNF instance is currently associated with an existing VnfProfile, it should dissociate with such VnfProfile prior to association with the new VnfProfile.
- Associate the VNF instance with the VnfProfile that has been updated.

In the present document version, the type of changes permitted in the VnfProfile which allow the re-association of the VNF to the new VnfProfile are:

- To support the changes of connectivity of the VNF, i.e. changes in the nsVirtualLinkConnectivity (refer to ETSI GS NFV-IFA 014 [3], clause 6.3.3).

8.3.4.42.2 Attributes

The attributes of the VnfProfileData information element shall follow the indications provided in table 8.3.4.42.2-1.

Table 8.3.4.42.2-1: Attributes of the VnfProfileData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance.
vnfProfileId	M	1	Identifier (Reference to VnfProfile)	Identifier of (Reference to) a VnfProfile. The VnfProfile can be an update of the existing VNF profile or a new VNF profile, which the VNF instance shall be associated with. See note.
NOTE: The VnfProfile may contain the new or changed NsVirtualLinkConnectivity for the VNF.				

8.3.4.43 WanConnectionData information element

8.3.4.43.1 Description

This information element provides the needed information required to connect to the WAN the comprising network resources realizing a VL, e.g. when the VL is deployed on several sites across a WAN.

8.3.4.43.2 Attributes

The WanConnectionData information element shall follow the indications provided in table 8.3.4.43.2-1.

Table 8.3.4.43.2-1: Attributes of the WanConnectionData information element

Attribute	Qualifier	Cardinality	Content	Description
virtualLinkDescId	M	1	Identifier (Reference to NsVirtualLinkDesc or VnfVirtualLinkDesc)	Identifier of the VLD in the NSD or the VNFD for this VL, for which the connection data is provided.
protocolData	M	0..1	Not specified	Protocol specific information for connecting to the WAN the virtualised networks in the NFVI-PoP managed by the VIM.

8.3.4.44 VnfLinkPortData information element

8.3.4.44.1 Description

This information element represents an externally provided link port to be used to connect a VNFC connection point to an externally-managed VL.

8.3.4.44.2 Attributes

The VnfLinkPortData information element shall follow the indications provided in table 8.3.4.44.2-1.

Table 8.3.4.44.2-1: Attributes of the VnfLinkPortData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfLinkPortId	M	1	Identifier	Identifier of this link port as provided by the entity that has created the link port.
resourceHandle	M	1	ResourceHandle	Resource handle of the virtualised resource that realizes the link port.

8.3.4.45 ChangeVnfPackageData information element

8.3.4.45.1 Description

The ChangeVnfPackageData information element specifies the information needed to change the current VNF package for a VNF instance.

8.3.4.45.2 Attributes

The attributes of the ChangeVnfPackageData information element shall follow the indications provided in table 8.3.4.45.2-1.

Table 8.3.4.45.2-1: Attributes of the ChangeVnfPackageData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifier of the VNF instance for which the current VNF package is requested to be changed.
vnfdId	M	1	Identifier (Reference to Vnfd)	Identifier of the VNFD which defines the destination VNF Package for the change.
extVirtualLink	M	0..N	ExtVirtualLinkData	Information about external VLs to connect the VNF to.
extManagedVirtualLink	M	0..N	ExtManagedVirtualLinkData	Information about internal VLs that are managed by other entities than the VNFM. See notes 1 and 2.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the modification process, specific to the VNF, whose VNF Package is requested to be changed, as declared in the VNFD (see clause 7.1.5.13 in ETSI GS NFV-IFA 011 [2]).
extension	M	0..N	KeyValuePair	This parameter provides values for the "extension" input parameter of the Change Current VNF Package operation.
vnfConfigurableProperties	M	0..N	KeyValuePair	This parameter provides values for the "vnfConfigurableProperties" input parameter of the Change Current VNF Package operation.

NOTE 1: The indication of externally-managed internal VLs is needed in case networks have been pre-configured for use with certain VNFs, for instance to ensure that these networks have certain properties such as security or acceleration features, or to address particular network topologies. The present document assumes that externally-managed internal VLs are managed by the NFVO and created towards the VIM as supported by the virtualised network resource management interface specified in ETSI GS NFV-IFA 005 [i.4].

NOTE 2: It is possible to have several ExtManagedVirtualLinkData for the same VNF internal VL in case of a multi-site VNF spanning several VIMs. The set of ExtManagedVirtualLinkData corresponding to the same VNF internal VL shall indicate so by referencing to the same VnfVirtualLinkDesc and externally-managed multi-site VL instance (refer to clause 8.3.4.28).

8.3.4.46 ModifyWanConnectionInfoData information element

8.3.4.46.1 Description

This information element provides information requested to be modified about WAN related connectivity information.

8.3.4.46.2 Attributes

The ModifyWanConnectionInfoData information element shall follow the indications provided in table 8.3.4.46.2-1.

Table 8.3.4.46.2-1: Attributes of the ModifyWanConnectionInfoData information element

Attribute	Qualifier	Cardinality	Content	Description
wanConnectionInfoId	M	1	Identifier (Reference to WanConnectionInfo)	References the WAN connection information to be modified.
newProtocolData	M	0..1	Not specified	New protocol specific information to be modified in the referred WAN connection information.

8.3.4.47 NestedNsLocationConstraint information element

8.3.4.47.1 Description

The NestedNsLocationConstraint information element defines the location constraints for the nested NS to be instantiated.

8.3.4.47.2 Attributes

The attributes of the NestedNsLocationConstraint information element shall follow the indications provided in table 8.3.4.47.2-1.

Table 8.3.4.47.2-1: Attributes of the NestedNsLocationConstraint information element

Attribute	Qualifier	Cardinality	Content	Description
nsProfileId	M	1	Identifier (Reference to NsProfile)	Identifier (reference to) of an NsProfile in the NSD used to manage the lifecycle of the nested NS instance.
locationConstraints	M	1	Not specified	Defines the location constraints for the nested NS instance to be created.

8.3.4.48 TerminateNsData information element

8.3.4.48.1 Description

This information element describes the information needed to terminate NS instance.

8.3.4.48.2 Attributes

The attributes of the TerminateNsData information element shall follow the indications provided in table 8.3.4.48.2-1.

Table 8.3.4.48.2-1: Attributes of the TerminateNsData information element

Attribute	Qualifier	Cardinality	Content	Description
additionalParamForNs	M	0..N	KeyValuePair	Additional parameter(s) passed by the OSS/BSS to the termination process at the NS level.

8.3.4.49 TerminateVnfData information element

8.3.4.49.1 Description

The information element describes the information needed to terminate a VNF instance that is part of an NS. The specific parameters passed by the NFVO to the VNFM are specified in clause 7.2.7.2 of ETSI GS NFV-IFA 007 [i.5].

8.3.4.49.2 Attributes

The attributes of the TerminateVnfData information element shall follow the indications provided in table 8.3.4.49.2-1.

Table 8.3.4.49.2-1: Attributes of the TerminateVnfData information element

Attribute	Qualifier	Cardinality	Content	Description
vnfInstanceId	M	1	Identifier	Identifies the VNF instance, part of the NS, to be terminated.
terminationType	M	0..1	Enum	Indicates the type of termination that is requested. VALUES: • FORCEFUL • GRACEFUL
gracefulTerminationTimeout	M	0..1	TimeDuration	The time interval to wait for the VNF to be taken out of service during graceful termination, before shutting down the VNF and releasing the resources.
additionalParam	M	0..N	KeyValuePair	Additional parameters passed by the OSS/BSS as input to the termination process, specific to the VNF being terminated. EXAMPLE: Input parameters to VNF-specific termination procedures.

8.3.5 NsLcmCapacityShortageNotification

8.3.5.1 Description

This notification informs the consumer of NS LCM operations about shortage conditions. It will be sent if an LCM operation could not be executed or completed because of a shortage condition. It is expected that the shortage condition is temporary and the consumer is again notified when the condition has ended. The support of the notification is mandatory.

8.3.5.2 Trigger conditions

This notification is produced when there is a shortage condition that caused an LCM operation to be not successfully completed: The shortage conditions include:

- Necessary resources could not be allocated during an LCM operation because of resource shortage.
- An NS instance with higher priority pre-empted an LCM operation because of resource shortage.
- An NS instance with higher priority pre-empted a running NS instance. Resources were de-allocated from the lower priority NS instance to allow the LCM operation on a higher priority NS instance.
- Due to a capacity or performance shortage within the MANO system an LCM operation could not be executed.
- The resource, capacity or MANO performance shortage situation has ended and it can be expected that an LCM operation that had failed could succeed now.

If this is a notification about pre-emption, it shall be sent to both consumers, that is the tenants of the lower priority NS instance and the higher priority NS instance. See ETSI GS NFV-IFA 010 [1], clause D.2 for use cases.

8.3.5.3 Attributes

The attributes of the NsLcmCapacityShortageNotification notification shall follow the indications provided in table 8.3.5.3-1.

Table 8.3.5.3-1: Attributes of the NsLcmCapacityShortageNotification

Attribute	Qualifier	Cardinality	Content	Description
lifecycleOperationOccurrenceId	M	1	Identifier	Identifier of the NS lifecycle operation occurrence this notification is related to.
nsInstanceId	M	1	Identifier	Identifier of the NS instance affected by that lifecycle operation.
status	M	1	Not specified	Indicates the situation of capacity shortage, including: • The lifecycle operation identified by the first parameter could not be executed because the necessary resources were not available. • The lifecycle operation identified by the first parameter pre-empted another lifecycle operation of an NS instance with lower priority. • The lifecycle operation identified by the first parameter was pre-empted by another lifecycle operation of an NS instance with higher priority. • The lifecycle operation identified by the first parameter pre-empted one or more running NS instances with lower priority. • The lifecycle operation identified by the first parameter was triggered by a lifecycle operation on a higher priority NS instance pre-empting a lower priority instance. • The shortage situation has ended and the lifecycle operation identified by the first parameter could be tried again.
shortageType	M	1	Not specified	Indicates whether this notification reports about a resource shortage or MANO capacity or performance shortage.
affectedNs	M	0..N	AffectedNs	Information about the NS instances that were affected by the shortage.
capacityInformation	O	1..N	Not specified	Information about the required, available, reserved, allocated/used, and total capacity as applicable for the notification.

8.4 Information elements and notifications related to NS Performance Management

8.4.1 Introduction

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

8.4.2 ObjectSelection information element

8.4.2.1 Description

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

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

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

The pattern proposes 2 exclusive options:

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

2) Provide a list of object instances.

In the present interface, the object type will be NS related measured object types (see note).

NOTE: The NS related measured object types are the measured object type(s) for which the performance measurements applicable to Os-Ma-nfvo reference point are defined in clause 7.3 of ETSI GS NFV-IFA 027 [5].

8.4.2.2 Attributes

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

Table 8.4.2.2-1: Attributes of the ObjectSelection information element

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

8.4.3 PmJob information element

8.4.3.1 Description

This information element provides the details of the PM Job.

The object instances for this information element will be the instances corresponding to the NS related measured object types.

8.4.3.2 Attributes

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

Table 8.4.3.2-1: Attributes of the PmJob information element

Attribute	Qualifier	Cardinality	Content	Description
pmJobId	M	1	Identifier	Identifier of this PmJob information element.
objectSelector	M	1	ObjectSelection	Defines the object instances for which performance information is requested to be collected. The object instances for this information element will be the instances corresponding to the NS related measured object types.
performanceMetric	M	0..N	String	This defines the type of performance metric(s) for the object instances. Valid values are specified as "Measurement Name" values of the performance measurements applicable to Os-Ma-nfvo reference point, as defined in clause 7.3 of ETSI GS NFV-IFA 027 [5]. At least one of the two (performance metric or metricGroup) shall be present.
performanceMetricGroup	M	0..N	String	Group of performance metrics. A metric group is a pre-defined list of metrics, known to the producer that it can decompose to individual metrics. Valid values are specified as "Measurement Group" values of the performance measurements applicable to Os-Ma-nfvo reference point, as defined in clause 7.3 of ETSI GS NFV-IFA 027 [5]. At least one of the two (performance metric or metricGroup) shall be present.
collectionPeriod	M	1	Not specified	Specifies the periodicity at which the producer will collect performance information. See note.
reportingPeriod	M	1	Not specified	Specifies the periodicity at which the producer will report to the consumer about performance information. See note.
reportingBoundary	O	0..1	Not specified	Identifies a boundary after which the reporting will stop. The boundary shall allow a single reporting as well as periodic reporting up to the boundary.

