



## **Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Or-Vnfm 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-IFA007ed341

---

**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 - NAF 742 C  
Association à but non lucratif enregistrée à la  
Sous-Préfecture de Grasse (06) N° 7803/88

---

**Important notice**

The present document can be downloaded from:

<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at [www.etsi.org/deliver](http://www.etsi.org/deliver).

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:

<https://portal.etsi.org/People/CommitteeSupportStaff.aspx>

---

**Copyright Notification**

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2020.

All rights reserved.

**DECT™**, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members.

**3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

**oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners.

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

# Contents

|   |    |
|---|----|
| Intellectual Property Rights .....  | 17 |
| Foreword.....   | 17 |
| Modal verbs terminology.....  | 17 |
| 1 Scope .....   | 18 |
| 2 References .....  | 18 |
| 2.1 Normative references .....  | 18 |
| 2.2 Informative references.....   | 18 |
| 3 Definition of terms, symbols and abbreviations.....   | 19 |
| 3.1 Terms.....  | 19 |
| 3.2 Symbols.....  | 19 |
| 3.3 Abbreviations .....   | 19 |
| 4 Overview of interfaces and information elements associated to the Or-Vnfm reference point ..... | 20 |
| 4.1 Introduction .....  | 20 |
| 4.2 Relation to other NFV Group Specifications.....   | 20 |
| 4.3 Conventions.....  | 21 |
| 5 Reference point and interface requirements .....  | 21 |
| 5.1 Introduction .....  | 21 |
| 5.2 Or-Vnfm reference point requirements .....  | 22 |
| 5.3 Interface requirements .....  | 22 |
| 5.3.1 VNF Package Management interface requirements.....  | 22 |
| 5.3.2 VNF Lifecycle Operation Granting interface requirements.....                                | 23 |
| 5.3.3 Virtualised Resources Management interfaces requirements .....                              | 23 |
| 5.3.3.1 Virtualised Resources Information Management interfaces requirements.....                 | 23 |
| 5.3.3.2 Virtualised Resources Management interfaces requirements.....                             | 24 |
| 5.3.3.3 Virtualised Resources Reservation Management interface requirements .....                 | 24 |
| 5.3.3.4 Virtualised Resources Reservation Change Notification interface requirements.....         | 24 |
| 5.3.3.5 Virtualised Resources Change Notification interfaces requirements .....                   | 25 |
| 5.3.3.6 Virtualised Resources Performance Management interface requirements .....                 | 25 |
| 5.3.3.7 Virtualised Resources Fault Management interface requirements .....                       | 26 |
| 5.3.3.8 Virtualised Resources Quota Management interfaces requirements .....                      | 26 |
| 5.3.3.9 Virtualised Resources Quota Change Notification interface requirements .....              | 26 |
| 5.3.3.10 Virtualised Resources Quota Available Notification interface requirements.....           | 27 |
| 5.3.4 VNF Lifecycle Management interface requirements .....                                       | 27 |
| 5.3.5 Void .....  | 28 |
| 5.3.6 VNF Performance Management interface requirements.....                                      | 29 |
| 5.3.7 VNF Fault Management interface requirements.....  | 29 |
| 5.3.8 Void .....  | 30 |
| 5.3.9 VNF Indicator interface requirements .....  | 30 |
| 5.3.10 Policy Management interface requirements.....  | 30 |
| 5.3.11 VNF Snapshot Package Management interface requirements .....                               | 31 |
| 6 NFVO exposed interfaces .....   | 31 |
| 6.1 Introduction .....  | 31 |
| 6.2 VNF Package Management interface .....  | 32 |
| 6.2.1 Description.....  | 32 |
| 6.2.2 Query VNF Package Info operation .....  | 32 |
| 6.2.2.1 Description .....   | 32 |
| 6.2.2.2 Input parameters.....   | 32 |
| 6.2.2.3 Output parameters .....   | 32 |
| 6.2.2.4 Operation results .....   | 33 |
| 6.2.3 Subscribe operation.....  | 33 |
| 6.2.3.1 Description .....   | 33 |
| 6.2.3.2 Input parameters.....   | 33 |
| 6.2.3.3 Output parameters .....   | 33 |

|         |   |    |
|---------|---|----|
| 6.2.3.4 | Operation results .....   | 34 |
| 6.2.4   | Notify operation .....  | 34 |
| 6.2.4.1 | Description .....   | 34 |
| 6.2.5   | Void .....  | 34 |
| 6.2.6   | Fetch VNF Package operation .....                                     | 34 |
| 6.2.6.1 | Description .....   | 34 |
| 6.2.6.2 | Input parameters .....  | 34 |
| 6.2.6.3 | Output parameters .....   | 35 |
| 6.2.6.4 | Operation results .....   | 35 |
| 6.2.7   | Fetch VNF Package Artifacts operation .....                           | 35 |
| 6.2.7.1 | Description .....   | 35 |
| 6.2.7.2 | Input parameters .....  | 35 |
| 6.2.7.3 | Output parameters .....   | 36 |
| 6.2.7.4 | Operation results .....   | 36 |
| 6.2.8   | Terminate Subscription operation .....                                | 36 |
| 6.2.8.1 | Description .....   | 36 |
| 6.2.8.2 | Input parameters .....  | 36 |
| 6.2.8.3 | Output parameters .....   | 36 |
| 6.2.8.4 | Operation results .....   | 36 |
| 6.2.9   | Query Subscription Info operation .....                               | 37 |
| 6.2.9.1 | Description .....   | 37 |
| 6.2.9.2 | Input parameters .....  | 37 |
| 6.2.9.3 | Output parameters .....   | 37 |
| 6.2.9.4 | Operation results .....   | 37 |
| 6.3     | VNF Lifecycle Operation Granting interface .....                      | 37 |
| 6.3.1   | Description .....   | 37 |
| 6.3.2   | Grant VNF Lifecycle Operation operation .....                         | 37 |
| 6.3.2.1 | Description .....   | 37 |
| 6.3.2.2 | Input parameters .....  | 39 |
| 6.3.2.3 | Output parameters .....   | 41 |
| 6.3.2.4 | Operation results .....   | 42 |
| 6.4     | Virtualised Resources Management interfaces in indirect mode .....    | 42 |
| 6.4.1   | Introduction .....  | 42 |
| 6.4.2   | Virtualised Compute interfaces .....                                  | 43 |
| 6.4.2.1 | Virtualised Compute Resources Management interface .....              | 43 |
| 6.4.2.2 | Virtualised Compute Resources Change Notification interface .....     | 43 |
| 6.4.2.3 | Virtualised Compute Resources Information Management interface .....  | 43 |
| 6.4.3   | Virtualised Network interfaces .....                                  | 43 |
| 6.4.3.1 | Virtualised Network Resources Management interface .....              | 43 |
| 6.4.3.2 | Virtualised Network Resources Change Notification interface .....     | 44 |
| 6.4.3.3 | Virtualised Network Resources Information Management interface .....  | 44 |
| 6.4.4   | Virtualised Storage interfaces .....                                  | 44 |
| 6.4.4.1 | Virtualised Storage Resources Management interface .....              | 44 |
| 6.4.4.2 | Virtualised Storage Resources Change Notification interface .....     | 45 |
| 6.4.4.3 | Virtualised Storage Resources Information Management interface .....  | 45 |
| 6.4.5   | Virtualised Resource Performance Management interface .....           | 45 |
| 6.4.6   | Virtualised Resource Fault Management interface .....                 | 46 |
| 6.4.7   | Virtualised Resources Quota Management interfaces .....               | 46 |
| 6.4.7.1 | Virtualised Compute Resources Quota Management interface .....        | 46 |
| 6.4.7.2 | Virtualised Network Resources Quota Management interface .....        | 46 |
| 6.4.7.3 | Virtualised Storage Resources Quota Management interface .....        | 46 |
| 6.4.7.4 | Virtualised Resources Quota Change Notification interface .....       | 46 |
| 6.4.8   | Virtualised Resource Reservation interfaces .....                     | 47 |
| 6.4.8.1 | Virtualised Compute Resources Reservation Management interface .....  | 47 |
| 6.4.8.2 | Virtualised Network Resources Reservation Management interface .....  | 47 |
| 6.4.8.3 | Virtualised Storage Resources Reservation Management interface .....  | 47 |
| 6.4.8.4 | Virtualised Resources Reservation Change Notification interface ..... | 47 |
| 6.5     | Virtualised Resources Quota Available Notification interface .....    | 47 |
| 6.5.1   | Description .....   | 47 |
| 6.5.2   | Subscribe operation .....   | 48 |
| 6.5.2.1 | Description .....   | 48 |
| 6.5.2.2 | Input parameters .....  | 48 |

|         |  |    |
|---------|--|----|
| 6.5.2.3 | Output parameters .....                                | 48 |
| 6.5.2.4 | Operation results .....                                | 48 |
| 6.5.3   | Notify operation .....                                 | 48 |
| 6.5.3.1 | Description .....                                      | 48 |
| 6.5.4   | Terminate Subscription operation .....                 | 49 |
| 6.5.4.1 | Description .....                                      | 49 |
| 6.5.4.2 | Input parameters .....                                 | 49 |
| 6.5.4.3 | Output parameters .....                                | 49 |
| 6.5.4.4 | Operation results .....                                | 49 |
| 6.5.5   | Query Subscription Info operation .....                | 49 |
| 6.5.5.1 | Description .....                                      | 49 |
| 6.5.5.2 | Input parameters .....                                 | 49 |
| 6.5.5.3 | Output parameters .....                                | 50 |
| 6.5.5.4 | Operation results .....                                | 50 |
| 6.6     | VNF Snapshot Package Management interface .....        | 50 |
| 6.6.1   | Description .....                                      | 50 |
| 6.6.2   | Fetch VNF Snapshot Package operation .....             | 50 |
| 6.6.2.1 | Description .....                                      | 50 |
| 6.6.2.2 | Input parameters .....                                 | 50 |
| 6.6.2.3 | Output parameters .....                                | 51 |
| 6.6.2.4 | Operation results .....                                | 51 |
| 6.6.3   | Fetch VNF Snapshot Package Artifacts operation .....   | 51 |
| 6.6.3.1 | Description .....                                      | 51 |
| 6.6.3.2 | Input parameters .....                                 | 51 |
| 6.6.3.3 | Output parameters .....                                | 51 |
| 6.6.3.4 | Operation results .....                                | 52 |
| 6.6.4   | Query VNF Snapshot Package Information operation ..... | 52 |
| 6.6.4.1 | Description .....                                      | 52 |
| 6.6.4.2 | Input parameters .....                                 | 52 |
| 6.6.4.3 | Output parameters .....                                | 52 |
| 6.6.4.4 | Operation results .....                                | 53 |
| 7       | VNFM exposed interfaces .....                          | 53 |
| 7.1     | Introduction .....                                     | 53 |
| 7.2     | VNF Lifecycle Management interface .....               | 53 |
| 7.2.1   | Description .....                                      | 53 |
| 7.2.2   | Create VNF Identifier operation .....                  | 54 |
| 7.2.2.1 | Description .....                                      | 54 |
| 7.2.2.2 | Input parameters .....                                 | 54 |
| 7.2.2.3 | Output parameters .....                                | 54 |
| 7.2.2.4 | Operation results .....                                | 55 |
| 7.2.3   | Instantiate VNF operation .....                        | 55 |
| 7.2.3.1 | Description .....                                      | 55 |
| 7.2.3.2 | Input parameters .....                                 | 55 |
| 7.2.3.3 | Output parameters .....                                | 56 |
| 7.2.3.4 | Operation results .....                                | 56 |
| 7.2.4   | Scale VNF operation .....                              | 57 |
| 7.2.4.1 | Description .....                                      | 57 |
| 7.2.4.2 | Input parameters .....                                 | 58 |
| 7.2.4.3 | Output parameters .....                                | 59 |
| 7.2.4.4 | Operation results .....                                | 59 |
| 7.2.5   | Scale VNF to Level operation .....                     | 59 |
| 7.2.5.1 | Description .....                                      | 59 |
| 7.2.5.2 | Input parameters .....                                 | 59 |
| 7.2.5.3 | Output parameters .....                                | 60 |
| 7.2.5.4 | Operation results .....                                | 60 |
| 7.2.6   | Change VNF Flavour operation .....                     | 60 |
| 7.2.6.1 | Description .....                                      | 60 |
| 7.2.6.2 | Input parameters .....                                 | 61 |
| 7.2.6.3 | Output parameters .....                                | 62 |
| 7.2.6.4 | Operation results .....                                | 62 |
| 7.2.7   | Terminate VNF operation .....                          | 62 |

|          |  |    |
|----------|--|----|
| 7.2.7.1  | Description .....                                | 62 |
| 7.2.7.2  | Input parameters .....                           | 62 |
| 7.2.7.3  | Output parameters .....                          | 63 |
| 7.2.7.4  | Operation results .....                          | 64 |
| 7.2.8    | Delete VNF Identifier operation .....            | 64 |
| 7.2.8.1  | Description .....                                | 64 |
| 7.2.8.2  | Input parameters .....                           | 64 |
| 7.2.8.3  | Output parameters .....                          | 64 |
| 7.2.8.4  | Operation results .....                          | 64 |
| 7.2.9    | Query VNF operation .....                        | 64 |
| 7.2.9.1  | Description .....                                | 64 |
| 7.2.9.2  | Input parameters .....                           | 65 |
| 7.2.9.3  | Output parameters .....                          | 65 |
| 7.2.9.4  | Operation results .....                          | 65 |
| 7.2.10   | Heal VNF operation .....                         | 65 |
| 7.2.10.1 | Description .....                                | 65 |
| 7.2.10.2 | Input parameters .....                           | 66 |
| 7.2.10.3 | Output parameters .....                          | 66 |
| 7.2.10.4 | Operation results .....                          | 66 |
| 7.2.11   | Operate VNF operation .....                      | 66 |
| 7.2.11.1 | Description .....                                | 66 |
| 7.2.11.2 | Input parameters .....                           | 67 |
| 7.2.11.3 | Output parameters .....                          | 68 |
| 7.2.11.4 | Operation results .....                          | 69 |
| 7.2.12   | Modify VNF Information operation .....           | 69 |
| 7.2.12.1 | Description .....                                | 69 |
| 7.2.12.2 | Input parameters .....                           | 69 |
| 7.2.12.3 | Output parameters .....                          | 69 |
| 7.2.12.4 | Operation results .....                          | 70 |
| 7.2.13   | Get Operation Status operation .....             | 70 |
| 7.2.13.1 | Description .....                                | 70 |
| 7.2.13.2 | Input parameters .....                           | 70 |
| 7.2.13.3 | Output parameters .....                          | 70 |
| 7.2.13.4 | Operation results .....                          | 71 |
| 7.2.14   | Subscribe operation .....                        | 71 |
| 7.2.14.1 | Description .....                                | 71 |
| 7.2.14.2 | Input parameters .....                           | 71 |
| 7.2.14.3 | Output parameters .....                          | 71 |
| 7.2.14.4 | Operation results .....                          | 72 |
| 7.2.15   | Notify operation .....                           | 72 |
| 7.2.15.1 | Description .....                                | 72 |
| 7.2.16   | Terminate Subscription operation .....           | 72 |
| 7.2.16.1 | Description .....                                | 72 |
| 7.2.16.2 | Input parameters .....                           | 72 |
| 7.2.16.3 | Output parameters .....                          | 73 |
| 7.2.16.4 | Operation results .....                          | 73 |
| 7.2.17   | Query Subscription Info operation .....          | 73 |
| 7.2.17.1 | Description .....                                | 73 |
| 7.2.17.2 | Input parameters .....                           | 73 |
| 7.2.17.3 | Output parameters .....                          | 73 |
| 7.2.17.4 | Operation results .....                          | 73 |
| 7.2.18   | Change External VNF Connectivity operation ..... | 74 |
| 7.2.18.1 | Description .....                                | 74 |
| 7.2.18.2 | Input parameters .....                           | 74 |
| 7.2.18.3 | Output parameters .....                          | 75 |
| 7.2.18.4 | Operation results .....                          | 75 |
| 7.2.19   | Query Snapshot Information operation .....       | 75 |
| 7.2.19.1 | Description .....                                | 75 |
| 7.2.19.2 | Input parameters .....                           | 76 |
| 7.2.19.3 | Output parameters .....                          | 76 |
| 7.2.19.4 | Operation results .....                          | 76 |
| 7.2.20   | Create Snapshot operation .....                  | 76 |

|          |   |    |
|----------|---|----|
| 7.2.20.1 | Description .....                           | 76 |
| 7.2.20.2 | Input parameters.....                       | 77 |
| 7.2.20.3 | Output parameters .....                     | 77 |
| 7.2.20.4 | Operation results .....                     | 77 |
| 7.2.21   | Revert-to Snapshot operation.....           | 77 |
| 7.2.21.1 | Description .....                           | 77 |
| 7.2.21.2 | Input parameters.....                       | 78 |
| 7.2.21.3 | Output parameters .....                     | 78 |
| 7.2.21.4 | Operation results .....                     | 78 |
| 7.2.22   | Delete Snapshot Information operation ..... | 78 |
| 7.2.22.1 | Description .....                           | 78 |
| 7.2.22.2 | Input parameters.....                       | 78 |
| 7.2.22.3 | Output parameters .....                     | 79 |
| 7.2.22.4 | Operation results .....                     | 79 |
| 7.2.23   | Change current VNF package operation.....   | 79 |
| 7.2.23.1 | Description .....                           | 79 |
| 7.2.23.2 | Input parameters.....                       | 80 |
| 7.2.23.3 | Output parameters .....                     | 81 |
| 7.2.23.4 | Operation results .....                     | 81 |
| 7.2.24   | Fetch VNF state snapshot .....              | 81 |
| 7.2.24.1 | Description .....                           | 81 |
| 7.2.24.2 | Input parameters.....                       | 82 |
| 7.2.24.3 | Output parameters .....                     | 82 |
| 7.2.24.4 | Operation results .....                     | 82 |
| 7.3      | Void.....                                   | 82 |
| 7.4      | VNF Performance Management interface .....  | 82 |
| 7.4.1    | Description.....                            | 82 |
| 7.4.2    | Create PM Job operation.....                | 83 |
| 7.4.2.1  | Description .....                           | 83 |
| 7.4.2.2  | Input parameters.....                       | 83 |
| 7.4.2.3  | Output parameters .....                     | 84 |
| 7.4.2.4  | Operation results .....                     | 84 |
| 7.4.3    | Delete PM Jobs operation .....              | 84 |
| 7.4.3.1  | Description .....                           | 84 |
| 7.4.3.2  | Input parameters.....                       | 84 |
| 7.4.3.3  | Output parameters .....                     | 85 |
| 7.4.3.4  | Operation results .....                     | 85 |
| 7.4.4    | Subscribe operation.....                    | 85 |
| 7.4.4.1  | Description .....                           | 85 |
| 7.4.4.2  | Input parameters.....                       | 85 |
| 7.4.4.3  | Output parameters .....                     | 85 |
| 7.4.4.4  | Operation results .....                     | 86 |
| 7.4.5    | Notify operation .....                      | 86 |
| 7.4.5.1  | Description .....                           | 86 |
| 7.4.6    | Query PM Job operation .....                | 86 |
| 7.4.6.1  | Description .....                           | 86 |
| 7.4.6.2  | Input parameters.....                       | 86 |
| 7.4.6.3  | Output parameters .....                     | 87 |
| 7.4.6.4  | Operation results .....                     | 87 |
| 7.4.7    | Create Threshold operation.....             | 87 |
| 7.4.7.1  | Description .....                           | 87 |
| 7.4.7.2  | Input parameters.....                       | 87 |
| 7.4.7.3  | Output parameters .....                     | 87 |
| 7.4.7.4  | Operation results .....                     | 88 |
| 7.4.8    | Delete Thresholds operation .....           | 88 |
| 7.4.8.1  | Description .....                           | 88 |
| 7.4.8.2  | Input parameters.....                       | 88 |
| 7.4.8.3  | Output parameters .....                     | 88 |
| 7.4.8.4  | Operation results .....                     | 88 |
| 7.4.9    | Query Threshold operation .....             | 89 |
| 7.4.9.1  | Description .....                           | 89 |
| 7.4.9.2  | Input parameters.....                       | 89 |

|          |  |    |
|----------|--|----|
| 7.4.9.3  | Output parameters .....                | 89 |
| 7.4.9.4  | Operation results .....                | 89 |
| 7.4.10   | Terminate Subscription operation.....  | 89 |
| 7.4.10.1 | Description .....                      | 89 |
| 7.4.10.2 | Input parameters.....                  | 90 |
| 7.4.10.3 | Output parameters .....                | 90 |
| 7.4.10.4 | Operation results .....                | 90 |
| 7.4.11   | Query Subscription Info operation..... | 90 |
| 7.4.11.1 | Description .....                      | 90 |
| 7.4.11.2 | Input parameters.....                  | 90 |
| 7.4.11.3 | Output parameters .....                | 90 |
| 7.4.11.4 | Operation results .....                | 91 |
| 7.5      | VNF Fault Management interface .....   | 91 |
| 7.5.1    | Description.....                       | 91 |
| 7.5.2    | Subscribe operation.....               | 91 |
| 7.5.2.1  | Description .....                      | 91 |
| 7.5.2.2  | Input parameters.....                  | 91 |
| 7.5.2.3  | Output parameters .....                | 92 |
| 7.5.2.4  | Operation results .....                | 92 |
| 7.5.3    | Notify operation.....                  | 92 |
| 7.5.3.1  | Description .....                      | 92 |
| 7.5.4    | Get Alarm List operation.....          | 92 |
| 7.5.4.1  | Description .....                      | 92 |
| 7.5.4.2  | Input parameters.....                  | 93 |
| 7.5.4.3  | Output parameters .....                | 93 |
| 7.5.4.4  | Operation results .....                | 93 |
| 7.5.5    | Terminate Subscription operation.....  | 93 |
| 7.5.5.1  | Description .....                      | 93 |
| 7.5.5.2  | Input parameters.....                  | 93 |
| 7.5.5.3  | Output parameters .....                | 94 |
| 7.5.5.4  | Operation results .....                | 94 |
| 7.5.6    | Query Subscription Info operation..... | 94 |
| 7.5.6.1  | Description .....                      | 94 |
| 7.5.6.2  | Input parameters.....                  | 94 |
| 7.5.6.3  | Output parameters .....                | 94 |
| 7.5.6.4  | Operation results .....                | 94 |
| 7.5.7    | Acknowledge alarms operation.....      | 94 |
| 7.5.7.1  | Description .....                      | 94 |
| 7.5.7.2  | Input parameters.....                  | 95 |
| 7.5.7.3  | Output parameters .....                | 95 |
| 7.5.7.4  | Operation results .....                | 95 |
| 7.6      | Void.....                              | 95 |
| 7.7      | VNF Indicator interface.....           | 95 |
| 7.7.1    | Description.....                       | 95 |
| 7.7.2    | Subscribe operation.....               | 96 |
| 7.7.2.1  | Description .....                      | 96 |
| 7.7.2.2  | Input parameters.....                  | 96 |
| 7.7.2.3  | Output parameters .....                | 96 |
| 7.7.2.4  | Operation results .....                | 96 |
| 7.7.3    | Notify operation.....                  | 96 |
| 7.7.3.1  | Description .....                      | 96 |
| 7.7.4    | Get Indicator Value operation.....     | 97 |
| 7.7.4.1  | Description .....                      | 97 |
| 7.7.4.2  | Input parameters.....                  | 97 |
| 7.7.4.3  | Output parameters .....                | 97 |
| 7.7.4.4  | Operation results .....                | 97 |
| 7.7.5    | Terminate Subscription operation.....  | 97 |
| 7.7.5.1  | Description .....                      | 97 |
| 7.7.5.2  | Input parameters.....                  | 98 |
| 7.7.5.3  | Output parameters .....                | 98 |
| 7.7.5.4  | Operation results .....                | 98 |
| 7.7.6    | Query Subscription Info operation..... | 98 |



|          |   |     |
|----------|---|-----|
| 7.7.6.1  | Description .....                       | 98  |
| 7.7.6.2  | Input parameters .....                  | 98  |
| 7.7.6.3  | Output parameters .....                 | 98  |
| 7.7.6.4  | Operation results .....                 | 99  |
| 7.8      | Policy Management interface .....       | 99  |
| 7.8.1    | Description .....                       | 99  |
| 7.8.2    | Transfer Policy operation .....         | 99  |
| 7.8.2.1  | Description .....                       | 99  |
| 7.8.2.2  | Input parameters .....                  | 99  |
| 7.8.2.3  | Output parameters .....                 | 100 |
| 7.8.2.4  | Operation results .....                 | 100 |
| 7.8.3    | Delete Policy operation .....           | 100 |
| 7.8.3.1  | Description .....                       | 100 |
| 7.8.3.2  | Input parameters .....                  | 100 |
| 7.8.3.3  | Output parameters .....                 | 101 |
| 7.8.3.4  | Operation results .....                 | 101 |
| 7.8.4    | Query Policy operation .....            | 101 |
| 7.8.4.1  | Description .....                       | 101 |
| 7.8.4.2  | Input parameters .....                  | 101 |
| 7.8.4.3  | Output parameters .....                 | 102 |
| 7.8.4.4  | Operation results .....                 | 102 |
| 7.8.5    | Activate Policy operation .....         | 102 |
| 7.8.5.1  | Description .....                       | 102 |
| 7.8.5.2  | Input parameters .....                  | 102 |
| 7.8.5.3  | Output parameters .....                 | 102 |
| 7.8.5.4  | Operation results .....                 | 103 |
| 7.8.6    | Deactivate Policy operation .....       | 103 |
| 7.8.6.1  | Description .....                       | 103 |
| 7.8.6.2  | Input parameters .....                  | 103 |
| 7.8.6.3  | Output parameters .....                 | 103 |
| 7.8.6.4  | Operation results .....                 | 103 |
| 7.8.7    | Subscribe operation .....               | 103 |
| 7.8.7.1  | Description .....                       | 103 |
| 7.8.7.2  | Input parameters .....                  | 104 |
| 7.8.7.3  | Output parameters .....                 | 104 |
| 7.8.7.4  | Operation results .....                 | 104 |
| 7.8.8    | Notify operation .....                  | 104 |
| 7.8.8.1  | Description .....                       | 104 |
| 7.8.9    | Terminate Subscription operation .....  | 105 |
| 7.8.9.1  | Description .....                       | 105 |
| 7.8.9.2  | Input parameters .....                  | 105 |
| 7.8.9.3  | Output parameters .....                 | 105 |
| 7.8.9.4  | Operation results .....                 | 105 |
| 7.8.10   | Query Subscription Info operation ..... | 105 |
| 7.8.10.1 | Description .....                       | 105 |
| 7.8.10.2 | Input parameters .....                  | 105 |
| 7.8.10.3 | Output parameters .....                 | 106 |
| 7.8.10.4 | Operation results .....                 | 106 |
| 7.8.11   | Associate Policy operation .....        | 106 |
| 7.8.11.1 | Description .....                       | 106 |
| 7.8.11.2 | Input parameters .....                  | 106 |
| 7.8.11.3 | Output parameters .....                 | 106 |
| 7.8.11.4 | Operation results .....                 | 107 |
| 7.8.12   | Disassociate Policy operation .....     | 107 |
| 7.8.12.1 | Description .....                       | 107 |
| 7.8.12.2 | Input parameters .....                  | 107 |
| 7.8.12.3 | Output parameters .....                 | 107 |
| 7.8.12.4 | Operation results .....                 | 107 |
| 7.9      | Void .....                              | 107 |
| 8        | Information elements exchanged .....    | 108 |
| 8.1      | Introduction .....                      | 108 |