Attribute	Qualifier	Cardinality	Content	Description
NOTE: At the end of each reportingPeriod, the producer will inform the consumer about availability of the performance data collected for each completed collection period during this reportingPeriod. While the exact definition of the types for collectionPeriod and reporting period is part of the protocol design, it is recommended that the reportingPeriod be equal or a multiple of the collectionPeriod. In the latter case, the performance data for the collection periods within one reporting period would be reported together.				

8.4.4 Threshold information element

8.4.4.1 Description

This information element provides the details of a threshold.

The object instances for this information element will be the instances corresponding to the NS related measured object types.

8.4.4.2 Attributes

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

Table 8.4.4.2-1: Attributes of the Threshold information element

Attribute	Qualifier	Cardinality	Content	Description
thresholdId	M	1	Identifier	Identifier of this Threshold information element.
objectSelector	M	1	ObjectSelection	Defines the object instances associated with the threshold. The object instances for this information element will be the instances corresponding to the NS related measured object types.
performanceMetric	M	1	String	Defines the performance metric associated with the threshold. Valid values are specified as "Measurement Name" values of the performance measurements applicable to Os-Ma-nfvo reference point, as defined in clause 7.3 of ETSI GS NFV-IFA 027 [5].
thresholdType	M	1	Enum	Type of threshold. The list of possible values is part of the protocol design and might include: single/multi valued threshold, static/dynamic threshold, template based threshold, etc. VALUES: • SIMPLE: Single-valued static threshold • etc.
thresholdDetails	M	1	Not specified	Details of the threshold: value to be crossed, details on the notification to be generated.

8.4.5 PerformanceReport information element

8.4.5.1 Description

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

The object instances for this information element will be the instances corresponding to the NS related measured object types.

8.4.5.2 Attributes

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

Table 8.4.5.2-1: Attributes of the PerformanceReport information element

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

8.4.6 PerformanceReportEntry information element

8.4.6.1 Description

This information element defines a single performance report entry.

The object instances for this information element will be the instances corresponding to the NS related measured object types.

8.4.6.2 Attributes

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

Table 8.4.6.2-1: Attributes of the PerformanceReportEntry information element

Attribute	Qualifier	Cardinality	Content	Description
objectType	M	1	String	Defines the object type. The object types for this information element will be the NS related measured object types.
objectInstanceId	M	1	Identifier	The object instance for which the performance metric is reported. The object instances for this information element will be the instances corresponding to the NS related measured object types.
performanceMetric	M	1	String	Name of the metric collected. This attribute's value contains the related "Measurement Name" value of the performance measurements applicable to Os-Ma-nfvo reference point, as defined in clause 7.3 of ETSI GS NFV-IFA 027 [5].
performanceValue	M	1..N	PerformanceValueEntry	List of performance values with associated timestamp and measurement context (see ETSI GS NFV-IFA 027 [5]).

8.4.7 PerformanceValueEntry information element

8.4.7.1 Description

This information element defines a single performance value with its associated time stamp and measurement context (see ETSI GS NFV-IFA 027 [5]).

8.4.7.2 Attributes

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

Table 8.4.7.2-1: Attributes of the PerformanceValueEntry information element

Attribute	Qualifier	Cardinality	Content	Description
timeStamp	M	1	DateTime	Timestamp indicating when the data was collected.
performanceValue	M	1	Value	Value of the metric collected. The type of this attribute corresponds to the related "Measurement Unit" of the performance measurements applicable to Os-Ma-nfvo reference point, as defined in clause 7.3 of ETSI GS NFV-IFA 027 [5].
measurementContext	M	0..1	Not specified	Measurement context of the metric collected. The specific measurement context for each kind of performance metrics is defined in ETSI GS NFV-IFA 027 [5].

8.4.8 PerformanceInformationAvailableNotification

8.4.8.1 Description

This notification informs the receiver that performance information is available. Delivery mechanism for the performance reports is not specified in the present document.

The object instances for this information element will be the instances corresponding to the NS related measured object types.

8.4.8.2 Trigger Conditions

The notification is produced when:

- New performance information is available.

8.4.8.3 Attributes

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

Table 8.4.8.3-1: Attributes of the PerformanceInformationAvailableNotification

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

8.4.9 ThresholdCrossedNotification

8.4.9.1 Description

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

The object instances for this information element will be the instances corresponding to the NS related measured object types.

8.4.9.2 Trigger Conditions

The notification is produced when:

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

8.4.9.3 Attributes

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

Table 8.4.9.3-1: Attributes of the ThresholdCrossedNotification

Attribute	Qualifier	Cardinality	Content	Description
thresholdId	M	1	Identifier (Reference to Threshold)	Threshold which has been crossed.
crossingDirection	M	1	Enum	An indication of whether the threshold was crossed in upward or downward direction. VALUES: • UP. • DOWN.
objectInstanceId	M	1	Identifier	Object instance for which the threshold has been crossed. The object instances for this information element will be the instances corresponding to the NS related measured object types.
performanceMetric	M	1	String	Performance metric associated with the threshold This attribute's value contains the related "Measurement Name" value of the performance measurements applicable to Os-Ma-nfvo reference point, as defined in clause 7.3 of ETSI GS NFV-IFA 027 [5].
performanceValue	M	1	Value	Value of the metric that resulted in threshold crossing.
measurementContext	M	0..1	Not specified	Measurement context of the metric collected. The specific measurement context for each kind of performance metrics is defined in ETSI GS NFV-IFA 027 [5].

8.5 Information elements and notifications NS Fault management

8.5.1 Introduction

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

8.5.2 AlarmNotification

8.5.2.1 Description

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

8.5.2.2 Trigger conditions

The notification is produced when:

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

8.5.2.3 Attributes

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

Table 8.5.2.3-1: Attributes of the AlarmNotification

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

8.5.3 AlarmClearedNotification

8.5.3.1 Description

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

8.5.3.2 Trigger conditions

The notification is produced when:

- An alarm has been cleared.

8.5.3.3 Attributes

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

Table 8.5.3.3-1: Attributes of the AlarmClearedNotification

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

8.5.4 Alarm information element

8.5.4.1 Description

The Alarm information element encapsulates information about an alarm.

8.5.4.2 Attributes

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

Table 8.5.4.2-1: Attributes of the Alarm information element

Attribute	Qualifier	Cardinality	Content	Description
alarmId	M	1	Identifier	Identifier of this Alarm information element.
managedObjectId	M	1	Identifier	Identifier of the affected managed object. The Managed Objects for the current specification will be network services.
rootCauseFaultyComponent	M	1	FaultyComponentInfo	The NS components (e.g. nested NS, NS virtual link and VNF) that are causing the NS fault.
rootCauseFaultyResource	M	0..1	FaultyResourceInfo	The virtualised resources that are causing the faulty component. It shall be present when the faulty component is "NS Virtual Link" or "VNF" (see clause 8.5.6).
alarmRaisedTime	M	1	DateTime	Timestamp indicating when the alarm was raised by the network service.

Attribute	Qualifier	Cardinality	Content	Description
alarmChangedTime	M	0..1	DateTime	Timestamp indicating when the alarm was last changed.
alarmClearedTime	M	0..1	DateTime	Timestamp indicating when the alarm was cleared.
ackState	M	1	Enum	State of the alarm. VALUES: • ACKNOWLEDGED • UNACKNOWLEDGED
perceivedSeverity	M	1	Enum	Perceived severity of the managed object failure. VALUES: • CRITICAL • MAJOR • MINOR • WARNING • INDETERMINATE • CLEARED
eventTime	M	1	DateTime	Timestamp indicating when the fault was observed.
eventType	M	1	Enum	Type of the event. The allowed values for the eventType attribute use the event type defined in Recommendation ITU-T X.733 [4]. VALUES: • COMMUNICATIONS_ALARM • PROCESSING_ERROR_ALARM • ENVIRONMENTAL_ALARM • QOS_ALARM • EQUIPMENT_ALARM
faultType	M	0..1	String	Additional information related to the type of the fault.
probableCause	M	1	String	Information about the probable cause of the fault.
isRootCause	M	1	Boolean	Parameter indicating if this fault is the root for other correlated alarms. If TRUE, then the alarms listed in the parameter correlatedAlarmId are caused by this fault.
correlatedAlarmId	M	0..N	Identifier (Reference to Alarm)	List of identifiers of other alarms correlated to this fault.
faultDetails	M	0..N	Not specified	Provides additional information about the fault.

8.5.5 AlarmListRebuiltNotification

8.5.5.1 Description

This notification informs the receiver that the active alarm list has been rebuilt by the NFVO. Upon receipt of this notification, the receiver needs to use the "Get Alarm List" operation to synchronize its view on current active alarms with that of the NFVO.

The notification is mandatory.

8.5.5.2 Trigger conditions

- Active alarm list has been rebuilt by the NFVO, e.g. if the NFVO detects its storage holding the alarm list is corrupted.

8.5.5.3 Attributes

The AlarmListRebuiltNotification does not contain any attributes.

8.5.6 FaultyComponentInfo information element

8.5.6.1 Description

The FaultyComponentInfo information element encapsulates information about faulty component that has a negative impact on an NS.

8.5.6.2 Attributes

The FaultyComponentInfo information element shall follow the indications provided in table 8.5.6.2-1.

Table 8.5.6.2-1: Attributes of the FaultyComponentInfo information element

Attribute	Qualifier	Cardinality	Content	Description
faultyNestedNsInstanceId	M	0..1	Identifier (Reference to NsInfo)	Identifier of the faulty nested NS instance. See note.
faultyNsVirtualLinkInstanceId	M	0..1	Identifier (Reference to NsVirtualLinkInfo)	Identifier of the faulty NS virtual link instance. See note.
faultyVnfInstanceId	M	0..1	Identifier (Reference to VnfInfo)	Identifier of the faulty VNF instance. See note.
NOTE: At least one of the parameters shall be present.				

8.5.7 FaultyResourceInfo information element

8.5.7.1 Description

The FaultyResourceInfo information element encapsulates information about faulty resource that has a negative impact on a VNF or an NS virtual link, which is the constituent component of the impacted NS.

8.5.7.2 Attributes

The FaultyResourceInfo information element shall follow the indications provided in table 8.5.7.2-1.

Table 8.5.7.2-1: Attributes of the FaultyResourceInfo information element

Attribute	Qualifier	Cardinality	Content	Description
faultyResource	M	1	ResourceHandle	Information that identifies the faulty resource instance and its managing entity. See clause 8.3.3.8.
faultyResourceType	M	1	Enum	Type of the faulty resource. VALUES: • COMPUTE • STORAGE • NETWORK

8.6 Information elements and notifications related to VNF Package

8.6.1 Introduction

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

8.6.2 VnfPkgInfo information element

8.6.2.1 Description

This information element provides the details of a VNF Package.

8.6.2.2 Attributes

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

Table 8.6.2.2-1: Attributes of the VnfPkgInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfPkgInfoId	M	1	Identifier	Identifier of the VNF Package information object. This identifier was allocated by the NFVO.
vnfdId	M	0..1	Identifier	Identifier of the on-boarded the VNF Package. See notes 1, 2 and 3.
vnfProvider	M	0..1	String	Provider of the on-boarded VNF package. See notes 2 and 3.
vnfProductName	M	0..1	String	Product name of the on-boarded VNF package. See notes 2 and 3.
vnfSoftwareVersion	M	0..1	Version	Software version of the on-boarded VNF package. See notes 2 and 3.
vnfdVersion	M	0..1	Version	VNFD version of the on-boarded VNF package. See notes 2 and 3.
checksum	M	0..1	Not specified	Checksum of the on-boarded VNF Package. See note 3.
vnfd	M	0..1	Identifier (Reference to Vnfd)	Reference to the VNFD contained in the on-boarded VNF Package, e.g. URL to the on-boarded VNFD. See note 3.
softwareImage	M	0..N	VnfPackageSoftwareImage Info	Information about VNF Package artifacts that are software images. See note 3.
additionalArtifact	M	0..N	VnfPackageArtifactInformat ion	Information about VNF Package artifacts contained in the VNF Package that are not software images. See note 4.
onboardingState	M	1	Enum	On-boarding state of the VNF Package. VALUES: • CREATED • UPLOADING • PROCESSING • ONBOARDED
operationalState	M	1	Enum	Operational state of the VNF Package. VALUES: • ENABLED • DISABLED
usageState	M	1	Enum	Usage state of the VNF Package. VALUES: • IN_USE • NOT_IN_USE
userDefinedData	O	0..N	KeyValuePair	User defined data for the VNF Package.