|          |  |     |
|----------|--|-----|
| 8.2      | Information elements and notifications related to VNF Package Management ..... | 108 |
| 8.2.1    | Introduction.....  | 108 |
| 8.2.2    | VnfPkgInfo information element .....   | 108 |
| 8.2.2.1  | Description .....  | 108 |
| 8.2.2.2  | Attributes.....  | 108 |
| 8.2.3    | Vnfd information element.....  | 109 |
| 8.2.3.1  | Description .....  | 109 |
| 8.2.3.2  | Attributes.....  | 109 |
| 8.2.4    | VnfPackageOnBoardingNotification .....   | 109 |
| 8.2.4.1  | Description .....  | 109 |
| 8.2.4.2  | Trigger Conditions .....   | 110 |
| 8.2.4.3  | Attributes.....  | 110 |
| 8.2.5    | VnfPackageChangeNotification .....   | 110 |
| 8.2.5.1  | Description .....  | 110 |
| 8.2.5.2  | Trigger Conditions .....   | 110 |
| 8.2.5.3  | Attributes.....  | 110 |
| 8.2.6    | Void .....   | 111 |
| 8.2.7    | VnfPackageSoftwareImageInfo information element.....                           | 111 |
| 8.2.7.1  | Description .....  | 111 |
| 8.2.7.2  | Attributes.....  | 111 |
| 8.2.8    | VnfPackageArtifactInformation information element .....                        | 111 |
| 8.2.8.1  | Description .....  | 111 |
| 8.2.8.2  | Attributes.....  | 111 |
| 8.2.9    | Void .....   | 111 |
| 8.3      | Information elements related to VNF Lifecycle Operation Granting.....          | 112 |
| 8.3.1    | Introduction.....  | 112 |
| 8.3.2    | ResourceDefinition information element.....                                    | 112 |
| 8.3.2.1  | Description .....  | 112 |
| 8.3.2.2  | Attributes.....  | 112 |
| 8.3.3    | GrantInfo information element .....  | 113 |
| 8.3.3.1  | Description .....  | 113 |
| 8.3.3.2  | Attributes.....  | 113 |
| 8.3.4    | ZoneInfo information element .....   | 114 |
| 8.3.4.1  | Description .....  | 114 |
| 8.3.4.2  | Attributes.....  | 114 |
| 8.3.5    | ZoneGroupInfo information element .....  | 115 |
| 8.3.5.1  | Description .....  | 115 |
| 8.3.5.2  | Attributes.....  | 115 |
| 8.3.6    | PlacementConstraint information element.....                                   | 115 |
| 8.3.6.1  | Description .....  | 115 |
| 8.3.6.2  | Attributes.....  | 115 |
| 8.3.7    | VimConstraint information element .....  | 116 |
| 8.3.7.1  | Description .....  | 116 |
| 8.3.7.2  | Attributes.....  | 116 |
| 8.3.8    | ConstraintResourceRef information element.....                                 | 116 |
| 8.3.8.1  | Description .....  | 116 |
| 8.3.8.2  | Attributes.....  | 117 |
| 8.3.9    | VimAssets information element .....  | 117 |
| 8.3.9.1  | Description .....  | 117 |
| 8.3.9.2  | Attributes.....  | 117 |
| 8.3.10   | VimComputeResourceFlavour information element .....                            | 118 |
| 8.3.10.1 | Description .....  | 118 |
| 8.3.10.2 | Attributes.....  | 118 |
| 8.3.11   | VimSoftwareImage information element.....                                      | 118 |
| 8.3.11.1 | Description .....  | 118 |
| 8.3.11.2 | Attributes.....  | 118 |
| 8.3.12   | VimSnapshotResource information element .....                                  | 119 |
| 8.3.12.1 | Description .....  | 119 |
| 8.3.12.2 | Attributes.....  | 119 |
| 8.3.13   | SnapshotResourceDefinition information element .....                           | 120 |
| 8.3.13.1 | Description .....  | 120 |
| 8.3.13.2 | Attributes.....  | 120 |

|           |  |     |
|-----------|--|-----|
| 8.4       | Information elements and notifications related to Virtualised Resources Management in indirect mode..... | 121 |
| 8.4.1     | Introduction.....  | 121 |
| 8.4.2     | Information elements related to Virtualised Compute .....  | 121 |
| 8.4.2.1   | Introduction.....  | 121 |
| 8.4.2.2   | ComputeResourceWithRpInfo information element.....   | 121 |
| 8.4.2.2.1 | Description .....  | 121 |
| 8.4.2.2.2 | Attributes .....   | 121 |
| 8.4.2.3   | ComputeResourceWithRpId information element .....  | 121 |
| 8.4.2.3.1 | Description .....  | 121 |
| 8.4.2.3.2 | Attributes .....   | 121 |
| 8.4.2.4   | VirtualComputeResourceWithRpInfo information element .....   | 122 |
| 8.4.2.4.1 | Description .....  | 122 |
| 8.4.2.4.2 | Attributes .....   | 122 |
| 8.4.3     | Information elements related to Virtualised Network.....   | 122 |
| 8.4.3.1   | Introduction.....  | 122 |
| 8.4.3.2   | NetworkResourceWithRpInfo information element .....  | 122 |
| 8.4.3.2.1 | Description .....  | 122 |
| 8.4.3.2.2 | Attributes .....   | 122 |
| 8.4.3.3   | NetworkResourceWithRpId information element.....   | 122 |
| 8.4.3.3.1 | Description .....  | 122 |
| 8.4.3.3.2 | Attributes .....   | 123 |
| 8.4.3.4   | VirtualNetworkResourceWithRpInfo information element .....   | 123 |
| 8.4.3.4.1 | Description .....  | 123 |
| 8.4.3.4.2 | Attributes .....   | 123 |
| 8.4.4     | Information elements related to Virtualised Storage .....  | 123 |
| 8.4.4.1   | Introduction.....  | 123 |
| 8.4.4.2   | StorageResourceWithRpInfo information element .....  | 123 |
| 8.4.4.2.1 | Description .....  | 123 |
| 8.4.4.2.2 | Attributes .....   | 123 |
| 8.4.4.3   | StorageResourceWithRpId information element .....  | 124 |
| 8.4.4.3.1 | Description .....  | 124 |
| 8.4.4.3.2 | Attributes .....   | 124 |
| 8.4.4.4   | VirtualStorageResourceWithRpInfo information element .....   | 124 |
| 8.4.4.4.1 | Description .....  | 124 |
| 8.4.4.4.2 | Attributes .....   | 124 |
| 8.4.5     | Notifications related to changes of virtualised resources.....   | 124 |
| 8.4.5.1   | Introduction.....  | 124 |
| 8.4.5.2   | VirtualisedResourceWithRpChangeNotification .....  | 125 |
| 8.4.5.2.1 | Description .....  | 125 |
| 8.4.5.2.2 | Trigger conditions .....   | 125 |
| 8.4.5.2.3 | Attributes .....   | 125 |
| 8.4.5.3   | InformationWithRpChangeNotification.....   | 125 |
| 8.4.5.3.1 | Description .....  | 125 |
| 8.4.5.3.2 | Trigger conditions .....   | 125 |
| 8.4.5.3.3 | Attributes .....   | 125 |
| 8.4.6     | Notifications related to Virtualised Resource Performance Management .....                               | 126 |
| 8.4.6.1   | Introduction.....  | 126 |
| 8.4.6.2   | PerformanceInformationWithRpAvailableNotification .....  | 126 |
| 8.4.6.2.1 | Description .....  | 126 |
| 8.4.6.2.2 | Trigger conditions .....   | 126 |
| 8.4.6.2.3 | Attributes .....   | 126 |
| 8.4.6.3   | ThresholdCrossedWithRpNotification.....  | 126 |
| 8.4.6.3.1 | Description .....  | 126 |
| 8.4.6.3.2 | Trigger conditions .....   | 126 |
| 8.4.6.3.3 | Attributes .....   | 126 |
| 8.4.7     | Information elements and notifications related to Virtualised Resource Fault Management .....            | 127 |
| 8.4.7.1   | Introduction.....  | 127 |
| 8.4.7.2   | AlarmWithRpInfo information element.....   | 127 |
| 8.4.7.2.1 | Description .....  | 127 |
| 8.4.7.2.2 | Attributes .....   | 127 |
| 8.4.7.3   | AlarmWithRpNotification.....   | 127 |
| 8.4.7.3.1 | Description .....  | 127 |

|           |  |     |
|-----------|--|-----|
| 8.4.7.3.2 | Trigger conditions .....   | 127 |
| 8.4.7.3.3 | Attributes .....   | 128 |
| 8.4.7.4   | AlarmClearedWithRpNotification .....   | 128 |
| 8.4.7.4.1 | Description .....  | 128 |
| 8.4.7.4.2 | Trigger conditions .....   | 128 |
| 8.4.7.4.3 | Attributes .....   | 128 |
| 8.4.8     | Information elements and notifications related to Virtualised Resources Quota .....      | 128 |
| 8.4.8.1   | Introduction .....   | 128 |
| 8.4.8.2   | VirtualComputeQuotaWithRpInfo information element.....                                   | 129 |
| 8.4.8.2.1 | Description .....  | 129 |
| 8.4.8.2.2 | Attributes .....   | 129 |
| 8.4.8.3   | VirtualNetworkQuotaWithRpInfo information element .....                                  | 129 |
| 8.4.8.3.1 | Description .....  | 129 |
| 8.4.8.3.2 | Attributes .....   | 129 |
| 8.4.8.4   | VirtualStorageQuotaWithRpInfo information element.....                                   | 129 |
| 8.4.8.4.1 | Description .....  | 129 |
| 8.4.8.4.2 | Attributes .....   | 129 |
| 8.4.8.5   | VirtualisedResourceQuotaWithRpChangeNotification.....                                    | 130 |
| 8.4.8.5.1 | Description .....  | 130 |
| 8.4.8.5.2 | Trigger conditions .....   | 130 |
| 8.4.8.5.3 | Attributes .....   | 130 |
| 8.4.9     | Information elements and notifications related to Virtualised Resources Reservation..... | 130 |
| 8.4.9.1   | Introduction.....  | 130 |
| 8.4.9.2   | ReservedVirtualComputeWithRpInfo information element .....                               | 130 |
| 8.4.9.2.1 | Description .....  | 130 |
| 8.4.9.2.2 | Attributes .....   | 131 |
| 8.4.9.3   | ReservedVirtualNetworkWithRpInfo information element .....                               | 131 |
| 8.4.9.3.1 | Description .....  | 131 |
| 8.4.9.3.2 | Attributes .....   | 131 |
| 8.4.9.4   | ReservedVirtualStorageWithRpInfo information element .....                               | 131 |
| 8.4.9.4.1 | Description .....  | 131 |
| 8.4.9.4.2 | Attributes .....   | 131 |
| 8.4.9.5   | VirtualisedResourceReservationWithRpChangeNotification .....                             | 132 |
| 8.4.9.5.1 | Description .....  | 132 |
| 8.4.9.5.2 | Trigger conditions .....   | 132 |
| 8.4.9.5.3 | Attributes .....   | 132 |
| 8.5       | Information elements related to VNF Lifecycle Management .....                           | 132 |
| 8.5.1     | Introduction.....  | 132 |
| 8.5.2     | VnfInfo information element .....  | 132 |
| 8.5.2.1   | Description .....  | 132 |
| 8.5.2.2   | Attributes.....  | 132 |
| 8.5.3     | InstantiatedVnfInfo information element.....   | 134 |
| 8.5.3.1   | Description .....  | 134 |
| 8.5.3.2   | Attributes.....  | 134 |
| 8.5.4     | VnfcResourceInfo information element.....  | 136 |
| 8.5.4.1   | Description .....  | 136 |
| 8.5.4.2   | Attributes.....  | 136 |
| 8.5.5     | VnfVirtualLinkResourceInfo information element.....                                      | 136 |
| 8.5.5.1   | Description .....  | 136 |
| 8.5.5.2   | Attributes.....  | 136 |
| 8.5.6     | VirtualStorageResourceInfo information element .....                                     | 137 |
| 8.5.6.1   | Description .....  | 137 |
| 8.5.6.2   | Attributes.....  | 137 |
| 8.5.7     | ResourceHandle information element.....  | 138 |
| 8.5.7.1   | Description .....  | 138 |
| 8.5.7.2   | Attributes.....  | 138 |
| 8.5.8     | ScaleInfo information element.....   | 138 |
| 8.5.8.1   | Description .....  | 138 |
| 8.5.8.2   | Attributes.....  | 138 |
| 8.5.9     | ExtVirtualLinkInfo information element .....   | 139 |
| 8.5.9.1   | Description .....  | 139 |
| 8.5.9.2   | Attributes.....  | 139 |

|          |  |     |
|----------|--|-----|
| 8.5.10   | ExtManagedVirtualLinkInfo information element .....                                | 139 |
| 8.5.10.1 | Description .....  | 139 |
| 8.5.10.2 | Attributes .....   | 139 |
| 8.5.11   | VnfLinkPortInfo information element .....  | 140 |
| 8.5.11.1 | Description .....  | 140 |
| 8.5.11.2 | Attributes .....   | 140 |
| 8.5.12   | VnfExtCpInfo information element .....   | 141 |
| 8.5.12.1 | Description .....  | 141 |
| 8.5.12.2 | Attributes .....   | 141 |
| 8.5.13   | ExtLinkPortInfo information element .....  | 141 |
| 8.5.13.1 | Description .....  | 141 |
| 8.5.13.2 | Attributes .....   | 141 |
| 8.5.14   | VnfcCpInfo information element .....   | 142 |
| 8.5.14.1 | Description .....  | 142 |
| 8.5.14.2 | Attributes .....   | 142 |
| 8.5.15   | CpProtocolInfo information element .....   | 142 |
| 8.5.15.1 | Description .....  | 142 |
| 8.5.15.2 | Attributes .....   | 142 |
| 8.5.16   | VnfSnapshotInfo information element .....  | 143 |
| 8.5.16.1 | Description .....  | 143 |
| 8.5.16.2 | Attributes .....   | 143 |
| 8.5.17   | VnfcSnapshotInfo information element .....   | 144 |
| 8.5.17.1 | Description .....  | 144 |
| 8.5.17.2 | Attributes .....   | 144 |
| 8.5.18   | StorageSnapshotResource information element .....                                  | 144 |
| 8.5.18.1 | Description .....  | 144 |
| 8.5.18.2 | Attributes .....   | 145 |
| 8.6      | Information elements and notifications related to VNF Lifecycle Changes .....      | 145 |
| 8.6.1    | Introduction .....   | 145 |
| 8.6.2    | VnfLcmOperationOccurrenceNotification .....  | 145 |
| 8.6.2.1  | Description .....  | 145 |
| 8.6.2.2  | Trigger conditions .....   | 145 |
| 8.6.2.3  | Attributes .....   | 146 |
| 8.6.3    | AffectedVnfc information element .....   | 147 |
| 8.6.3.1  | Description .....  | 147 |
| 8.6.3.2  | Attributes .....   | 147 |
| 8.6.4    | AffectedVirtualLink information element .....                                      | 148 |
| 8.6.4.1  | Description .....  | 148 |
| 8.6.4.2  | Attributes .....   | 148 |
| 8.6.5    | AffectedVirtualStorage information element .....                                   | 149 |
| 8.6.5.1  | Description .....  | 149 |
| 8.6.5.2  | Attributes .....   | 149 |
| 8.6.6    | Void .....   | 150 |
| 8.6.7    | VnfIdentifierCreationNotification .....  | 150 |
| 8.6.7.1  | Description .....  | 150 |
| 8.6.7.2  | Trigger conditions .....   | 150 |
| 8.6.7.3  | Attributes .....   | 150 |
| 8.6.8    | VnfIdentifierDeletionNotification .....  | 151 |
| 8.6.8.1  | Description .....  | 151 |
| 8.6.8.2  | Trigger conditions .....   | 151 |
| 8.6.8.3  | Attributes .....   | 151 |
| 8.7      | Information elements and notifications related to VNF Performance Management ..... | 151 |
| 8.7.1    | Introduction .....   | 151 |
| 8.7.2    | ObjectSelection information element .....  | 151 |
| 8.7.2.1  | Description .....  | 151 |
| 8.7.2.2  | Attributes .....   | 151 |
| 8.7.3    | PmJob information element .....  | 152 |
| 8.7.3.1  | Description .....  | 152 |
| 8.7.3.2  | Attributes .....   | 152 |
| 8.7.4    | Threshold information element .....  | 153 |
| 8.7.4.1  | Description .....  | 153 |
| 8.7.4.2  | Attributes .....   | 153 |

|          |   |     |
|----------|---|-----|
| 8.7.5    | PerformanceReport information element.....                                  | 153 |
| 8.7.5.1  | Description.....  | 153 |
| 8.7.5.2  | Attributes.....   | 154 |
| 8.7.6    | PerformanceReportEntry information element.....                             | 154 |
| 8.7.6.1  | Description.....  | 154 |
| 8.7.6.2  | Attributes.....   | 154 |
| 8.7.7    | PerformanceValueEntry information element.....                              | 154 |
| 8.7.7.1  | Description.....  | 154 |
| 8.7.7.2  | Attributes.....   | 154 |
| 8.7.8    | PerformanceInformationAvailableNotification.....                            | 155 |
| 8.7.8.1  | Description.....  | 155 |
| 8.7.8.2  | Trigger Conditions.....   | 155 |
| 8.7.8.3  | Attributes.....   | 155 |
| 8.7.9    | ThresholdCrossedNotification.....   | 155 |
| 8.7.9.1  | Description.....  | 155 |
| 8.7.9.2  | Trigger Condition.....  | 155 |
| 8.7.9.3  | Attributes.....   | 155 |
| 8.8      | Information elements and notifications related to VNF Fault Management..... | 156 |
| 8.8.1    | Introduction.....   | 156 |
| 8.8.2    | AlarmNotification.....  | 156 |
| 8.8.2.1  | Description.....  | 156 |
| 8.8.2.2  | Trigger conditions.....   | 156 |
| 8.8.2.3  | Attributes.....   | 156 |
| 8.8.3    | AlarmClearedNotification.....   | 157 |
| 8.8.3.1  | Description.....  | 157 |
| 8.8.3.2  | Trigger conditions.....   | 157 |
| 8.8.3.3  | Attributes.....   | 157 |
| 8.8.4    | Alarm information element.....  | 157 |
| 8.8.4.1  | Description.....  | 157 |
| 8.8.4.2  | Attributes.....   | 157 |
| 8.8.5    | FaultyResourceInfo information element.....                                 | 158 |
| 8.8.5.1  | Description.....  | 158 |
| 8.8.5.2  | Attributes.....   | 159 |
| 8.8.6    | AlarmListRebuiltNotification.....   | 159 |
| 8.8.6.1  | Description.....  | 159 |
| 8.8.6.2  | Trigger conditions.....   | 159 |
| 8.8.6.3  | Attributes.....   | 159 |
| 8.9      | Void.....   | 159 |
| 8.10     | Information elements and notifications related to VNF Indicators.....       | 159 |
| 8.10.1   | Introduction.....   | 159 |
| 8.10.2   | IndicatorValueChangeNotification.....                                       | 159 |
| 8.10.2.1 | Description.....  | 159 |
| 8.10.2.2 | Trigger conditions.....   | 159 |
| 8.10.2.3 | Attributes.....   | 160 |
| 8.10.3   | IndicatorInformation information element.....                               | 160 |
| 8.10.3.1 | Description.....  | 160 |
| 8.10.3.2 | Attributes.....   | 160 |
| 8.10.4   | SupportedIndicatorsChangeNotification.....                                  | 160 |
| 8.10.4.1 | Description.....  | 160 |
| 8.10.4.2 | Trigger conditions.....   | 160 |
| 8.10.4.3 | Attributes.....   | 160 |
| 8.10.5   | SupportedIndicatorInformation information element.....                      | 161 |
| 8.10.5.1 | Description.....  | 161 |
| 8.10.5.2 | Attributes.....   | 161 |
| 8.11     | Notifications related to Virtualised Resources Quota.....                   | 161 |
| 8.11.1   | Introduction.....   | 161 |
| 8.11.2   | VirtualisedResourceQuotaAvailableNotification.....                          | 161 |
| 8.11.2.1 | Description.....  | 161 |
| 8.11.2.2 | Trigger Conditions.....   | 161 |
| 8.11.2.3 | Attributes.....   | 161 |
| 8.12     | Information elements and notifications related to multiple interfaces.....  | 162 |
| 8.12.1   | Introduction.....   | 162 |

|                               |  |            |
|-------------------------------|--|------------|
| 8.12.2                        | ExtVirtualLinkData information element .....                                   | 162        |
| 8.12.2.1                      | Description .....  | 162        |
| 8.12.2.2                      | Attributes .....   | 162        |
| 8.12.2a                       | ExtLinkPortData information element.....                                       | 162        |
| 8.12.2a.1                     | Description .....  | 162        |
| 8.12.2a.2                     | Attributes.....  | 162        |
| 8.12.3                        | VnfExtCpData information element .....   | 163        |
| 8.12.3.1                      | Description .....  | 163        |
| 8.12.3.2                      | Attributes.....  | 163        |
| 8.12.3a                       | VnfExtCpConfig information element.....  | 163        |
| 8.12.3a.1                     | Description .....  | 163        |
| 8.12.3a.2                     | Attributes.....  | 163        |
| 8.12.4                        | ExtManagedVirtualLinkData information element .....                            | 164        |
| 8.12.4.1                      | Description .....  | 164        |
| 8.12.4.2                      | Attributes.....  | 164        |
| 8.12.5                        | VimConnectionInfo information element .....                                    | 165        |
| 8.12.5.1                      | Description .....  | 165        |
| 8.12.5.2                      | Attributes.....  | 165        |
| 8.12.6                        | VnfLinkPortData information element .....                                      | 166        |
| 8.12.6.1                      | Description .....  | 166        |
| 8.12.6.2                      | Attributes.....  | 166        |
| 8.13                          | Information elements and notifications related to Policy Management .....      | 166        |
| 8.13.1                        | Introduction.....  | 166        |
| 8.13.2                        | Information elements related to Policy Management Operations .....             | 166        |
| 8.13.2.1                      | Introduction.....  | 166        |
| 8.13.2.2                      | PolicyInfo information element .....   | 166        |
| 8.13.2.2.1                    | Description .....  | 166        |
| 8.13.2.2.2                    | Attributes .....   | 166        |
| 8.13.3                        | PolicyChangeNotification.....  | 166        |
| 8.13.3.1                      | Description .....  | 166        |
| 8.13.3.2                      | Trigger Conditions .....   | 167        |
| 8.13.3.3                      | Attributes.....  | 167        |
| 8.13.4                        | PolicyConflictNotification.....  | 167        |
| 8.13.4.1                      | Description .....  | 167        |
| 8.13.4.2                      | Trigger Conditions .....   | 167        |
| 8.13.4.3                      | Attributes.....  | 167        |
| 8.14                          | Information elements related to VNF Snapshot Package Management .....          | 167        |
| 8.14.1                        | Introduction.....  | 167        |
| 8.14.2                        | VnfSnapshotPkgInfo information element .....                                   | 167        |
| 8.14.2.1                      | Description .....  | 167        |
| 8.14.2.2                      | Attributes.....  | 167        |
| 8.14.3                        | SnapshotPkgArtifactInformation information element .....                       | 168        |
| 8.14.3.1                      | Description .....  | 168        |
| 8.14.3.2                      | Attributes.....  | 168        |
| 8.14.4                        | VnfSnapshotImageInfo information element.....                                  | 169        |
| 8.14.4.1                      | Description .....  | 169        |
| 8.14.4.2                      | Attributes.....  | 169        |
| 8.14.5                        | VnfStateSnapshotInfo information element.....                                  | 169        |
| 8.14.5.1                      | Description .....  | 169        |
| 8.14.5.2                      | Attributes.....  | 169        |
| <b>Annex A (informative):</b> | <b>    Examples of VNF connectivity patterns .....</b>                         | <b>171</b> |
| A.1                           | Introduction .....   | 171        |
| A.2                           | Example of a VNF with two different types of external connections points ..... | 171        |
| A.3                           | Example of changing VNF connectivity .....                                     | 172        |
| <b>Annex B (informative):</b> | <b>    VNF software modification .....</b>                                     | <b>173</b> |
| B.1                           | Introduction .....   | 173        |
| B.2                           | VNF software modification not assisted by NFV-MANO.....                        | 173        |

|                               |  |            |
|-------------------------------|--|------------|
| B.2.1                         | Description .....  | 173        |
| B.2.2                         | Procedure.....   | 173        |
| B.3                           | VNF software modification assisted by NFV-MANO via change of current VNF Package ..... | 175        |
| B.3.1                         | Overview .....   | 175        |
| B.3.2                         | Procedure.....   | 176        |
| B.4                           | VNF software modification relationship to NS and NSD management.....                   | 178        |
| B.4.1                         | Introduction .....   | 178        |
| <b>Annex C (informative):</b> | <b>Change History .....</b>  | <b>180</b> |
| History .....                 |  | 192        |



---

## Intellectual Property Rights

### Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is 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 IPR Policy, no investigation, 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.