Attribute	Qualifier	Cardinality	Content	Description
NOTE 1: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2. NOTE 2: This information is copied from the VNFD of the on-boarded VNF Package. NOTE 3: These attributes shall be present after the VNF Package is on-boarded. NOTE 4: It may be present after the VNF Package is on-boarded and shall be absent otherwise.				

8.6.3 Vnfd information element

8.6.3.1 Description

This information element provides the details of the VNFD.

8.6.3.2 Attributes

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

8.6.4 VnfPackageSoftwareImageInfo information element

8.6.4.1 Description

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

8.6.4.2 Attributes

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

Table 8.6.4.2-1: Attributes of the VnfPackageSoftwareImageInfo information element

Attribute	Qualifier	Cardinality	Content	Description
id	M	1	Identifier	The identifier of this software image.
name	M	1	Not specified	The name of this software image.
provider	M	1	Not specified	The provider of this software image.
version	M	1	Not specified	The version of this software image.
checksum	M	1	Not specified	The checksum of the software image file.
containerFormat	M	1	Not specified	The container format indicates whether the software image is in a file format that also contains metadata about the actual software.
diskFormat	M	1	Not specified	The disk format of a software image is the format of the underlying disk image.
createdAt	M	1	Not specified	The created time of this software image.
minDisk	M	1	Not specified	The minimal Disk for this software image.
minRam	M	1	Not specified	The minimal RAM for this software image.
size	M	1	Not specified	The size of this software image.
userMetadata	M	0..N	KeyValuePair	User-defined metadata.
accessInformation	M	1	Not specified.	Information (such as a URL, a path in the VNF Package, or an identifier) that allows to access a copy of this software image artifact.

8.6.5 Void

8.6.6 VnfPackageArtifactInformation information element

8.6.6.1 Description

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

8.6.6.2 Attributes

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

Table 8.6.6.2-1: Attributes of the VnfPackageArtifactInformation information element

Attribute	Qualifier	Cardinality	Content	Description
selector	M	1	Not specified	Information (such as a URL, a path in the VNF Package, or an identifier) that allows to access a copy of this artifact.
metadata	M	1	Not specified	The metadata of the artifact that are available in the VNF Package, such as Content type, size, creation date, etc.

8.6.7 Void

8.6.8 VnfPackageOnBoardingNotification

8.6.8.1 Description

This notification indicates a VNF Package is on-boarded, after all the on-boarding steps (e.g. uploading and processing) are done. A change in on-boarding state before the VNF Package is on-boarded is not reported.

Support of this notification is mandatory.

8.6.8.2 Trigger Conditions

- New VNF Package on-boarded.

8.6.8.3 Attributes

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

Table 8.6.8.3-1: Attributes of the VnfPackageOnBoardingNotification

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

8.6.9 VnfPackageChangeNotification

8.6.9.1 Description

This notification indicates a change of status in an on-boarded VNF Package. Only changes in operational state and the deletion of the VNF Package will be reported, change in usage state is not reported.

Support of this notification is mandatory.

8.6.9.2 Trigger Conditions

- Change of the operational state of an on-boarded VNF Package.
- Deletion of an on-boarded VNF Package.

8.6.9.3 Attributes

The attributes of the VnfPackageChangeNotification shall follow the indications provided in table 8.6.9.3-1.

Table 8.6.9.3-1: Attributes of the VnfPackageChangeNotification

Attribute	Qualifier	Cardinality	Content	Description
onboardedVnfPkgInfoId	M	1	Identifier	Identifier of the VNF Package information object.
vnfdId	M	1	Identifier	Identifier of the on-boarded VNF Package. See note.
changeType	M	1	Enum	It categorizes the type of change. Possible values can be change of operational state of an on-boarded VNF Package and deletion of a VNF Package. VALUES: • OP_STATE_CHANGE: change of operational state of an on-boarded VNF Package • PKG_DELETE: deletion of a VNF Package
operationalState	M	0..1	Enum	New operational state of the VNF Package. VALUES: • ENABLED • DISABLED
NOTE: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way. See ETSI GS NFV-IFA 011 [2], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package.				

8.7 Information elements and notifications related to NFVI Capacity Information

8.7.1 Description

The clause below defines a notification related to NFVI Capacity Information.

8.7.2 CapacityThresholdCrossedNotification

8.7.2.1 Description

This notification informs the receiver that the available NFVI capacity has crossed a threshold value in either up or down direction.

8.7.2.2 Trigger Conditions

The notification is produced when:

- The NFVI capacity has crossed a threshold.

8.7.2.3 Attributes

The attributes of the CapacityThresholdCrossedNotification shall follow the indications provided in table 8.7.2.3-1.

Table 8.7.2.3-1: Attributes of the CapacityThresholdCrossedNotification

Attribute	Qualifier	Cardinality	Content	Description
subscriptionId	M	1	Identifier	Identifies the subscription associated with this notification.
resourceZoneId	M	0..1	Identifier	Identifies the resource zone in which the available NFVI capacity has crossed the threshold. Cardinality is 0 if the subscription does not specify a particular resource zone and the overall available NFVI capacity has crossed the threshold.
vimId	M	0..1	Identifier	Identifies the VIM in which the available NFVI capacity has crossed below the threshold. Cardinality is 0 if the subscription does not specify a particular VIM and the overall available NFVI capacity has crossed the threshold.
direction	M	1	Enum {UP; DOWN}	Specifies the direction the threshold has been crossed. VALUES: • UP • DOWN
capacityInformation	M	1	Not specified	Information about the available, reserved, allocated/used, and total capacity of the NFVI. If the subscription does specify a resource zone and/or vimId, the information is provided for the resource zone/vimId where the NFVI capacity has crossed the threshold.

8.7.3 NfviCapacityThreshold information element

8.7.3.1 Description

This information element provides the details of a NFVI capacity threshold.

8.7.3.2 Attributes

The attributes of the NfviCapacityThreshold information element shall follow the indications provided in table 8.7.3.2-1.

Table 8.7.3.2-1: Attributes of the NfviCapacityThreshold information element

Attribute	Qualifier	Cardinality	Content	Description
thresholdId	M	1	Identifier	Identifier of this NfviCapacityThreshold information element.
objectInstanceId	M	1	Identifier	Identifier of the VIM instance for which the threshold is defined.
thresholdType	M	1	Enum	Type of threshold. The list of possible values is part of the protocol design and might include: single/multi valued threshold, static/dynamic threshold, template based threshold, etc. VALUES: • SIMPLE: Single-valued static threshold • etc.
thresholdDetails	M	1	Not specified	Details of the threshold: value to be crossed, details on the notification to be generated.

8.8 Information elements and notifications related to Policy Management

8.8.1 Introduction

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

8.8.2 Information elements related to Policy Management Operations

8.8.2.1 Introduction

The clauses below define information elements related to policy management operations.

8.8.2.2 PolicyInfo information element

8.8.2.2.1 Description

This information element provides policy related information. It contains the policy itself and additional information related to the policy.

8.8.2.2.2 Attributes

The PolicyInfo information element shall follow the indications provided in table 8.8.2.2.2-1.

Table 8.8.2.2.2-1: Attributes of the PolicyInfo information element

Attribute	Qualifier	Cardinality	Content	Description
policyInfoId	M	1	Identifier	Identifier of policy information.
designer	M	1	String	Human readable name of designer of the policy.
name	M	1	String	Human readable name of the policy.
version	M	1	Version	Version of the policy.
policy	M	1	Not specified	Specifies the policy.
activationStatus	M	1	Enum	Status of the policy. VALUES: • ACTIVATED • DEACTIVATED

8.8.3 PolicyChangeNotification

8.8.3.1 Description

This notification indicates a change of a NFV-MANO policy related to operations of transferring policy, deleting policy, activating policy and deactivating policy.

Support of this notification is mandatory.

8.8.3.2 Trigger Conditions

The notification is produced when:

- A policy has been changed as a result of an operation of TransferPolicy, DeletePolicy, ActivatePolicy, DeactivatePolicy, AssociatePolicy or DisassociatePolicy.

8.8.3.3 Attributes

The attributes of the PolicyChangeNotification shall follow the indications provided in table 8.8.3.3-1.

Table 8.8.3.3-1: Attributes of the PolicyChangeNotification

Attribute	Qualifier	Cardinality	Content	Description
policyInfoId	M	1	Identifier (Reference to PolicyInfo)	Identifier of policy information.
operation	M	1	Enum	Policy management operation that causes the change of the policy. VALUES: • TRANSFER_POLICY; • DELETE_POLICY; • ACTIVATE_POLICY; • DEACTIVATE_POLICY; • ASSOCIATE_POLICY; • DISASSOCIATE_POLICY; • etc.

8.8.4 PolicyConflictNotification

8.8.4.1 Description

This notification indicates a policy conflict is detected by the NFVO. A policy conflict can include any conflicted monitored events, conditions or actions among two or more polices enforced by the NFVO.

Support of this notification is mandatory.

8.8.4.2 Trigger Conditions

The notification is produced when:

- A policy conflict is detected by the NFVO.

8.8.4.3 Attributes

The attributes of the PolicyConflictNotification shall follow the indications provided in table 8.8.4.3-1.

Table 8.8.4.3-1: Attributes of the PolicyConflictNotification

Attribute	Qualifier	Cardinality	Content	Description
policyInfoId	M	2..N	Identifier (Reference to PolicyInfo)	Identifiers of conflicted policy information.
conflictDescription	M	1	Not specified	Description of the detected policy conflicts, e.g. conflicted events, conditions or actions among the policies.

8.9 Information elements related to VNF Snapshot Package Management

8.9.1 Introduction

This clause defines information elements related to VNF Snapshot Package Management.

8.9.2 VnfSnapshotPkgInfo information element

8.9.2.1 Description

This information element provides the details of a VNF Snapshot Package, which the NFVO creates and stores as part of the ongoing operational VNF Snapshot Package management process.

8.9.2.2 Attributes

The VnfSnapshotPkgInfo information element shall follow the indications provided in table 8.9.2.2-1.

Table 8.9.2.2-1: Attributes of the VnfSnapshotPkgInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfSnapshotPkgInfoId	M	1	Identifier	Identifier of information held by the NFVO about a specific VNF Snapshot Package.
vnfSnapshotPkgId	M	0..1	Identifier	Identifier that identifies the VNF Snapshot Package. See notes 1 and 2.
name	M	1	String	Human-readable name of the VNF Snapshot Package.
checksum	M	0..1	Not specified	Checksum of the stored VNF Snapshot Package. See note 2.
createdAt	M	0..1	DateTime	Timestamp indicating when the VNF Snapshot Package creation has been completed. See note 2.
vnfSnapshotInfoId	M	0..1	Identifier (Reference to VnfSnapshotInfo)	Reference to the information held by the NFVO about a specific VNF Snapshot. This identifier was allocated by the VNFM. See note 2.
isFullSnapshot	M	1	Boolean	Value is 1 (true) in case of a "full" VNF Snapshot Package, i.e. containing all snapshotted VNFC instances; otherwise the value is 0 (false).
vnfd	M	0..1	Vnfd	VNFD of the snapshotted VNF instance that is contained in the stored VNF Snapshot Package. See note 2.
vnfInfo	M	0..1	VnfInfo	VnfInfo of the snapshotted VNF instance that is contained in the stored VNF Snapshot Package. See note 2.
vnfcSnapshotInfoId	M	0..N	Identifier (Reference to VnfcSnapshotInfo)	Identifier of information held by the VNFM about specific VNFC Snapshot(s). These identifiers were allocated by the VNFM. See note 2.