---

## 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 specifies the interfaces supported over the Or-Vnfm reference point of the Network Functions Virtualisation Management and Orchestration (NFV-MANO) architectural framework ETSI GS NFV-MAN 001 [i.7] 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 006: "Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Vi-Vnfm reference point - Interface and Information Model Specification".
- [2] ETSI GS NFV-IFA 010: "Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Functional requirements specification".
- [3] ETSI GS NFV-IFA 011: "Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; VNF Descriptor and Packaging 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 3; 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 GS NFV 002: "Network Functions Virtualisation (NFV); Architectural Framework".
- [i.2] ETSI GS NFV 003: "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".
- [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 3; Management and Orchestration; Or-Vi reference point - Interface and Information Model Specification".

- [i.5] ETSI GS NFV-IFA 008: "Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Ve-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] ETSI GS NFV-MAN 001: "Network Functions Virtualisation (NFV); Management and Orchestration".
- [i.8] ETSI GS NFV-IFA 013: "Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Os-Ma-Nfvo reference point - Interface and Information Model Specification".
- [i.9] Void.
- [i.10] ETSI GS NFV-REL 006: "Network Functions Virtualisation (NFV) Release 3; Reliability; Maintaining Service Availability and Continuity Upon Software Modification".

## 3 Definition of terms, symbols and abbreviations

### 3.1 Terms

For the purposes of the present document, the terms given in ETSI GS NFV 003 [i.2] apply.

### 3.2 Symbols

Void.

### 3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GS NFV 003 [i.2] and the following apply:

**NOTE:** An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in ETSI GS NFV 003 [i.2].

|           |  |
|-----------|--|
| CP        | Connection Point   |
| CPD       | Connection Point Descriptor                                      |
| DF        | Deployment Flavour   |
| FB        | Functional Block   |
| NFVI-Node | Network Functions Virtualisation Infrastructure Node             |
| NFVI-PoP  | Network Function Virtualisation Infrastructure Point of Presence |
| VDU       | VNF Deployment Unit  |
| VL        | Virtual Link   |
| VLD       | Virtual Link Descriptor  |

---

## 4 Overview of interfaces and information elements associated to the Or-Vnfm reference point

### 4.1 Introduction

This clause provides an overview of interfaces and information elements associated to the Or-Vnfm reference point.

The Or-Vnfm reference point is used for exchanges between Network Functions Virtualisation Orchestrator (NFVO) and Virtualised Network Function Manager (VNFM), and supports the following interfaces:

- Virtualised Network Function (VNF) Package Management (produced by NFVO, consumed by VNFM).
- VNF Lifecycle Operation Granting (produced by NFVO, consumed by VNFM).
- Virtualised Resources Management (produced by NFVO, consumed by VNFM).
- Virtualised Resources Quota Available Notification (produced by NFVO, consumed by VNFM).
- VNF Lifecycle Management (produced by VNFM, consumed by NFVO).
- VNF Performance Management (produced by VNFM, consumed by NFVO).
- VNF Fault Management (produced by VNFM, consumed by NFVO).
- VNF Indicator (produced by VNFM, consumed by NFVO).
- Policy Management (produced by VNFM, consumed by NFVO).
- VNF Snapshot Package Management (produced by NFVO, consumed by VNFM).

The information elements exchanged by 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 NFV Group Specifications:

- Report on Architectural Options ETSI GS NFV-IFA 009 [i.6]:
  - This report describes architectural options that can influence the way some of the Or-Vnfm interfaces are used or might even suggest the need for extension.
- Functional Requirements Specification ETSI GS NFV-IFA 010 [2]:
  - Interfaces associated with the Or-Vnfm reference point are based on the functional requirements specified in ETSI GS NFV-IFA 010 [2] for the NFVO and VNFM Functional Blocks (FBs).
- Vi-Vnfm reference point - Interface and Information Model Specification ETSI GS NFV-IFA 006 [1]:
  - The interfaces related to Virtualised Resources Management defined in ETSI GS NFV-IFA 006 [1] are also used on the Or-Vnfm reference point.
- Ve-Vnfm reference point - Interface and Information Model Specification ETSI GS NFV-IFA 008 [i.5]:
  - VNF Fault Management, VNF Performance Management and VNF Indicator interfaces defined in ETSI GS NFV-IFA 008 [i.5] are also used on the Or-Vnfm reference point.
- VNF Packaging Specification ETSI GS NFV-IFA 011 [3]:
  - The specification of the Virtualised Network Function Descriptor (VNFD) in ETSI GS NFV-IFA 011 [3] defines information elements that are also relevant in the present document.

- Os-Ma-nfvo reference point - Interface and Information Model Specification ETSI GS NFV-IFA 013 [i.8]:
  - The VNF Package Management interface defined in ETSI GS NFV-IFA 013 [i.8] is also used on the Or-Vnfm reference point.

Information about the reference points in the ETSI NFV architecture can be found in ETSI GS NFV 002 [i.1].

## 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, VirtualLink 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 or references requirements applicable to interfaces in the specific context of the Or-Vnfm reference point.

## 5.2 Or-Vnfm reference point requirements

Table 5.2-1 specifies requirements applicable to the Or-Vnfm reference point.

**Table 5.2-1: Or-Vnfm reference point requirements**

| Numbering   | Requirement  |
|-------------|--|
| Or-Vnfm.001 | The Or-Vnfm reference point shall support the VNF Package Management interface produced by the NFVO.   |
| Or-Vnfm.002 | The Or-Vnfm reference point shall support the VNF Lifecycle Operation Granting interface produced by the NFVO.   |
| Or-Vnfm.003 | When VNF-related resource management in indirect mode is applicable, the Or-Vnfm reference point shall support the Virtualised Resources Information Management interfaces produced by the NFVO.         |
| Or-Vnfm.004 | When VNF-related resource management in indirect mode is applicable, the Or-Vnfm reference point shall support the Virtualised Resources Management interfaces produced by the NFVO.                     |
| Or-Vnfm.005 | When VNF-related resource management in indirect mode is applicable, the Or-Vnfm reference point shall support the Virtualised Resources Change Notification interfaces produced by the NFVO.            |
| Or-Vnfm.006 | When VNF-related resource management in indirect mode is applicable, the Or-Vnfm reference point shall support the Virtualised Resources Reservation interfaces produced by the NFVO.                    |
| Or-Vnfm.007 | When VNF-related resource management in indirect mode is applicable, the Or-Vnfm reference point shall support the Virtualised Resources Reservation Change Notification interface produced by the NFVO. |
| Or-Vnfm.008 | When VNF-related resource management in indirect mode is applicable, the Or-Vnfm reference point shall support the Virtualised Resources Performance Management interface produced by the NFVO.          |
| Or-Vnfm.009 | When VNF-related resource management in indirect mode is applicable, the Or-Vnfm reference point shall support the Virtualised Resources Fault Management interface produced by the NFVO.                |
| Or-Vnfm.010 | When VNF-related resource management in indirect mode is applicable, the Or-Vnfm reference point shall support the Virtualised Resources Quota Management interfaces produced by the NFVO.               |
| Or-Vnfm.011 | When VNF-related resource management in indirect mode is applicable, the Or-Vnfm reference point shall support the Virtualised Resources Quota Change Notification interface produced by the NFVO.       |
| Or-Vnfm.012 | The Or-Vnfm reference point shall support the VNF Lifecycle Management interface produced by the VNFM.   |
| Or-Vnfm.013 | Void.  |
| Or-Vnfm.014 | The Or-Vnfm reference point shall support the VNF Performance Management interface produced by the VNFM.   |
| Or-Vnfm.015 | The Or-Vnfm reference point shall support the VNF Fault Management interface produced by the VNFM.   |
| Or-Vnfm.016 | Void.  |
| Or-Vnfm.017 | The Or-Vnfm reference point shall support the VNF Indicator interface produced by the VNFM.  |
| Or-Vnfm.018 | The Or-Vnfm reference point should support the Virtualised Resources Quota Available Notification interface produced by the NFVO.  |
| Or-Vnfm.019 | The Or-Vnfm reference point shall support the Policy Management interface produced by the VNFM.  |
| Or-Vnfm.020 | The Or-Vnfm reference point shall support the VNF Snapshot Package Management interface produced by the NFVO.  |

## 5.3 Interface requirements

### 5.3.1 VNF Package Management interface requirements

Table 5.3.1-1 specifies requirements applicable to the VNF Package Management interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.1-1: VNF Package Management interface requirements**

| Numbering           | Requirement  |
|---------------------|--|
| Or-Vnfm.VnfPkgm.001 | The VNF Package Management interface produced by the NFVO on the Or-Vnfm reference point shall support querying VNF Package information (see note).  |
| Or-Vnfm.VnfPkgm.002 | The VNF Package Management interface produced by the NFVO on the Or-Vnfm reference point shall support providing notifications as a result of changes on VNF Package states, and managing subscriptions to such notifications. |
| Or-Vnfm.VnfPkgm.003 | The VNF Package Management interface produced by the NFVO on the Or-Vnfm reference point shall support providing notifications about the on-boarding of VNF Packages, and managing subscriptions to such notifications.        |
| Or-Vnfm.VnfPkgm.004 | The VNF Package Management interface produced by the NFVO on the Or-Vnfm reference point shall support fetching a VNF Package, or selected artifacts contained in a VNF Package.   |
| NOTE:               | 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.  |

## 5.3.2 VNF Lifecycle Operation Granting interface requirements

Table 5.3.2-1 specifies requirements applicable to the VNF Lifecycle Operation Granting interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.2-1: VNF Lifecycle Operation Granting interface requirements**

| Numbering           | Requirement  |
|---------------------|--|
| Or-Vnfm.VnfLcog.001 | The VNF Lifecycle Operation Granting interface produced by the NFVO on the Or-Vnfm reference point shall support granting lifecycle operations.  |
| Or-Vnfm.VnfLcog.002 | The VNF Lifecycle Operation Granting interface produced by the NFVO on the Or-Vnfm reference point shall support indicating the type of the lifecycle event for which a granting is being requested for a VNF instance, together with an identifier of the lifecycle operation occurrence.             |
| Or-Vnfm.VnfLcog.003 | The VNF Lifecycle Operation Granting interface produced by the NFVO on the Or-Vnfm reference point shall enable the VNFM to indicate the virtualised resources impacted by the VNF lifecycle operation (e.g. allocated or released).   |
| Or-Vnfm.VnfLcog.004 | The VNF Lifecycle Operation Granting interface produced by the NFVO on the Or-Vnfm reference point shall enable the VNFM obtaining information about the identification and configuration information to access the Virtualised Infrastructure Manager (VIM).  |
| Or-Vnfm.VnfLcog.005 | The VNF Lifecycle Operation Granting interface produced by the NFVO on the Or-Vnfm reference point shall enable the VNFM obtaining, if a reservation is applicable, resource reservation identification information applicable for consuming virtualised resources as part of the lifecycle operation. |
| Or-Vnfm.VnfLcog.006 | The VNF Lifecycle Operation Granting interface produced by the NFVO on the Or-Vnfm reference point shall enable the VNFM to provide information to identify the VNF Instance and VNFD for the intended lifecycle operation.  |

## 5.3.3 Virtualised Resources Management interfaces requirements

### 5.3.3.1 Virtualised Resources Information Management interfaces requirements

When VNF-related resource management in indirect mode is applicable, the Virtualised Resources Information Management interfaces as produced by the VIM on the Vi-Vnfm reference point are produced by the NFVO on the Or-Vnfm reference point.

Table 5.3.3.1-1 specifies requirements applicable to the Virtualised Resources Information Management interfaces produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.1-1: Virtualised Resources Information Management interfaces requirements**

| Numbering       | Requirement   |
|-----------------|---|
| Or-Vnfm.Vrim.01 | The Virtualised Resources Information Management interfaces produced by the NFVO on the Or-Vnfm reference point shall support the NFVO receiving indication information to enable the NFVO to invoke the virtualised resources information management operations towards the appropriate VIM (see notes 1 and 2). |
| NOTE 1:         | The Virtualised Resources Information Management interface requirements defined in clause 5.3.2 in ETSI GS NFV-IFA 006 [1] are applicable in the present clause too, in addition to the requirement(s) above.   |
| NOTE 2:         | The indication information is used by the NFVO to determine the entity responsible for the management of the virtualised resources.   |

### 5.3.3.2 Virtualised Resources Management interfaces requirements

When VNF-related resource management in indirect mode is applicable, the Virtualised Resources Management interfaces as produced by the VIM on the Vi-Vnfm reference point are produced by the NFVO on the Or-Vnfm reference point.

Table 5.3.3.2-1 specifies requirements applicable to the Virtualised Resources Management interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.2-1: Virtualised Resources Management interfaces requirements**

| Numbering      | Requirement   |
|----------------|---|
| Or-Vnfm.Vrm.01 | The Virtualised Resources Management interfaces produced by the NFVO on the Or-Vnfm reference point shall support the NFVO receiving indication information to enable the NFVO to invoke the virtualised resources management operations towards the appropriate VIM (see notes 1 and 2). |
| NOTE 1:        | The Virtualised Resources Management interfaces requirements defined in clause 5.3.3 in ETSI GS NFV-IFA 006 [1] are applicable in the present clause too, in addition to the requirement(s) above.  |
| NOTE 2:        | The indication information is used by the NFVO to determine the entity responsible for the management of the virtualised resources.   |

### 5.3.3.3 Virtualised Resources Reservation Management interface requirements

When VNF-related resource management in indirect mode is applicable, the Virtualised Resources Reservation Management interface as produced by the VIM on the Vi-Vnfm reference point is produced by the NFVO on the Or-Vnfm reference point.

Table 5.3.3.3-1 specifies requirements applicable to the Virtualised Resources Reservation Management interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.3-1: Virtualised Resources Reservation Management interface requirements**

| Numbering       | Requirement  |
|-----------------|--|
| Or-Vnfm.Vrrm.01 | The Virtualised Resources Reservation Management interface produced by the NFVO on the Or-Vnfm reference point shall support the NFVO receiving indication information to enable the NFVO to invoke the virtualised resources reservation management operations towards the appropriate VIM (see notes 1 and 2). |
| NOTE 1:         | The Virtualised Resources Reservation Management interfaces requirements defined in clause 5.3.4 in ETSI GS NFV-IFA 006 [1] are applicable in the present clause too, in addition to the requirement(s) above.   |
| NOTE 2:         | The indication information is used by the NFVO to determine the entity responsible for the management of the virtualised resources.  |

### 5.3.3.4 Virtualised Resources Reservation Change Notification interface requirements

When VNF-related resource management in indirect mode is applicable, the Virtualised Resources Reservation Change Notification interface as produced by the VIM on the Vi-Vnfm reference point is produced by the NFVO on the Or-Vnfm reference point.



Table 5.3.3.4-1 specifies requirements applicable to the Virtualised Resources Reservation Change Notification interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.4-1: Virtualised Resources Reservation Change Notification interface requirements**

| Numbering  | Requirement  |
|--|--|
| Or-Vnfm.Vrrcn.01   | The Virtualised Resources Reservation Change Notification interface produced by the NFVO on the Or-Vnfm reference point shall support the NFVO receiving indication information to enable the NFVO to identify the original provider of notifications, and to allow the VNFM to uniquely determine the virtualised resource reservation(s) to which a change notification applies (see notes 1 and 2). |
| NOTE 1: The Virtualised Resources Reservation Change Notification interface requirements defined in clause 5.3.6 in ETSI GS NFV-IFA 006 [1] are applicable in the present clause too, in addition to the requirement(s) above. |  |
| NOTE 2: The indication information is used by the NFVO to determine the entity responsible for the management of the virtualised resources.  |  |

### 5.3.3.5 Virtualised Resources Change Notification interfaces requirements

When VNF-related resource management in indirect mode is applicable, the Virtualised Resources Change Notification interface as produced by the VIM on the Vi-Vnfm reference point are produced by the NFVO on the Or-Vnfm reference point.

Table 5.3.3.5-1 specifies requirements applicable to the Virtualised Resources Change Notification interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.5-1: Virtualised Resources Change Notification interface requirements**

| Numbering  | Requirement   |
|--|---|
| Or-Vnfm.Vrcn.01  | The Virtualised Resources Change Notification interfaces produced by the NFVO on the Or-Vnfm reference point shall support the NFVO receiving indication information to enable the NFVO to identify the original provider of notifications, and to allow the VNFM to uniquely determine the virtualised resource(s) to which a change notification applies (see notes 1 and 2). |
| NOTE 1: The Virtualised Resources Change Notification interface requirements defined in clause 5.3.5 in ETSI GS NFV-IFA 006 [1] are applicable in the present clause too, in addition to the requirement(s) above. |   |
| NOTE 2: The indication information is used by the NFVO to determine the entity responsible for the management of the virtualised resources.  |   |

### 5.3.3.6 Virtualised Resources Performance Management interface requirements

When VNF-related resource management in indirect mode is applicable, the Virtualised Resources Performance Management interface as produced by the VIM on the Vi-Vnfm reference point is produced by the NFVO on the Or-Vnfm reference point.

Table 5.3.3.6-1 specifies requirements applicable to the Virtualised Resources Performance Management interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.6-1: Virtualised Resources Performance Management interface requirements**

| Numbering   | Requirement  |
|---|--|
| Or-Vnfm.Vrpm.01   | The Virtualised Resources Performance Management interface produced by the NFVO on the Or-Vnfm reference point shall support the NFVO receiving indication information to enable the NFVO to identify the original provider of PM information, and to allow the VNFM to uniquely determine the virtualised resource(s) to which such PM information applies (see notes 1 and 2). |
| NOTE 1: The Virtualised Resources Performance Management interface requirements defined in clause 5.3.8 in ETSI GS NFV-IFA 006 [1] are applicable in the present clause too, in addition to the requirement(s) above. |  |
| NOTE 2: The indication information is used by the NFVO to determine the entity responsible for the management of the virtualised resources.   |  |

### 5.3.3.7 Virtualised Resources Fault Management interface requirements

When VNF-related resource management in indirect mode is applicable, the Virtualised Resources Fault Management interface as produced by the VIM on the Vi-Vnfm reference point is produced by the NFVO on the Or-Vnfm reference point.

Table 5.3.3.7-1 specifies requirements applicable to the Virtualised Resources Fault Management interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.7-1: Virtualised Resources Fault Management interface requirements**

| Numbering   | Requirement   |
|---|---|
| Or-Vnfm.Vrfm.01   | The Virtualised Resources Fault Management interface produced by the NFVO on the Or-Vnfm reference point shall support the NFVO receiving indication information to enable the NFVO to identify the original provider of alarms, and to allow the VNFM to uniquely determine the virtualised resource(s) to which an alarm applies (see notes 1 and 2). |
| NOTE 1: The Virtualised Resources Fault Management interface requirements defined in clause 5.3.9 in ETSI GS NFV-IFA 006 [1] are applicable in the present clause too, in addition to the requirement(s) above. |   |
| NOTE 2: The indication information is used by the NFVO to determine the entity responsible for the management of the virtualised resources.   |   |

### 5.3.3.8 Virtualised Resources Quota Management interfaces requirements

When VNF-related resource management in indirect mode is applicable, the Virtualised Resources Quota Management interfaces as produced by the VIM on the Vi-Vnfm reference point are produced by the NFVO on the Or-Vnfm reference point.

Table 5.3.3.8-1 specifies requirements applicable to the Virtualised Resources Quota Management interfaces produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.8-1: Virtualised Resources Quota Management interface requirements**

| Numbering  | Requirement   |
|--|---|
| Or-Vnfm.Vrqm.01  | The Virtualised Resources Quota Management interfaces produced by the NFVO on the Or-Vnfm reference point shall support the NFVO receiving indication information to enable the NFVO to invoke the virtualised resources quota management operations towards the appropriate VIM (see notes 1 and 2). |
| NOTE 1: The Virtualised Resources Quota Management interfaces requirements defined in clause 5.3.7 in ETSI GS NFV-IFA 006 [1] are applicable in the present clause too, in addition to the requirement(s) above. |   |
| NOTE 2: The indication information is used by the NFVO to determine the entity responsible for the management of the virtualised resources quota.  |   |

### 5.3.3.9 Virtualised Resources Quota Change Notification interface requirements

When VNF-related resource management in indirect mode is applicable, the Virtualised Resources Quota Change Notification interface as produced by the VIM on the Vi-Vnfm reference point is produced by the NFVO on the Or-Vnfm reference point.

Table 5.3.3.9-1 specifies requirements applicable to the Virtualised Resources Quota Change Notification interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.9-1: Virtualised Resources Quota Change Notification interface requirements**

| Numbering       | Requirement   |
|-----------------|---|
| Or-Vnfm.Vrcn.01 | The Virtualised Resources Quota Change Notification interface produced by the NFVO on the Or-Vnfm reference point shall support notification of changes related to virtualised resource quotas.   |
| Or-Vnfm.Vrcn.02 | The Virtualised Resources Quota Change Notification interface produced by the NFVO on the Or-Vnfm reference point shall support the NFVO receiving indication information to enable the NFVO to identify the original provider of notifications, and to allow the VNFM to uniquely determine the virtualised resources quota to which a change notification applies (see note). |
| NOTE:           | The indication information is used by the NFVO to determine the entity responsible for the management of the virtualised resources quota.   |

### 5.3.3.10 Virtualised Resources Quota Available Notification interface requirements

Table 5.3.3.10-1 specifies requirements applicable to the Virtualised Resources Quota Available Notification interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.3.10-1: Virtualised Resources Quota Available Notification interface requirements**

| Numbering        | Requirement  |
|------------------|--|
| Or-Vnfm.Vrqan.01 | The Virtualised Resources Quota Available Notification interface produced by the NFVO on the Or-Vnfm reference point should support the capability to notify the availability of virtualised resource quota(s) applicable to this VNFM or the VNF(s) which the VNFM manages and to manage subscriptions to notifications about the availability of such quota. |

### 5.3.4 VNF Lifecycle Management interface requirements

Table 5.3.4-1 specifies requirements applicable to the VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point.

**Table 5.3.4-1: VNF Lifecycle Management interface requirements**

| Numbering          | Requirement  |
|--------------------|--|
| Or-Vnfm.VnfLcm.001 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support instantiating a VNF.  |
| Or-Vnfm.VnfLcm.002 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support terminating a VNF instance.   |
| Or-Vnfm.VnfLcm.003 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support scaling a VNF instance.   |
| Or-Vnfm.VnfLcm.004 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support querying information about a VNF instance.  |
| Or-Vnfm.VnfLcm.005 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support requesting VNF healing.   |
| Or-Vnfm.VnfLcm.006 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support requesting to change the state of a VNF instance (see note 1).  |
| Or-Vnfm.VnfLcm.007 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support querying the status of a VNF lifecycle management operation.  |
| Or-Vnfm.VnfLcm.008 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support changing the deployment flavour (DF) of a VNF instance.   |
| Or-Vnfm.VnfLcm.009 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support modifying information about a VNF instance (see note 2).  |
| Or-Vnfm.VnfLcm.010 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support creating a VNF instance identifier and the associated instance of a VNF information element.  |
| Or-Vnfm.VnfLcm.011 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support deleting a VNF instance identifier and the associated instance of a VNF information element.  |
| Or-Vnfm.VnfLcm.012 | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support providing to the NFVO notifications about changes of a VNF instance that are related to VNF lifecycle management operation occurrences, further referred to as VNF lifecycle management operation occurrence notifications. |

| Numbering   | Requirement   |
|---|---|
| Or-Vnfm.VnfLcm.013  | VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall contain information about the type of VNF lifecycle management operation, the identification of the VNF instance, and the identification of the lifecycle management operation occurrence.   |
| Or-Vnfm.VnfLcm.014  | VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall contain information about the addition/deletion of VNFCs, and about the changes on virtualised resources associated to Virtualised Network Function Component(s) (VNFC(s)) as result of the VNF lifecycle management operation occurrence. |
| Or-Vnfm.VnfLcm.015  | VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall contain information about the virtual networks and Connection Points (CPs) that are added/deleted as part of the VNF lifecycle management operation occurrence (see note 3).   |
| Or-Vnfm.VnfLcm.016  | VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support indicating the start of the lifecycle management operation occurrence the end and the results of the lifecycle management operation occurrence including any error produced from the lifecycle management operation occurrence.    |
| Or-Vnfm.VnfLcm.017  | VNF lifecycle management operation occurrence notifications provided on the VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support indicating updates to the VNF instance information including configurable properties.  |
| Or-Vnfm.VnfLcm.018  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support providing to the NFVO notifications about creation and deletion of a VNF identifier and the associated instance of a VNF information element, further referred to as VNF identifier creation/deletion notifications.   |
| Or-Vnfm.VnfLcm.019  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support managing subscriptions to VNF lifecycle management operation occurrence notifications and to VNF identifier creation/deletion notifications.   |
| Or-Vnfm.VnfLcm.020  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support providing to the VNFM configuration parameters for a VNF instance. See note 4.   |
| Or-Vnfm.VnfLcm.021  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support requesting to change the external connectivity of a VNF instance.  |
| Or-Vnfm.VnfLcm.022  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support the capability to invoke VNF error handling operations after the VNF life cycle operation occurrence fails. See notes 5 and 6.   |
| Or-Vnfm.VnfLcm.023  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support creating VNF Snapshots.  |
| Or-Vnfm.VnfLcm.024  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support reverting to VNF Snapshots.  |
| Or-Vnfm.VnfLcm.025  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support querying information about available VNF Snapshots (see note 7).   |
| Or-Vnfm.VnfLcm.026  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support deleting information associated to VNF Snapshots.  |
| Or-Vnfm.VnfLcm.027  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support changing the current VNF package.  |
| Or-Vnfm.vnfLcm.028  | The VNF Lifecycle Management interface produced by the VNFM on the Or-Vnfm reference point shall support fetching the VNF state snapshot associated to a VNF snapshot.  |
| NOTE 1: 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 2: The requirement refers to the information that is writable.   |   |
| NOTE 3: This provides information about virtual networks and connections points that are internal to the VNF and whose creation was triggered by the VNFM.  |   |
| NOTE 4: Configuration parameters referred to in this clause are declared in the VNFD. They include: those that are set prior to instantiation and that cannot be modified if the VNF is instantiated, those that are set prior to instantiation (are part of initial configuration) and that can be modified later, and those that can be set only after instantiation. |   |
| NOTE 5: It is up to the protocol design stage to design the detailed error handling operations.   |   |
| NOTE 6: It depends on the VNF capabilities whether and how the operations are supported by a particular VNF.  |   |
| NOTE 7: VNF Snapshot information can include information including creation date, configuration data of included VNF Snapshots, and identifiers of snapshotted VNF instances.   |   |

### 5.3.5 Void

### 5.3.6 VNF Performance Management interface requirements

Table 5.3.6-1 specifies requirements applicable to the VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point.