Attribute	Qualifier	Cardinality	Content	Description
vnfcSnapshotImage	M	0..N	VnfcSnapshotImageInfo	Information about VNFC Snapshot artifact(s) that are VNFC Snapshot Images. See note 2.
additionalArtifact	M	0..N	SnapshotPkgArtifactInformation	Information about VNF Snapshot artifact(s) that are not VNFC Snapshot images.
state	M	1	Enum	State of the VNF Snapshot Package. VALUES: • CREATED • BUILDING • UPLOADING • AVAILABLE • EXTRACTING • PROCESSING • ERROR
userDefinedData	O	0..N	KeyValuePair	User defined data for the VNF Snapshot Package.
accessInformation	M	0..1	Not specified	Information (such as a URL, or an identifier) that allows to access a copy of this VNF Snapshot Package. See note 2.

NOTE 1: This identifier identifies the VNF Snapshot Package in a globally unique way. It is created during the Build VNF Snapshot Package operation. Multiple instances of the same VNF Snapshot Package share the same vnfSnapshotPkgId.
NOTE 2: Cardinality is 0 when the VnfSnapshotPkgInfo was created but the VNF Snapshot Package was not yet built or uploaded.

8.9.3 SnapshotPkgArtifactInformation information element

8.9.3.1 Description

This information element represents an artifact other than a VNFC Snapshot Image which is contained in the VNF Snapshot Package.

8.9.3.2 Attributes

The SnapshotPkgArtifactInformation information element shall follow the indications provided in table 8.9.3.2-1.

Table 8.9.3.2-1: Attributes of the SnapshotPkgArtifactInformation information element

Attribute	Qualifier	Cardinality	Content	Description
selector	M	1	Not specified	Information (such as a path) that identifies/addresses this artifact in the VNF Snapshot Package.
metadata	M	1	Not specified	The metadata of the artifact that are available in the VNF Snapshot Package, such as content type, size, creation date, etc.

8.9.4 VnfcSnapshotImageInfo information element

8.9.4.1 Description

This information element represents VNFC Snapshot Image Information.

8.9.4.2 Attributes

The VnfcSnapshotImageInfo information element shall follow the indications provided in table 8.9.4.2-1.

Table 8.9.4.2-1: Attributes of the VnfcSnapshotImageInfo information element

Attribute	Qualifier	Cardinality	Content	Description
vnfcSnapshotImageId	M	1	Identifier	The identifier of this VNFC Snapshot image.
name	M	1	Not specified	The name of this VNFC Snapshot image.
checksum	M	1	Not specified	The checksum of the VNFC Snapshot image file.
vnfcInstanceId	M	1	Identifier	Identifier of the snapshotted VNFC instance that this VNFC Snapshot image belongs to.
containerFormat	M	1	Not specified	The container format indicates whether the VNFC Snapshot image is in a file format that also contains metadata about the actual snapshot.
diskFormat	M	1	Not specified	The disk format of a VNFC Snapshot image is the format of the underlying disk image.
createdAt	M	1	DateTime	The time when this VNFC Snapshot image creation has been completed.
minDisk	M	1	Not specified	The minimal Disk for this VNFC Snapshot image.
minRam	M	1	Not specified	The minimal RAM for this VNFC Snapshot image.
size	M	1	Not specified	The size of this VNFC Snapshot image.
userMetadata	M	0..N	KeyValuePair	User-defined metadata.
accessInformation	M	1	Not specified	Information such as a path (if the image is included in the VNF Snapshot Package), or an URL or identifier (if the image is not included in the VNF Snapshot Package) that allows to access a copy of this VNFC Snapshot Image.

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

The following bullets list the main principles related to VNF lifecycle management and NS lifecycle management:

1) **Principle #1:** The VNFM offers a layer of decoupling for VNFs:

   - VNF lifecycle management is handled by the VNFM corresponding to a given VNF.
   - The VNF can be modified without a call though the VNF lifecycle management interface, for instance in reaction to some alarms/metrics resulting in a scaling. So the FBs using the service of the VNF lifecycle management interface, like NFVO, should expect that some changes are possible for a given VNF. These changes are communicated through notifications.
   - This does not prevent NFVO from being able to send VNF lifecycle management operations to the VNFM.

2) **Principle #2:** The NFVO offers a layer of decoupling for NSs:

   - NS lifecycle management is handled by the NFVO corresponding to a given NS.
   - The NS can be modified without a request from the OSS/BSS through the NS lifecycle management interface, either because of a VNF change or because the NFVO has reacted to some event and decided to change something in the NS or in a VNF that is part of an NS. So the OSS/BSS should expect that some changes in the NS (and the VNFs that are part of it) are possible. These changes are communicated through notifications.
   - This does not prevent OSS/BSS from being able to send directly NS lifecycle management operations to the NFVO.

3) **Principle #3:**

   - With respect to the Os-Ma-nfvo reference point, any interaction concerning a VNF is associated with at least one NS instance.

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

B.1 Re-use of VNFs from a terminated NS

In this use case, the consumer (OSS/BSS FB) requests that the provider (NFVO) terminate a given NS instance, and the NFVO retains some of the VNF instances for use in future NS instances (in NS instantiate and/or NS update).

NOTE: The OSS/BSS FB can instruct the NFVO to retain VNF instances by adding the VNF instances in another NS (e.g. a VNF Pool NS).

Some additional points to consider with regard to the reuse of VNF instances:

- At the time the NS instance is terminated, the consumer may not know what specific NS instances are to use the retained VNF instances but only that the VNF instances should be retained. The VNF instances to be retained need to be part of another NS instance (e.g. a VNF Pool NS) before terminating the NS instance to which those VNFs belong.
- The retained VNF instances are of various types, could be a relative large number (consider an NS type such a virtual CDN) and may be reused in multiple NS instances (which may not even be of the same type as the original NS).
- A retained VNF instance may need to be updated before being used in a new NS instance.
- Some reasons for VNF instance retention:
  - The particular type of VNF may take some "cost" (e.g. time) to instantiate and it is easier to retain the VNF instances for future use.
  - There is a known demand for a given type of VNF and it is easier to just retain the VNF instance rather than to terminate and wait for when it needs to be re-used (which might be immediately).
  - As part of some sort of NS instance migration, it may be necessary (or easier) to retain some of the VNF instances from the NS instance that is to be migrated.
  - The VNF instance is part of a leased network segment (allocated to a specific consumer) and whether to retain the VNF instance or not is basically a decision of the consumer.

B.2 Creation of VNF instances in anticipation of future NS demand

Based on (for example, customer demand forecasts) VNF instances are created in advance for use in NS instances at some later point in time. This might be done because instances of the given type of VNF may take some time to instantiate and the OSS/BSS FB wants to ensure quick instantiation of the NS instances that use these VNF instances.

In a variation of this use case, the OSS/BSS FB may create VNF instances in advance for use in pre-ordered NS instances (for some future date by a customer). In this case, the OSS/BSS FB could first request instantiation of the required resources (e.g. VNF instances) in a context of a holding NS (e.g. a VNF Pool NS), and then request the NS instance later when it is needed, with the knowledge that the required VNF instances (for example) are there and ready to be used immediately.

B.3 Bottom-up NS instantiation

It may be that VNF instances are created in advance for a given NS instance, with the knowledge that some of the VNF instances will take longer to create than others and that an immediate NS instantiate request will fail unless all the required VNF instances are instantiated beforehand.

B.4 Shared VNF instances

Some VNF instances can be shared by several NS instances. In some cases, the OSS/BSS FB will indicate to the NFVO that a VNF instance (already being used by at least one NS instance) is be used in another NS instance.

A shared VNF instance may need to be updated before it is reused in another NS instance.

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

C.1 Introduction

This annex describes how the use cases presented in annex B can be supported by fine-grain control over VNF instances in NS lifecycle management operations.

The approach uses the basic NS lifecycle management operations and within the NS lifecycle management operations additional attributes are created to support the fine grained VNF lifecycle management tasks.

Table C.1-1 provides the mapping of all identified use cases and proposed solutions.

Table C.1-1

Existing Use Cases in annex B	Solutions in annex C
Clause B.1 Re-use of VNFs from a terminated NS	"Terminating an NS with retained VNF instance(s)"
Clause B.2 Creation of VNFs in anticipation of future NS demand	"VNF pool creation"
Clause B.3 Bottom-up NS instantiation	"VNF pool creation" (steps 6 and 7)
Clause B.4 Shared VNF instances	"Terminating an NS with retained VNF instance(s)" (steps 1a, 1b and 2) and "New NS with existing VNF instance(s)"

C.2 New NS with VNF pools

A VNF pool NS can be utilized for the use case of creation of VNF instances in anticipation of future NS demand. (The associated use case is defined in clause B.2.)

The VNF Pool NS is just a normal NS which contains VNF instances and may not have any connectivity between the VNF instances. There are many ways how the VNF Pool NS can be "filled up" with VNF instances. VNFs can be instantiated when an NS is instantiated or OSS/BSS can add VNF instances to the NS instance by the Update NS operation or OSS/BSS can associates existing VNF instances to the NS with the Update NS operation.

Figure C.2-1 shows the flow how a VNF pool NS can be created.

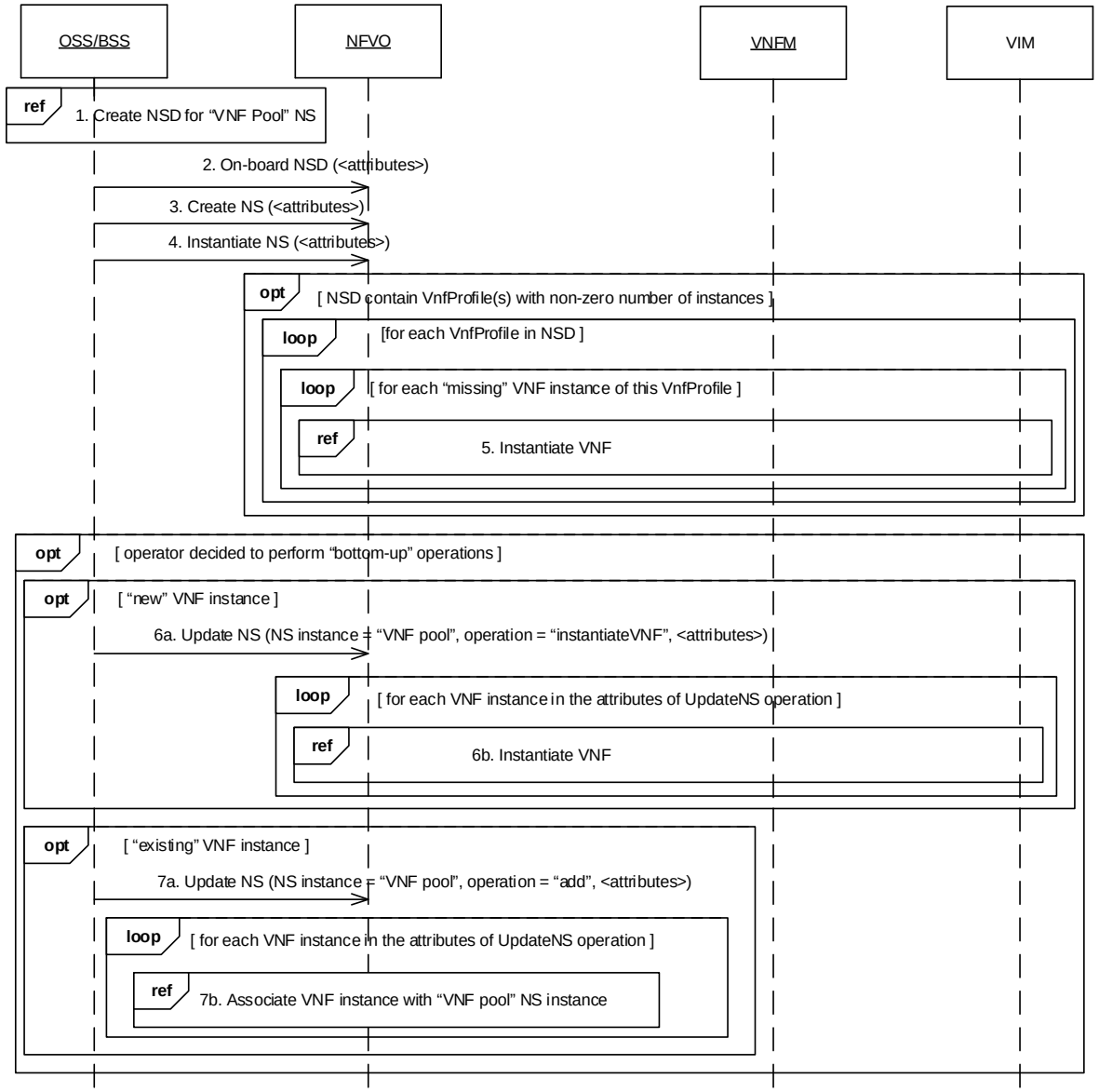

Figure C.2-1: VNF Pool NS creation

1) As a pre-requisite for instantiation of an NS, the corresponding NSD has to be created.

2) The NSD is on-boarded to the NFVO. It is assumed that all VNF Packages who's VNFDs are referred in the NSD have already been on-boarded to the NFVO.

3) The OSS/BSS issues the Create NS operation to the NFVO.

4) The OSS/BSS issues the Instantiate NS operation to the NFVO.

5) If the NSD contains VNF Profiles with a non-zero number of VNF instances to be instantiated in the NS, then the NFVO performs the instantiation procedure for each VNF that requires instantiation.

6) When the NS is instantiated and an NS instance ID is available, the OSS/BSS may request additional VNF instances to be added to the "VNF pool" NS instance either by instantiating them (steps 6a and 6b) or by associating the existing VNF instances with the "VNF pool" NS instance (steps 7a and 7b). This allows the operator to perform a "bottom-up" NS instantiation - complete (if no VNF instances were instantiated in step 5, according to the numbers in the NSD) or partial (in addition to those VNF instances that were instantiated in step 5).

6a) In this operation the OSS/BSS tells what new VNF instances should be added to the "VNF pool" NS instance.

6b) The NFVO performs the VNF instantiation(s) according to the request received from the OSS/BSS.

7a) In this operation the OSS/BSS tells what existing VNF instances(e.g. those that are associated with other NS instances) should be added (or possibly removed) to (from) the "VNF pool" NS instance.

7b) The NFVO associates the VNF instance to the "VNF Pool" NS instance according to the request received from the OSS/BSS.

C.3 New NS utilizing VNF instances from a VNF pool NS

In this clause the flow is described how an NS can be built up from existing VNF instances.

The sequence diagram of the case is shown in figure C.3-1.

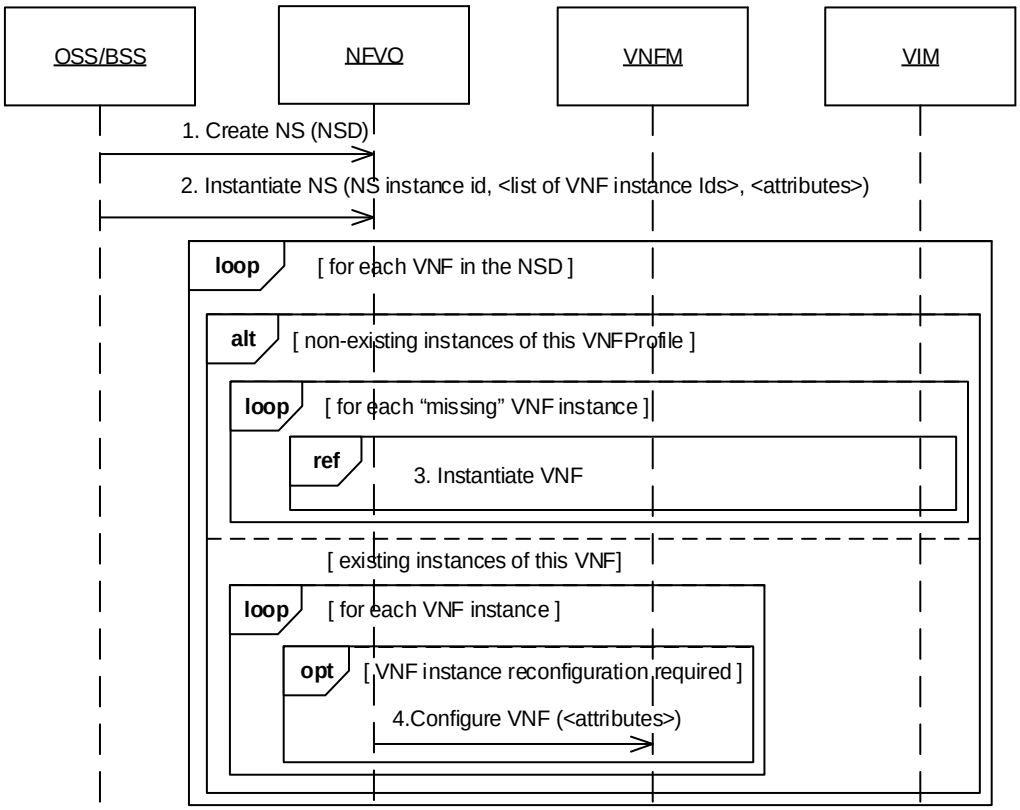

Figure C.3-1: Creating NS with existing VNF instance(s)

0) It is assumed, that there is already an existing "VNF pool" NS instance with some VNF instances in it (see "VNF Pool Creation" use case for details). It is also assumed, that the NSD of a "real" NS has been created and on-boarded to the NFVO.

1) The OSS/BSS requests the creation of an NS instance for the "real" network service.

2) The OSS/BSS requests the instantiation of the NS. As part of the Instantiate NS operation the OSS/BSS can indicate the VNF instances (by listing their ids) to be used in the newly instantiated NS. These VNFs become shared between the "VNF Pool" NS and the "real" NS that is being instantiated.

3) The NFVO performs VNF instantiation procedure for each VNFs that is "missing" (does not have an already existing instance) according to the NSD of the NS.

4) The NFVO may need to request VNFM to reconfigure the "existing" VNF instances (e.g. those from the "VNF pool") according to the attributes of the NS lifecycle management operation.

C.4 Terminating NS instance with retained VNF instances

In this scenario an NS is terminated, while certain VNF instances need to be retained. The VNFs to be retained are added to the VNF Pool NS before the "normal" NS is terminated. The flow is described in figure C.4-1.

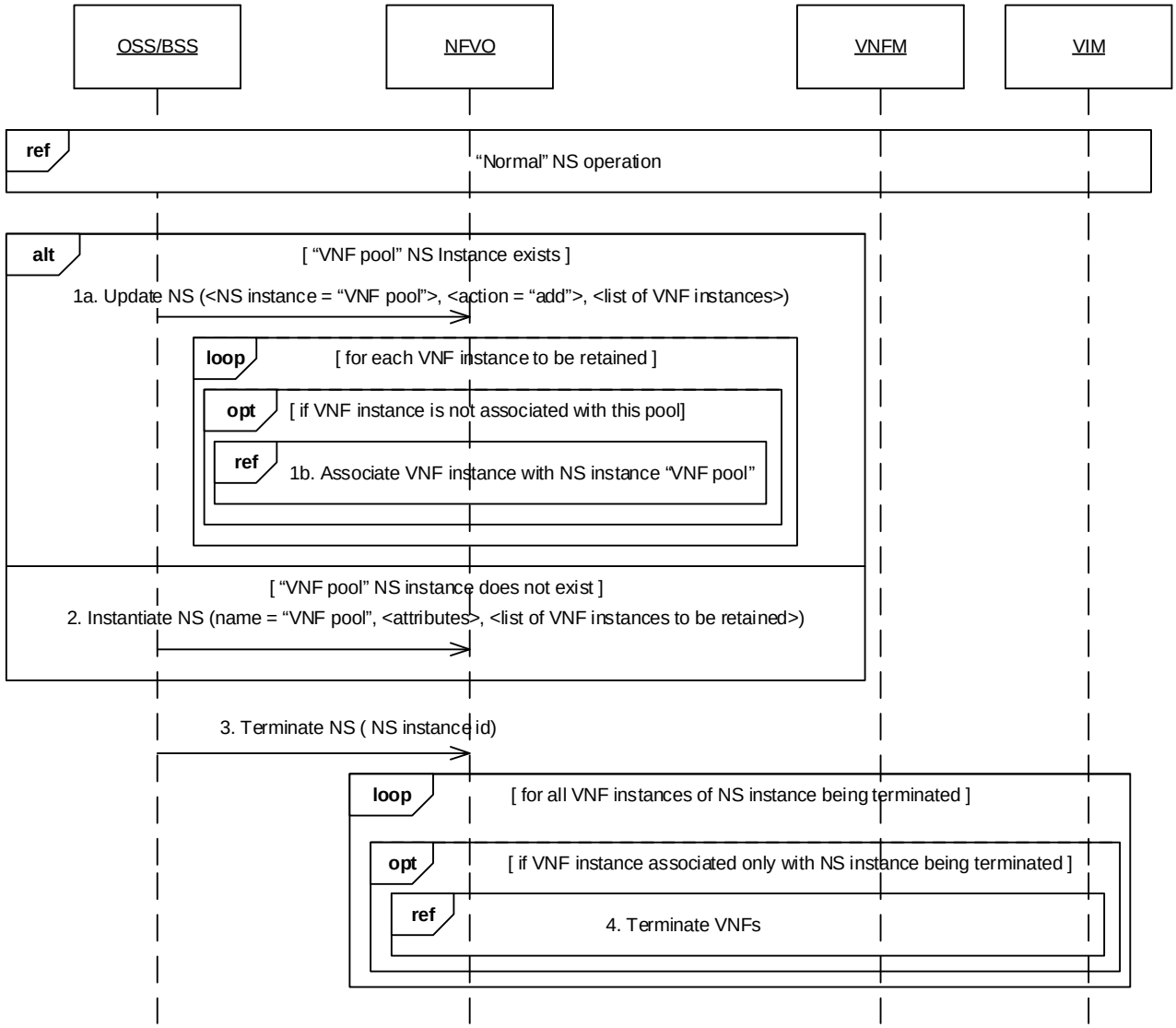

Figure C.4-1: Terminating an NS with retained VNF instance(s)

0) It is assumed, that there is an existing instance of an NS.

1a) If operator decides to use an existing "VNF pool" NS instance to retain the VNF instances from the NS being terminated, then the OSS/BSS issues an Update NS operation indicating the NS instance Id to be the "VNF Pool" NS and the list of VNF Instance Ids (to be retained) that should be added to the "VNF Pool" NS. Based on operator's decision, there could be multiple "VNF pool" NS instances (e.g. one per VNF type or one per operator's intention) - in such cases, there will be a separate Update NS operation per "VNF pool" NS instance.

1b) The NFVO associates each VNF instance being retained to the "VNF Pool" NS instance, according to the Update NS operation.

2) If operator decides to use a new "VNF Pool" NS instance, then OSS/BSS needs to create one. In the Instantiate NS operation, OSS/BSS can provide the list of VNF instance Ids to be retained.

3) OSS/BSS issues the Terminate NS operation indicating the NS instance Id to be terminated. The VNF instances being retained are already associated with more than one NS (the NS instance being terminated and "VNF pool" NS instance).

4) The NFVO terminates all those VNF instances associated only with the NS instance that is being terminated.

Annex D (informative): State models

D.1 VNF Package state model

D.1.1 Introduction

This annex proposes a state model a given VNF Package in the NFVO, including both on-boarding phase and operational phase.

All the steps before the on-boarding of the VNF Package are not part of this state model.

D.1.2 State model

A given VNF Package has three states, i.e. on-boarding state, operational state and usage state.

The on-boarding state is represented by the "onboardingState" attribute in the "VnfPkgInfo" information element with below values:

- CREATED: The VNF Package information object is created.
- UPLOADING: The VNF Package is being uploaded.
- PROCESSING: The VNF Package is being processed, e.g. validation.
- ONBOARDED: The VNF Package is successfully on-boarded.

The operational state is represented by the "operationalState" attribute in the "VnfPkgInfo" information element with below values:

- ENABLED: The VNF Package is enabled.
- DISABLED: The VNF Package is disabled.

The usage state is represented by the "usageState" attribute in the "VnfPkgInfo" information element with below values:

- IN_USE: The VNF Package is in use.
- NOT_IN_USE: The VNF Package is not in use.