**Table 5.3.6-1: VNF Performance Management interface requirements**

| Numbering  | Requirement  |
|--|--|
| Or-Vnfm.VnfPm.001  | The VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point shall support the NFVO to control the collection and reporting of VNF performance information, resulting from virtualised resources performance information, on the VNF(s) it manages (see note 1). |
| Or-Vnfm.VnfPm.002  | The VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point shall support the capability to notify the availability of VNF performance information.   |
| Or-Vnfm.VnfPm.003  | The VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point shall support the NFVO to create a PM job specifying the VNF performance information that the NFVO requires from the VNFM.  |
| Or-Vnfm.VnfPm.004  | The VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point shall support the NFVO to delete one or more PM job(s).   |
| Or-Vnfm.VnfPm.005  | The VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point shall enable the NFVO to receive notifications of data availability for a PM job, and to manage subscriptions to such notifications.  |
| Or-Vnfm.VnfPm.006  | The VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point shall support the NFVO to query the details of one or more PM job(s).   |
| Or-Vnfm.VnfPm.007  | The VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point shall support the NFVO to manage the thresholds on specified VNF performance information and VNF(s) (see note 2).   |
| Or-Vnfm.VnfPm.008  | The VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point shall support the capability to notify about a threshold defined for a specified metric of a VNF being crossed.   |
| Or-Vnfm.VnfPm.009  | The VNF Performance Management interface produced by the VNFM on the Or-Vnfm reference point shall enable the NFVO to receive notifications related to threshold crossing, and to manage subscriptions to such notifications.  |
| NOTE 1: Performance information on a given VNF results from collected performance information of the virtualised resources that are mapped to this VNF instance. |  |
| NOTE 2: Management of thresholds include creation, deletion and query the thresholds on specified VNF performance information and VNF(s).                        |  |

### 5.3.7 VNF Fault Management interface requirements

Table 5.3.7-1 specifies requirements applicable to the VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point.

**Table 5.3.7-1: VNF Fault Management interface requirements**

| Numbering         | Requirement  |
|-------------------|--|
| Or-Vnfm.VnfFm.001 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall support collecting VNF fault information (see note).  |
| Or-Vnfm.VnfFm.002 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall support providing alarm notifications related to faults on VNF instances.                         |
| Or-Vnfm.VnfFm.003 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall support providing notification when there is a change in alarm information on VNF instances.      |
| Or-Vnfm.VnfFm.004 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall support the sending of notification to the NFVO when an alarm on a VNF instance has been created. |
| Or-Vnfm.VnfFm.005 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall support the sending of notification to the NFVO when an alarm on a VNF instance has been cleared. |
| Or-Vnfm.VnfFm.006 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall allow unambiguous identification of the alarm on a VNF instance sent to the NFVO.                 |
| Or-Vnfm.VnfFm.007 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall allow unambiguous identification of the VNF instance causing the alarm.                           |

| Numbering         | Requirement  |
|-------------------|--|
| Or-Vnfm.VnfFm.008 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall allow unambiguous identification of the alarm cause.  |
| Or-Vnfm.VnfFm.009 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall support providing to the NFVO notifications about alarms on a VNF instance as a consequence of state changes in the virtualised resources used by the VNF.  |
| Or-Vnfm.VnfFm.010 | Notifications related to the alarms associated with state changes of virtualised resources of a VNF instance provided on the VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall contain information necessary to identify the VNF and the VNFC(s), the origin (VIM and virtualised resource(s)) of the virtualised resource change notification(s), the type of alarm, and information about the cause of the alarm. |
| Or-Vnfm.VnfFm.011 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall enable the NFVO to manage subscriptions to notifications related to alarms.   |
| Or-Vnfm.VnfFm.012 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall support alarm acknowledgement.  |
| Or-Vnfm.VnfFm.013 | The VNF Fault Management interface produced by the VNFM on the Or-Vnfm reference point shall support the sending of notification(s) to the NFVO when the alarm list has been rebuilt.  |
| NOTE:             | Fault information on a given VNF instance can include the information related to the alarm (e.g. alarm created, alarm cleared, etc.), alarm causes and identification of this VNF instance and fault information concerning the virtualised resources supporting the constituent VNF instance.   |

### 5.3.8 Void

### 5.3.9 VNF Indicator interface requirements

Table 5.3.9-1 specifies requirements applicable to the VNF Indicator interface produced by the VNFM on the Or-Vnfm reference point.

**Table 5.3.9-1: VNF Indicator interface requirements**

| Numbering          | Requirement   |
|--------------------|---|
| Or-Vnfm.VnfInd.001 | The VNF Indicator interface produced by the VNFM on the Or-Vnfm reference point shall support providing notifications related to indicator value change, and to manage subscriptions related to such notifications. |
| Or-Vnfm.VnfInd.002 | The VNF Indicator interface produced by the VNFM on the Or-Vnfm reference point shall support retrieving indicator values.  |

### 5.3.10 Policy Management interface requirements

Table 5.3.10-1 specifies requirements applicable to the policy management interface produced by the VNFM on the Or-vnfm reference point.

**Table 5.3.10-1: Policy management interface requirements**

| Numbering  | Requirement   |
|--|---|
| Or-Vnfm.Plcm.001   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support transferring NFV-MANO policies. See notes 1 and 2.  |
| Or-Vnfm.Plcm.002   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support deleting NFV-MANO policies. See note 1.   |
| Or-Vnfm.Plcm.003   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support querying NFV-MANO policies. See note 1.   |
| Or-Vnfm.Plcm.004   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support activating NFV-MANO policies. See note 1.   |
| Or-Vnfm.Plcm.005   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support deactivating NFV-MANO policies. See note 1.   |
| Or-Vnfm.Plcm.006   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support providing to the NFVO 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. |
| Or-Vnfm.Plcm.007   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support providing to the NFVO notifications about any detected policy conflicts.  |
| Or-Vnfm.Plcm.008   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support subscribing to policy management related notifications.   |
| Or-Vnfm.Plcm.009   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support associating NFV-MANO policies to VNF instances.   |
| Or-Vnfm.Plcm.010   | The Policy Management interface produced by the VNFM on the Or-Vnfm reference point shall support disassociating NFV-MANO policies from VNF instances.  |
| NOTE 1: For this reference point, NFV-MANO policies include policies applied in VNF lifecycle management (instantiation, scaling, healing and termination).  |   |
| NOTE 2: The case of transferring NFV-MANO policy applies when:   |   |
| <ul style="list-style-type: none"> <li>- a new policy is imported from the NFVO, which results in the creation of a new policy locally; or</li> <li>- the changes for an existing policy are imported from the NFVO, which results in the update of a policy locally.</li> </ul> |   |

### 5.3.11 VNF Snapshot Package Management interface requirements

Table 5.3.11-1 specifies requirements applicable to the VNF Snapshot Package Management interface produced by the NFVO on the Or-Vnfm reference point.

**Table 5.3.11-1: VNF Snapshot Package Management interface requirements**

| Numbering  | Requirement   |
|--|---|
| Or-Vnfm.VnfSnapPkgm.001  | Void.   |
| Or-Vnfm.VnfSnapPkgm.002  | Void.   |
| Or-Vnfm.VnfSnapPkgm.003  | Void.   |
| Or-Vnfm.VnfSnapPkgm.004  | Void.   |
| Or-Vnfm.VnfSnapPkgm.005  | The VNF Snapshot Package Management interface produced by the NFVO on the Or-Vnfm reference point shall support querying information about available VNF Snapshot Packages (see note).                      |
| Or-Vnfm.VnfSnapPkgm.006  | The VNF Snapshot Package Management interface produced by the NFVO on the Or-Vnfm reference point shall support fetching a VNF Snapshot Package, or selected artifacts contained in a VNF Snapshot Package. |
| Or-Vnfm.VnfSnapPkgm.007  | Void.   |
| Or-Vnfm.VnfSnapPkgm.008  | Void.   |
| 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. |   |

## 6 NFVO exposed interfaces

### 6.1 Introduction

This clause defines the interfaces exposed by the NFVO towards the VNFM over the Or-Vnfm 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 can be encoded in the URL, in the message header, in the message body or any combination thereof.

## 6.2 VNF Package Management interface

### 6.2.1 Description

This interface allows the VNFM to access VNF Package information.

The interface also includes a notify operation for new VNF Package on-boarding or for VNF Package changes, and operations to manage subscriptions to such notifications.

### 6.2.2 Query VNF Package Info operation

#### 6.2.2.1 Description

This operation will enable the VNFM to query the NFVO for details of one or more VNF Package information objects. Table 6.2.2.1-1 lists the information flow exchanged between the NFVO and the VNFM.

NOTE: The VNFD is an attribute of the VnfPkgInfo.

**Table 6.2.2.1-1: Query VNF Package Info operation**

| Message                 | Requirement | Direction   |
|-------------------------|-------------|-------------|
| QueryVnfPkgInfoRequest  | Mandatory   | VNFM → NFVO |
| QueryVnfPkgInfoResponse | Mandatory   | NFVO → VNFM |

#### 6.2.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.2.2.2-1.

**Table 6.2.2.2-1: 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 the VnfPkgInfo.<br>It can also be used to specify one or more VNF Package information objects to be queried by providing their vnfld 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 the vnfPkgInfo matching the filter. If absent, the complete vnfPkgInfo is returned.   |
| NOTE: The vnfld, 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 [3], 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. |           |             |         |   |

#### 6.2.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.2.2.3-1.



**Table 6.2.2.3-1: Query VNF Package Info operation output parameters**

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

## 6.2.2.4 Operation results

After successful 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.

## 6.2.3 Subscribe operation

### 6.2.3.1 Description

This operation enables the VNFM to subscribe with a filter for the notifications related to new VNF Package on-boarded or to changes of VNF Packages sent by the NFVO.

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

Table 6.2.3.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.2.3.1-1: Subscribe operation**

| Message           | Requirement | Direction   |
|-------------------|-------------|-------------|
| SubscribeRequest  | Mandatory   | VNFM → NFVO |
| SubscribeResponse | Mandatory   | NFVO → VNFM |

### 6.2.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.2.3.2-1.

**Table 6.2.3.2-1: Subscribe operation input parameters**

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

### 6.2.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.2.3.3-1.

**Table 6.2.3.3-1: Subscribe operation output parameters**

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

### 6.2.3.4 Operation results

After successful subscription, the VNFM 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 VNFM.

## 6.2.4 Notify operation

### 6.2.4.1 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 (VNFM).

In order to receive notifications, the VNFM shall have a subscription.

Table 6.2.4.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.2.4.1-1: Notify operation**

| Message | Requirement | Direction   |
|---------|-------------|-------------|
| Notify  | Mandatory   | NFVO → VNFM |

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

- VnfPackageOnBoardingNotification (see clause 8.2.4).
- VnfPackageChangeNotification (see clause 8.2.5).

### 6.2.5 Void

## 6.2.6 Fetch VNF Package operation

### 6.2.6.1 Description

This operation enables the VNFM to fetch a whole on-boarded VNF Package from the NFVO. The package is addressed using an identifier of information held by the NFVO about the specific on-boarded VNF Package. This identifier is contained within the VnfPackageOnBoardingNotification or is returned as a result of Query VNF Package Info operation.

Table 6.2.6.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.2.6.1-1: Fetch VNF Package operation**

| Message                 | Requirement | Direction   |
|-------------------------|-------------|-------------|
| FetchVnfPackageRequest  | Mandatory   | VNFM → NFVO |
| FetchVnfPackageResponse | Mandatory   | NFVO → VNFM |

### 6.2.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.2.6.2-1.

**Table 6.2.6.2-1: Fetch VNF Package operation input parameters**

| Parameter    | Qualifier | Cardinality | Content    | Description  |
|--------------|-----------|-------------|------------|--|
| vnfPkgInfold | 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. |

### 6.2.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.2.6.3-1.

**Table 6.2.6.3-1: Fetch VNF Package operation output parameters**

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

### 6.2.6.4 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 VNFM a copy of the requested VNF Package.

## 6.2.7 Fetch VNF Package Artifacts operation

### 6.2.7.1 Description

This operation enables the VNFM to fetch selected artifacts contained in an on-boarded VNF Package. Artifacts are addressed using selector information that can be obtained using the QueryVNF Package Info operation.