The state model of on-boarding phase in figure D.1.2-1 applies to a given VNF Package being on-boarded. Besides the operations and conditions specified in the figure, below operations are also considered as available during the on-boarding phase:

- Query VNF Package Info.
- Update VNF Package Info (with user defined data only).

The state model of operational phase in figure D.1.2-1 applies to an on-boarded VNF Package. Besides the operations and conditions specified in the figure, below operations are also considered as available during the operational phase:

- Query VNF Package Info.
- Update VNF Package Info (with user defined data only).
- Fetch VNF Package.
- Fetch VNF Package Artifacts.

The "onboardingState" details the state changes during the VNF Package on-boarding phase. The value of this attribute during the VNF Package operational phase is "ONBOARDED".

The "operationalState" and "usageState" detail the state changes during the VNF Package operational phase. During the VNF Package on-boarding phase, the value of the "operationalState" is "DISABLED" and the value of the "usageState" is "NOT_ IN_USE". Right after the VNF Package becomes on-boarded, the value of the "operationalState" is changed to "ENABLED" and the value of the "usageState" is kept as "NOT_ IN_USE", as shown in figure D.1.2-1.

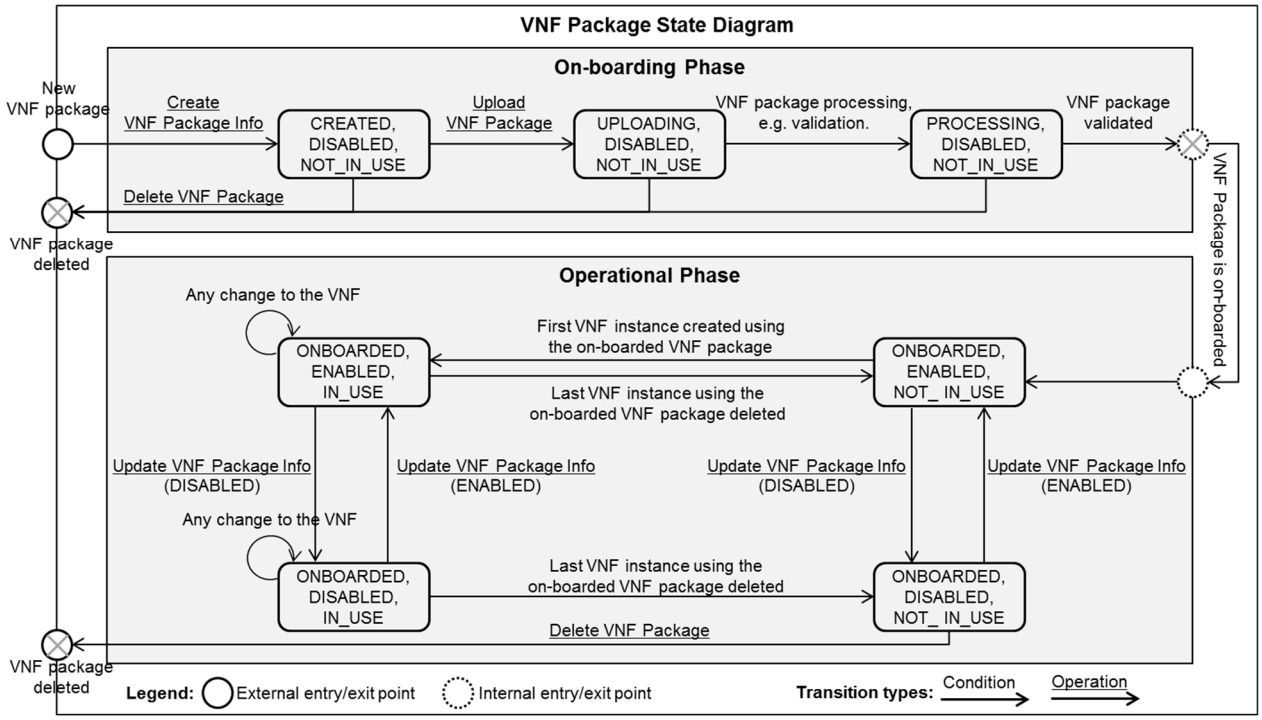

Figure D.1.2-1: VNF Package state model

D.2 NSD state model

D.2.1 Introduction

This clause proposes a state model for a given NSD in the NFVO, including both on-boarding phase and operational phase.

D.2.2 State model

A given NSD has three states, i.e. on-boarding state, operational state and usage state.

The on-boarding state is represented by the "onboardingState" attribute in the "NsdInfo" information element with below values:

- CREATED: The NSD information object is created.
- UPLOADING: The NSD is being uploaded.
- PROCESSING: The NSD is being processed, e.g. validation.
- ONBOARDED: The NSD is successfully on-boarded.

The operational state is represented by the "operationalState" attribute in the "NsdInfo" information element with below values:

- ENABLED: The NSD is enabled.
- DISABLED: The NSD is disabled.

The usage state is represented by the "usageState" attribute in the "NsdInfo" information element with below values:

- IN_USE: The NSD is in use.
- NOT_IN_USE: The NSD is not in use.

The state model of on-boarding phase in figure D.2.2-1 applies to a given NSD being on-boarded. Besides the operations and conditions specified in the figure, below operations are also considered as available during the on-boarding phase:

- Query NSD Info.
- Update NSD Info (with user defined data only).

The state model of operational phase in figure D.2.2-1 applies to an on-boarded NSD. Besides the operations and conditions specified in the figure, below operations are also considered as available during the operational phase:

- Query NSD Info.
- Update NSD Info (with user defined data only).
- Fetch NSD.

The "onboardingState" details the state changes during the NSD on-boarding phase. The value of this attribute during the NSD operational phase is "ONBOARDED".

The "operationalState" and "usageState" detail the state changes during the NSD operational phase. During the NSD on-boarding phase, the value of the "operationalState" is "DISABLED" and the value of the "usageState" is "NOT_ IN_USE". Right after the NSD becomes on-boarded, the value of the "operationalState" is changed to "ENABLED" and the value of the "usageState" is kept as "NOT_ IN_USE", as shown in figure D.2.2-1.

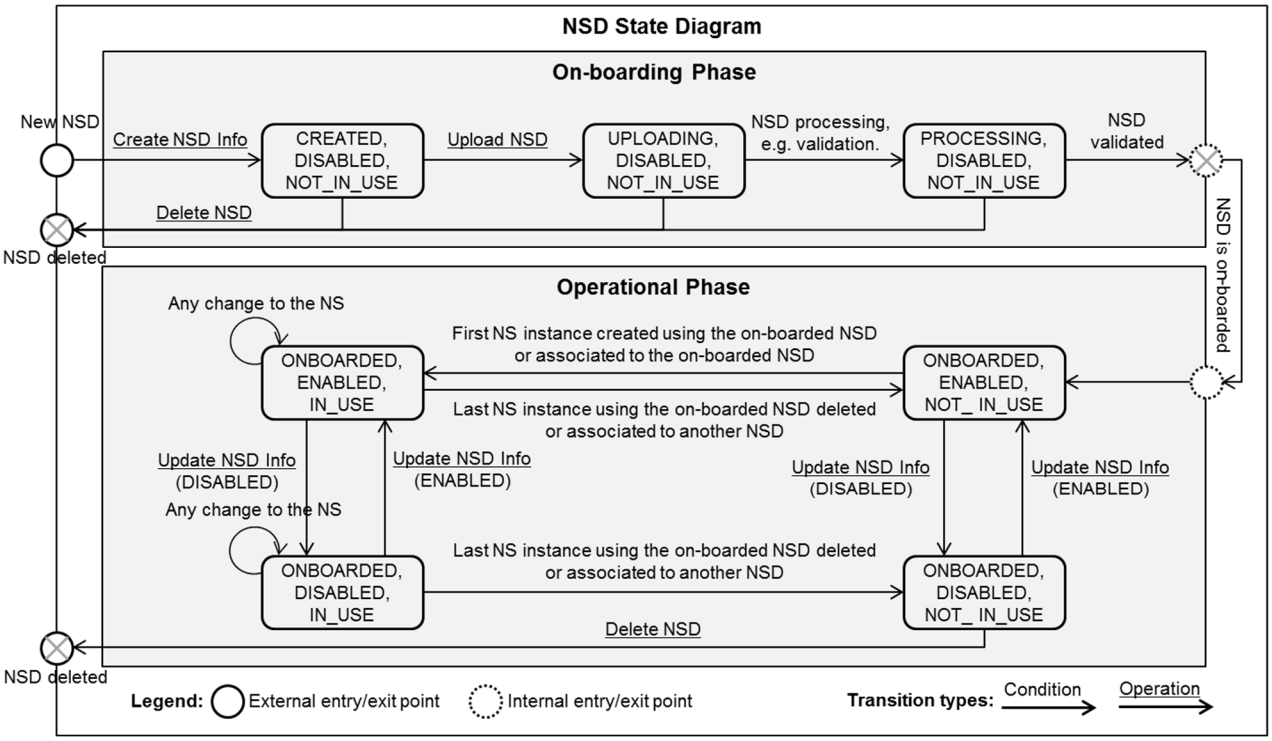

Figure D.2.2-1: NSD state model

D.3 NS state model

D.3.1 Introduction

This clause proposes a state model for the NS instance in the NFVO.

All the steps before the initial Create NS are not part of this state model.

D.3.2 State model

A given NS instance has 2 elementary state values in the NFVO: INSTANTIATED, NOT_INSTANTIATED, INSTANTIATED.

The state model, shown in figure D.3.2-1, applies to a given NS instance.

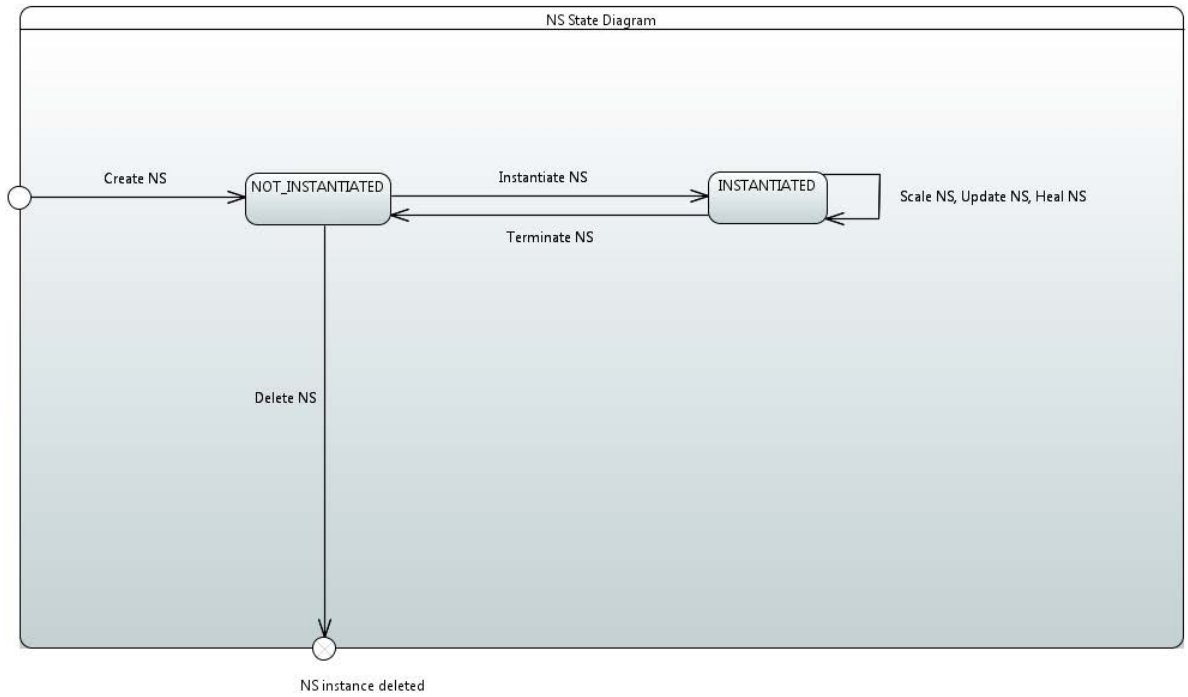

Figure D.3.2-1: Network Service instance state model

The query NS operation is considered as available in all the states above.

Annex E (informative): NS scaling

E.1 Forms of NS scaling

The aim of NS scaling is to increase or decrease the capacity of a network Service instance. This can be achieved in various ways:

- Scaling a VNF instance: if a constituent VNF instance in the NS instance is not yet scaled to its limits. However it is also possible that either the VNF instance is already scaled to is limits or the VNF instance does not support scaling, then VNF scale cannot be used as part of NS scale operation.
- Changing the DF of a VNF instance. If a VNF is already scaled to its limits, then the next step of increasing the capacity of a VNF is to use the DF change (if there is a higher/lower capacity DF of the VNF). If the DF change of the VNF requires topology changes, NFVO can create/modify the required VLs. Also the VNF DF changes may require application level configuration task, which is done by the traditional management system therefore interaction with OSS/BSS or EM may be required.
- Adding/removing VNF instance(s) to/from the NS: In this scenario the capacity of the NS is changed by adding/removing VNF instance(s) to/from the NS instance. In this scenario the NFVO need to take care of creating the necessary links between the VNF instances according to the connectivity requirements defined in the NSD. This type of NS scaling may also require application level configuration task, which needs to be performed by the traditional management system (by OSS/BSS or EM).
- Scaling to a new NS DF: In this scenario the NS DF is changed to a new one, which contains higher/lower capacities. The NFVO may requires instantiation/termination of VNF instances according to the NSD of the new NS DF.
- Scaling a nested NS: The capacity of an NS can also be changed by changing the capacity of a nested NS if the network service contains a nested network service.
- Scaling of a VL: It may also be necessary as part of the capacity change need of an NS to change the capacity of a VL in an NS. This may be achieved either by changing the properties of a VL or by adding/removing VL in an NS. The latter may require application level configuration as well, therefore interaction with OSS/BSS may be required.

E.2 NS scaling triggers

E.2.1 NS auto-scale

In this case the NS scale decision is made at NFVO based on the information provided in the NSD. The main attributes to be used for the NS auto-scale functionality is the monitoring parameter and the indicators that are re-exposed in the Or-Vnfm reference point specified in the NSD and the associated auto-scale rule.

E.2.2 NS scale triggered by OSS/BSS

In this case the NS scale operation is requested by OSS/BSS via the Os-Ma-nfvo reference point. OSS/BSS can specify what to scale by providing the scale rules to NFVO (similar to the auto-scale rules defined in the NSD. The OSS/BSS may also provide explicit guidance to the NFVO what to scale and in what way, i.e. the OSS/BSS may tell to NFVO to scale a specific VNF instance to a specific scale level.

E.3 Relation to NS DF

The NS scale operates either within the boundaries of a network service DF as specified in the NSD or by changing the NS DF. The NS DF provides the minimum and maximum number of instances of each VNFs the NS is built upon. Each VNF in the NSD references to a specific VNF DF as specified in the VNFD.

As a consequence of the above an NS scale can operate via a VNF scale within the boundaries of a VNF DF as specified in the VNFD or by adding/removing VNF instances within the boundaries of an NS DF as specified in the NSD.

The capacity of an NS may be changed by changing or moving from one NS DF to another DF.

If an NS contains a nested NS, the change of the capacity can be achieved by scaling the nested NS.

E.4 Input and tools for NS auto-scaling

E.4.1 Monitoring parameter

Monitoring parameters are defined in NSD and may be used to trigger the necessary NS scaling actions at NFVO. Monitoring parameters can specify the values/threshold of a PM counter that is available at NFVO (e.g. derived from virtualised resource performance metrics) and the associated auto-scaling rules/policies.

E.4.2 VNF indicator

VNF indicators are declared in the VNFD and are provided either by the VNF or by the EM managing the VNF. These VNF indicators are forwarded to the NFVO by the VNFM managing the VNFs. These VNF indicators may also be used by the NFVO for its NS auto-scale functionality.

E.4.3 Auto-scale policies/rules

This should define the required scaling actions based on the monitoring parameters and/or VNF indicators. It should define in priority order e.g. what VNFs to scale or whether a VNF instance to be added or removed to the NS instance. In case of VNF scale it should specify also the Scale aspect of the VNF scale operation that can be used from NFVO towards the VNFM.

Annex F (informative): Example interaction flows

F.1 LCM Coordination flow

Figure F.1-1 illustrates the use of LCM Coordination interface (specified in clause 6.1).

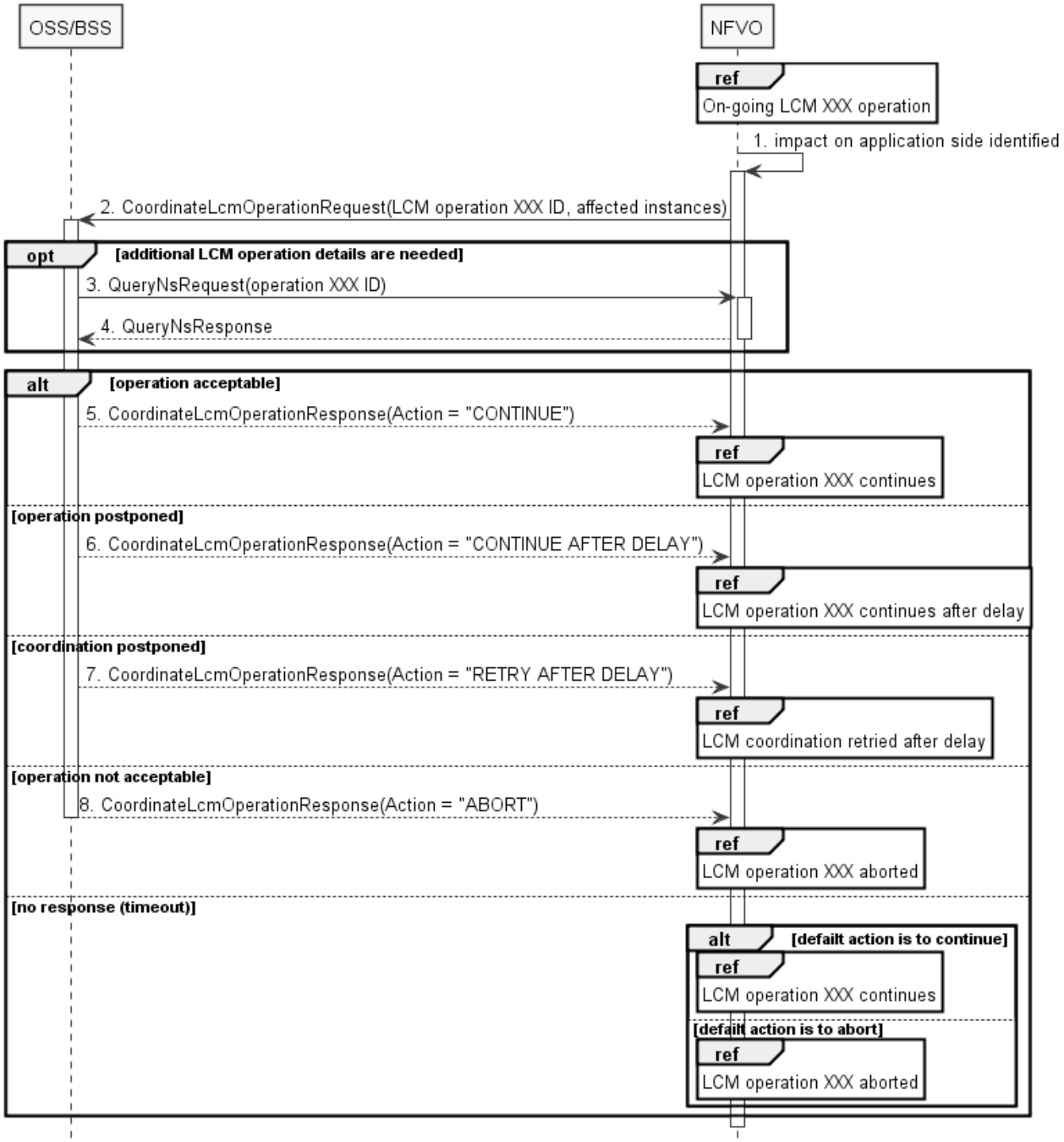

Figure F.1-1: LCM Coordination example interaction flow

Annex G (informative): Change history

Date	Version	Information about changes
9 October 2017	V2.3.2	NFVIFA(17)000839r3, IFA013ed241: Refactor the NSD Management Interface NFVIFA(17)000762r1, IFA013 Change of description of nsdId attribute in NsInfo NFVIFA(17)000682r1, IFA013ed241: Rename "NsLifecycleChangeNotification" to "NsLcmOperationOccurrenceNotification" NFVIFA(17)000737, IFA013ed231: Enhance NS Fault Management Interface NFVIFA(17)000742r1, IFA013ed241: Update Alarm IE on NS Fault Management Interface
November 2017	V2.3.3	NFVIFA(17)000805r2, IFA013 Adding NfpRule specification NFVIFA(17)000876r1, IFA013ed241: Refactor NSD management interface - fixing the consistency issues NFVIFA(17)000903, IFA013ed241: Add NS fault management related requirements NFVIFA(17)000931r1, IFA013ed241: NsInfo information element fix NFVIFA(17)000941r1, IFA013ed241: Clarify actionsHealing parameter in the HealNsData information element NFVIFA(17)000943, IFA013ed241: Harmonize NFP related Information Elements NFVIFA(17)000859r5, IFA013ed241: Refactor the VNF Package Management Interface NFVIFA(17)000878r3, IFA013ed241: Refactor PNFD management
December 2017	V2.3.4	NFVIFA(17)000932r3, IFA013ed241: Update NsLcmOperationOccurrenceNotification IE NFVIFA(17)0001040r3, IFA013ed241: Update ResourceHandle IE NFVIFA(17)0001080r13, IFA013v020303_Add_PNF_Updating_Operation_in_Update_NS_Operation_And_Add_New_IEs_For_PNF NFVIFA(17)0001092, IFA013ed241 Add fetching and updating related interface requirements NFVIFA(17)0001093, IFA013ed241 Correction on inconsistent change of on-board NSD and VNF package NFVIFA(17)0001096, IFA013ed241: Query filter design on NSD and VNF package management Interface NFVIFA(17)0001097, IFA013ed241: Remove the port number from the SapData/SapInfo/PnfExtCpInfo IE NFVIFA(17)0001098, IFA013ed241: Update the NSD state diagram NFVIFA(17)0001103, IFA013ed241 Clarification of ExtManagedVirtualLink and ExtVirtualLink NFVIFA(17)0001104r2, IFA013ed241 Interface requirement fix for update NSD/VNF package operations NFVIFA(17)0001105, IFA013ed241 Align description of VNF Package Change Notification NFVIFA(17)0001113r3, IFA013ed241 - Corrections related to multiple layer protocol support NFVIFA(17)001118r1, IFA013ed241 Correction on notes in clause 5.3.2 NFVIFA(17)001119, IFA013ed241 Correction on notes in clauses 8.2.2.2 and 8.2.4.2 NFVIFA(17)0001132r1, IFA013ed241 Correction on the name of NS lifecycle change notification interface NFVIFA(17)0001095r3, IFA013ed241: Add new NS Change Notification on NS LCM interface NFVIFA(17)0001117r1, IFA013ed241 LCM operation response and notification in operation results (mirror of 1111) NFVIFA(17)0001139r1, IFA013ed241: Add Notification for PNFD Management NFVIFA(17)0001145r1, IFA013ed241 - Clarifications on NsVL, NsLinkPort and ExtVL, ExtLinkPort NFVIFA(17)0001150r1, IFA013ed241: Add "changedVnfInfo" parameter to the AffectedVnf information element NFVIFA(17)001152, IFA013ed241 Remove redundant description of vnfConfigurableProperty NFVIFA(17)0001153r1, IFA013ed241: Mirror from IFA#1037 about external connectivity and address data NFVIFA(17)001008r2, IFA013ed241: Update AffectedXXX IE NFVIFA(17)0001165, IFA013ed241: Rename NsVirtualLink and NsLinkPort
March 2018	V2.4.2	NFVIFA(18)000171, IFA013ed251 - Add profile id for NsVirtualLinkInfo NFVIFA(18)000176, IFA013ed251 - Remove IFA012 references
May 2018	V2.4.3	NFVIFA(18)000191r1, IFA013ed251 Add monitoringParameter attribute to NsInfo IE
May 2018	V3.0.0	V3.0.0 created from V2.4.3 NFVIFA(18)000119r1 already included as NFVIFA000171 NFVIFA(18)000177r1 already included as NFVIFA(18)000176
May 2018	V3.0.1	NFVIFA(18)000422r2, IFA013 MegaCR FEAT04 Compute Host Reservation
June 2108	V3.0.2	NFVIFA(18)000429r1, FEAT07 IFA013 MegaCR Support of policy management interface

Date	Version	Information about changes
June 2018	V3.0.3	NFVIFA(18)000534, IFA013ed311 - Mirror - Linking VNFC CP and VnfExt CP NFVIFA(18)000537r1, IFA013ed311 - Mirror - Clarifying association from VnfLinkPort to VnfcCp and VnfExtCp NFVIFA(18)000566, IFA013ed311: Remove the current values of the monitoringParameter attribute from the NS LCM interface NFVIFA(18)000603, IFA013ed311 Rel3Mirror different names for virtual link descriptor ids NFVIFA(18)000616r1, IFA013ed311 - Rel3Mirror - Fixing sentence related to PM delivery mechanism NFVIFA(18)000625, IFA013ed311 Rel3Mirror Add NestedNsInstanceData for NS LCM operation NFVIFA(18)000626, IFA013ed311 Rel3Mirror Add ParamsForNestedNs for InstantiateNs operation NFVIFA(18)000639r2, IFA013ed311 - Fixing NFP management NFVIFA(18)000641, IFA013ed311 Rel3Mirror of 435r2 metadata for CP IEs NFVIFA(18)000541r1, IFA013 MegaCR FEAT15 VNF Snapshot
August 2018	V3.1.2	NFVIFA(18)000719r4: associate PNF or VNF instance with Profile
November 2018	V3.1.3	NFVIFA(18)000819 - IFA013ed321 Use on vnfProfile in Instantiate NS operation NFVIFA(18)000836 - IFA013ed321 Mirror of 834 (IFA007ed321 Mirror for SOL contribution on making the API surface consistent for bootData) NFVIFA(18)000883r1 - IFA013ed321 Fix bug 7760: Inconsistency between IFA013 and SOL005 NFVIFA(18)000896 - IFA013ed321 Add pfId in Policy Transfer operation NFVIFA(18)000889r2 - IFA013ed321_update_of_IEs_related_to_PM_interface_for_IFA027
December 2018	V3.1.4	NFVIFA(18)000961_IFA013ed321_Metadata_Extension_ConfigurableProps_clarificati.docx NFVIFA(18)0001001_IFA013ed321_CR_add_policy_associate_disassociate_operations.docx NFVIFA(18)0001083r1 - IFA013ed321 - Add LCM Coordination interface NFVIFA(18)0001057 - IFA013ed321 Fix bug 7815 Superfluous word NFVIFA(18)0001034 - IFA013 VNF snapshot createdAt and userDefinedData
January 2019	V3.1.5	NFVIFA(19)000021 - IFA013ed321 small improvements in Clause 4-1 NFVIFA(19)000028 - IFA013 Small bugfix clause 538 NFVIFA(19)000014r1 - IFA013 MegaCR FEAT05 Slicing
February 2019	V3.1.6	NFVIFA(18)0001110r4 - FEAT10 IFA013 MegaCR NFVIFA(19)000077r3 - FEAT02 IFA013 Adding Changing VNF Package NFVIFA(19)000121 - IFA013ed321_editorial_improvements_for_DateTime Some editorial changes
February 2019	V3.1.7	NFVIFA(19)000137r2 - IFA013ed321 PNF Package operations NFVIFA(19)000138 - IFA013ed321 Fix references to other documents NFVIFA(19)000148r1 - FEAT02 IFA013 mirror of 142r2 NFVIFA(19)000159r1 - IFA013ed321 Clause 8-3-2-3 adding changeType values
May 2019	V3.2.2	Base Line for Release 3 Drop 3 created from published version 3.2.1
June 2019	V3.2.3	NFVIFA(19)000362r2 - IFA013ed331 Mirror - Correction cardinality VnfcSnapshotInfo and description VnfSnapshotInfo NFVIFA(19)000508 - IFA013 Alignment with Stage 3 work on VNF snapshot feature NFVIFA(19)000546r1 - IFA013 capacity threshold management NFVIFA(19)000552r2 - IFA013ed331 capacity threshold management
July 2019	V3.2.4	NFVIFA(19)000584 - IFA013ed331 Fix bug 7850 (correctly_implement_639r2) NFVIFA(19)000623 - IFA013ed331 Rel3Mirror 8.3.4.13 ExtVirtualLinkData IE NFVIFA(19)000637 - IFA013ed331 VnfLinkPortInfo IE NFVIFA(19)000638 - IFA013ed331 VnfSnapshotPkgInfo IE NFVIFA(19)000642 - IFA013ed331 Rel3Mirror 8.3.4.36 VnfExtCpConfig IE NFVIFA(19)000666 - IFA013ed331 Rel3Mirror 8.6.4 VnfPackageSoftwareImageInformation IE Some editorial changes
October 2019	V3.3.2	Initial version for maintenance Fix capitalization of OS-Ma-Nfvo to Os-Ma-nfvo
October 2019	V3.3.3	NFVIFA(19)000762 - IFA013ed341 measurementContext in ThresholdCrossedNotification NFVIFA(19)000809r1 - IFA013ed341 improving several IE descriptions NFVIFA(19)000819 - IFA013ed341 modifying VNF package references
November 2019	V3.3.4	NFVIFA(19)000850 IFA013ed341 Support upload of VNF Package as separate files NFVIFA(19)000853r1 IFA013ed341 Clarifying usage of pfId in Transfer Policy operation NFVIFA(19)000862 IFA013ed341 Provide additional description in VnfInfo NFVIFA(19)000885 IFA013ed341 mirror of 825r1 exposing maxScaleLevels NFVIFA(19)000889 IFA013ed341 mirror of 841 relaxing PM subscriptions NFVIFA(19)000892 IFA013-Remove Annex -Authors and contributors

Date	Version	Information about changes	
December 2019	V3.3.5	NFVIFA(19)000874r3	IFA013ed341 Enhancements in ChangeExtVnfConnectivity
		NFVIFA(19)000936	IFA013ed341 FEAT10 Bugs and clarifications multi-site connectivity
		NFVIFA(19)000949	IFA013ed341 Add update VNF snapshot package operation
		NFVIFA(19)000944r5	IFA013 Improve wording left for protocol design stage
		NFVIFA(19)000925r4	IFA013ed341 Support for dynamic creation and deletion of NsVirtualLink instances
January 2020	V3.3.6	NFVIFA(20)000001	IFA013ed341 mirror of 1004 adding vnfConfProps to ChangeCurrentVnfPackage
		NFVIFA(20)000022r1	IFA013ed341 Fetch PNFD and NSD artifact operations
		Rapporteur's action to use NFV-MANO consistently (one remaining case)	
March 2020	V3.3.7	NFVIFA(19)0001020	IFA013ed341 add 2 small notes for consistency
		NFVIFA(20)000061	IFA013ed341 FEAT10 Handling of WAN connectivity information
		NFVIFA(20)000064r1	IFA013ed341 locationConstraints for nested NS
		NFVIFA(20)000014r1	IFA013ed341 mirror of 12 adding missing extensions and vnfConfigrableProperties to ChangeVnfFlavour
		NFVIFA(20)000132	IFA013ed341 sync to IFA015 work according to 942r13 part4
		NFVIFA(20)000135r1	IFA013ed341 editorial changes
March 2020	V3.3.8	NFVIFA(20)000043r3	IFA013ed341 sync to IFA015 work according to 942r1 part1
		NFVIFA(20)000130r2	IFA013ed341 sync to IFA015 work according to 942r13 part2
		NFVIFA(20)000131r1	IFA013ed341 sync to IFA015 work according to 942r13 part3
		NFVIFA(20)000138	IFA013ed341 Labelling conditional mandatory
		NFVIFA(20)000165r1	IFA013 fix enum values part 1
		NFVIFA(20)000183	IFA013ed341_locationConstraints_for_nested_NS_additional_changes
		NFVIFA(20)000174r2	IFA013ed341 FEAT15 alignment with stage 3
		NFVIFA(20)000133r4	IFA013ed341 sync to IFA015 work according to 942r13 part5
		NFVIFA(20)000166r3	IFA013 fix enum values part 2
		NFVIFA(20)000195r1	IFA013 fix enum values part 3
		NFVIFA(20)000196r1	IFA013 fix enum values part 4
April 2020	V3.3.9	NFVIFA(20)000231r1	IFA013ed341 FEAT15 Moving VNF snapshot package API
April 2020	V3.3.10	NFVIFA(20)000275	IFA013 alignment efforts fix typos part 1
		NFVIFA(20)000277	IFA013 alignment efforts fix typos part 2
		NFVIFA(20)000278r1	IFA013ed341 align location constraints to SOL169r5 (some editorial corrections in 8.3.4.47)
		NFVIFA(20)000293	IFA013ed341 Fix notes in Clause 8.3.4.16
		NFVIFA(20)000294r1	IFA013ed341 FEAT15 Alignment of VnfcSnapshotInfo
April 2020	V3.3.11	NFVIFA(20)000315	IFA013ed341 Relaxing NFVI capacity notification subscriptions
June 2020	V4.0.1	Initial version for Release 4	
September 2020	V4.0.2	NFVIFA(20)000475	IFA013ed4111 Mirror of 382 Fix Type mismatch VnfDeploymetnFlavour
		NFVIFA(20)000476	IFA013ed411 Mirror of 454 FEAT15 Addressing gap additionalParams VNF snapshots
		NFVIFA(20)000505	IFA013ed411 mirror of 494 Adding Trunk Logical Topology between VNFC CPs
		NFVIFA(20)000528r1	IFA013ed411 mirror of 379r3 Modifications to NsLcmOperationOccurrenceNotification
		NFVIFA(20)000531	IFA013ed411 Mirror of 333r3 Changes in Terminate NS operation
		NFVIFA(20)000535	IFA013edr411 Mirror of NFVIFA(20)000534 fix usage of NFV003
October 2020	V4.0.3	NFVIFA(20)000675	IFA013ed411 Rel4 mirror of 666 VipCp related changes in linkport referencing
		NFVIFA(20)000699	IFA013ed421 Rel-4 mirror of 608, VIP and external connectivity related updates
December 2020	V4.0.4	NFVIFA(20)000662r2	IFA013ed421 MegaCR ENH02.02 NS feasibility check
		NFVIFA(20)000778	IFA013ed421 (forward mirror of 775) Aligning with SOL317 fixing notifying information about extLinkPort
		NFVIFA(20)000823	IFA013ed421 Rel-4 Mirror of 802, VnfExtCpInfo update
March 2021	V4.0.5	NFVIFA(20)000819r2	IFA013ed421 mirror of 818 Adding vnfProfileId to InstantiateVnfData
		NFVIFA(20)000924	IFA013ed411 Rel4 mirror of 862 VipCp related small fix
		NFVIFA(20)000845r3	ENH02.05 IFA013ed421 Introduction of nsScaleInfo complementing nsInstantionLevelId (overlap with 819r2 resolved by rapporteur)
March 2021	V4.0.6	NFVIFA(21)000126r1	IFA013ed421 mirror of 118 Cross stages alignment w.r.t. LCM coordination
		NFVIFA(21)000154r1	IFA013ed421 Rel4 mirror of 152 Cross stages alignment coordination delay
		NFVIFA(21)000188	IFA013ed421 Rel4 mirror of 144 adding additional attributes in Terminate NS and Update NS
		NFVIFA(21)000219r1	IFA013ed421 Mirror of 218 Avoid Reference to MAN001
		NFVIFA(20)000838r3	Enh02.04-IFA013ed421 Adding descriptor Ids to NS LCM operations
March 2021	V4.0.7	NFVIFA(21)000204	IFA013ed421 MegaCR FEAT17 Cloud-native VNFs

History

Document history		
V4.2.1	May 2021	Publication