**NOTE:** The VNFD is an attribute of the OnboardedVnfPkgInfo and it is retrieved, if queried individually, with the QueryVnfPackageInfo operation. Fetching the whole VNF Package will also return the VNFD, which is part of the VNF Package.

Table 6.2.7.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.2.7.1-1: Fetch VNF Package Artifacts operation**

| Message                          | Requirement | Direction   |
|----------------------------------|-------------|-------------|
| FetchVnfPackageArtifactsRequest  | Mandatory   | VNFM → NFVO |
| FetchVnfPackageArtifactsResponse | Mandatory   | NFVO → VNFM |

### 6.2.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.2.7.2-1.

**Table 6.2.7.2-1: Fetch VNF Package Artifacts operation input parameters**

| Parameter  | Qualifier | Cardinality | Content       | Description  |
|--|-----------|-------------|---------------|--|
| vnfPkgInfold   | M         | 1           | Identifier    | Identifier of the VNF Package information object associated with the VNF Package artifacts to be fetched. This identifier was allocated by the NFVO. |
| artifactSelector   | M         | 1..N        | Not specified | Selector to address an individual VNF package artifact, or list of selectors to address multiple of those. See note.                                 |
| <b>NOTE:</b> 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. |           |             |               |  |

### 6.2.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.2.7.3-1.

**Table 6.2.7.3-1: Fetch VNF Package Artifacts operation output parameters**

| Parameter   | Qualifier | Cardinality | Content       | Description   |
|---|-----------|-------------|---------------|---|
| vnfPackageArtifact  | M         | 1..N        | Not specified | A VNF package artifact (e.g. files). 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. |           |             |               |   |

### 6.2.7.4 Operation results

The result of the operation indicates whether the fetching of the VNF Package Artifacts has been successful or not in the NFVO with a standard success/error result. After successful operation, the NFVO has provided to the VNFM a copy/copies of the requested artifact(s) contained in the on-boarded VNF Package.

## 6.2.8 Terminate Subscription operation

### 6.2.8.1 Description

This operation enables the VNFM to terminate a particular subscription.

Table 6.2.8.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.2.8.1-1: Terminate Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| TerminateSubscriptionRequest  | Mandatory   | VNFM → NFVO |
| TerminateSubscriptionResponse | Mandatory   | NFVO → VNFM |

### 6.2.8.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.2.8.2-1.

**Table 6.2.8.2-1: Terminate Subscription operation input parameters**

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

### 6.2.8.3 Output parameters

None.

### 6.2.8.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the VNFM 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.

## 6.2.9 Query Subscription Info operation

### 6.2.9.1 Description

This operation enables the VNFM to query information about subscriptions.

Table 6.2.9.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.2.9.1-1: Query Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| QuerySubscriptionInfoRequest  | Mandatory   | VNFM → NFVO |
| QuerySubscriptionInfoResponse | Mandatory   | NFVO → VNFM |

### 6.2.9.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.2.9.2-1.

**Table 6.2.9.2-1: 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. |

### 6.2.9.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.2.9.3-1.

**Table 6.2.9.3-1: 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. |

### 6.2.9.4 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 onboarding or VNF package changes that the VNFM has access to and that are matching the filter shall be returned.

## 6.3 VNF Lifecycle Operation Granting interface

### 6.3.1 Description

This interface defines one operation that allows the NFVO to grant lifecycle operations.

### 6.3.2 Grant VNF Lifecycle Operation operation

#### 6.3.2.1 Description

This operation allows the VNFM to request a grant for authorization of a VNF lifecycle operation. This interface supports multiple use cases, such as:

- The NFVO can approve or reject a request based on policies (e.g. dependencies between VNFs) and available capacity.

- When applicable, the NFVO can reserve resources based on the VNFM's virtualised resources request. Depending on operator policies, the NFVO can decide on whether to reserve virtualised resources or physical compute hosts.
- The NFVO can provide to the VNFM information about the VIM where cloud resources are allocated. This can include additional information such as the resource zone.

When requesting resource creation or modification, the VNFM references the resource definitions that are available to the NFVO in the VNFD. When resources are to be released or modified, the VNFM provides references to the existing resources in the request.

Per each VNFM, one of the following operator policies can be selected as a configuration to determine how the NFVO and the VNFM handle resource reservations in a grant request:

- 1) Policy GRANT\_RESERVE: The NFVO guarantees the availability of the VIM resources to be allocated. The NFVO provides to the VNFM reservation identifier(s). Each such identifier identifies the reservation which is applicable to the resource requirements and which the VNFM shall use in the subsequent resource management operation.
- 2) Policy GRANT\_APPROVE: The NFVO approves the VIM resources to be allocated by the VNFM. In general, resource availability is not guaranteed. No explicit reservation identifier is returned to the VNFM. Optionally, to guarantee resource availability, the NFVO may do a reservation and use implicit reservation identification towards the VNFM, i.e. associate the reservation to the VIM access information.

These policies are used to configure the behaviour of both the NFVO and the VNFM identically, also considering the resource reservation capabilities of the VIM:

- resource definitions refer to: either a resource template in the VNFD (VnfVirtualLinkDesc, VirtualComputeDesc, VirtualStorageDesc plus Vdu, if applicable) for the creation of new resources; or
- to information about an existing resource.

In the GrantVnfLifecycleOperation response, the NFVO can return information that allows to distribute the resources of a VNF over multiple resource zones. This decision is guided by affinity/anti-affinity rules in the VNFD as well as by placement constraints passed in the GrantVnfLifecycleOperation request. The NFVO can also return information that allows to manage the resources of a VNF using multiple VIMs, guided by VIM selection constraints passed in the GrantVnfLifecycleOperation request.

NOTE: In the present document, as part of that mechanism, attributes are defined for signalling the decision to use multiple VIMs per VNF. However, to actually support VNFs that include resources managed by multiple VIMs, additionally a mechanism is needed to manage the VNF-internal Virtual Link (VL) requirements across multiple VIMs. Such functionality is not specified; neither in the present document, nor in other documents referenced by the present document. Also, the current mechanism of signalling external and externally-managed VLs in the lifecycle management operations assumes single-VIM VNFs, and does not fulfil the requirements of multi-VIM scenarios.

Table 6.3.2.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.3.2.1-1: Grant VNF Lifecycle Operation operation**

| Message                            | Requirement | Direction   |
|------------------------------------|-------------|-------------|
| GrantVnfLifecycleOperationRequest  | Mandatory   | VNFM → NFVO |
| GrantVnfLifecycleOperationResponse | Mandatory   | NFVO → VNFM |

### 6.3.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.3.2.2-1.

**Table 6.3.2.2-1: Grant VNF Lifecycle Operation operation input parameters**

| Parameter                      | Qualifier | Cardinality | Content    | Description  |
|--------------------------------|-----------|-------------|------------|--|
| vnfInstanceId                  | M         | 1           | Identifier | Identifier of the VNF instance which this grant request is related to. Shall also be provided for VNFs that not yet exist but are planned to exist in the future, i.e. if the grant is requested for InstantiateVNF.   |
| vnfdId                         | M         | 1           | Identifier | Identifier of the VNFD that defines the VNF for which the LCM operation is to be granted.<br>In case the operation changes the current VNF Package, this identifier refers to the VNFD which defines the VNF before the LCM operation to be granted.   |
| dstVnfdId                      | M         | 0..1        | Identifier | Identifier of the "destination" VNFD which will define the VNF after executing the LCM operation to be granted. Shall be included if the operation changes the current VNF Package and shall be absent otherwise.  |
| flavourId                      | M         | 0..1        | Identifier | Identifier of the VNF deployment flavour (DF) of the VNFD that defines the VNF for which the LCM operation is to be granted.<br>Shall be provided when instantiating the VNF or changing the DF of the VNF instance.   |
| lifecycleOperation             | M         | 1           | Enum       | The lifecycle management operation for which granting is requested.<br>VALUES: <ul style="list-style-type: none"> <li>• InstantiateVnf</li> <li>• ScaleVnf</li> <li>• ScaleVnfToLevel</li> <li>• ChangeVnfFlavour</li> <li>• TerminateVnf</li> <li>• HealVnf</li> <li>• OperateVnf</li> <li>• ChangeExtVnfConnectivity</li> <li>• CreateSnapshot</li> <li>• RevertToSnapshot</li> <li>• ChangeCurrentVnfPackage</li> </ul> See note 1. |
| isAutomaticInvocation          | M         | 1           | Boolean    | Set to true if this VNF LCM operation occurrence has been triggered by an automated procedure inside the VNFM (i.e. ScaleVnf/ScaleVnfToLevel triggered by auto-scale, or HealVnf triggered by auto-heal).<br>Set to false otherwise.   |
| lifecycleOperationOccurrenceId | M         | 1           | Identifier | The identifier of the VNF lifecycle operation occurrence associated to the GrantVnfLifecycleOperationRequest.  |

| Parameter   | Qualifier | Cardinality | Content             | Description  |
|---|-----------|-------------|---------------------|--|
| instantiationLevelId  | M         | 0..1        | Identifier          | If the granting request is requested for InstantiateVNF, the identifier of the instantiation level may be provided as an alternative way to define the resources to be added. This attribute shall only be used for Instantiate VNF requests.<br>See note 2.   |
| addResource   | M         | 0..N        | ResourceDefinition  | List of resource definitions in the VNFD for resources to be added by the LCM operation which is related to this grant request, with one entry per resource.<br>See note 2.  |
| tempResource  | M         | 0..N        | ResourceDefinition  | List of resource definitions in the VNFD for resources to be temporarily instantiated during the runtime of the LCM operation which is related to this grant request, with one entry per resource (see note 3).  |
| removeResource  | M         | 0..N        | ResourceDefinition  | Provides the definitions of resources to be removed by the LCM operation which is related to this grant request, with one entry per resource.  |
| updateResource  | M         | 0..N        | ResourceDefinition  | Provides the definitions of resources to be modified by the LCM operation which is related to this grant request, with one entry per resource.   |
| placementConstraint   | M         | 0..N        | PlacementConstraint | Placement constraints that the VNFM may send to the NFVO in order to influence the resource placement decision. If sent, the NFVO shall take the constraints into consideration when making resource placement decisions, and shall reject the grant if they cannot be honoured (see notes 4, 5 and 6).  |
| vimConstraint   | CM        | 0..N        | VimConstraint       | Used by the VNFM to require that multiple resources are managed through the same VIM connection. If sent, the NFVO shall take the constraints into consideration when making VIM selection decisions, and shall reject the grant if they cannot be honoured.<br>This parameter shall be supported if VNF-related Resource Management in direct mode is applicable. |
| additionalParam   | M         | 0..N        | KeyValuePair        | Additional parameters passed by the VNFM, specific to the VNF and the LCM operation.   |
| <p>NOTE 1: The VNF LCM operations CreateVnfIdentifier, DeleteVnfIdentifier, QueryVnf, ModifyVnfInformation, ConfirmVnfSnapshot, QuerySnapshotInfo, and DeleteSnapshotInfo can be executed by the VNFM without requesting granting.</p> <p>NOTE 2: If the granting request is for InstantiateVNF, either instantiationLevel or addResource shall be present.</p> <p>NOTE 3: The NFVO will assume that the VNFM will be responsible to both allocate and release the temporary resource during the runtime of the LCM operation. This means, the resource can be allocated and consumed after the "start" notification for the LCM operation is sent by the VNFM, and the resource will be-released before the "result" notification of the VNF LCM operation is sent by the VNFM.</p> <p>NOTE 4: The affinity/anti-affinity rules defined in the VNFD using the AffinityOrAntiAffinityGroup and the LocalAffinityOrAntiAffinityRule information elements (see ETSI GS NFV-IFA 011 [3]), and the placement constraints in the GrantVnfLifecycleOperation as defined in this clause should be conflict-free. In case of conflicts, the placement constraints in the GrantVnfLifecycleOperation shall take precedence.</p> <p>NOTE 5: Passing constraints allows the VNFM or the lifecycle management scripts to influence resource placement decisions by the NFVO to ensure VNF properties such as performance or fault tolerance.</p> <p>NOTE 6: If fallbackBestEffort is present in placement constraints and set to "true", the NFVO processes the Affinity/AntiAffinity constraint in a best effort manner, in which case, if specified resources cannot be allocated based on specified placement constraint, the NFVO looks for an alternate best effort placement for the specified resources to be granted.</p> |           |             |                     |  |



### 6.3.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.3.2.3-1.

**Table 6.3.2.3-1: Grant VNF Lifecycle Operation operation output parameters**

| Parameter             | Qualifier | Cardinality | Content                   | Description  |
|-----------------------|-----------|-------------|---------------------------|--|
| vimConnection         | CM        | 0..N        | VimConnectionInfo         | Provides information regarding VIM connections that are approved to be used by the VNFM to allocate resources, and provides parameters of these VIM connections. Absent in case of rejection or if the VIM information was configured to the VNFM in another way, present otherwise. This parameter shall be supported when VNF-related Resource Management in direct mode is applicable (see note 1). |
| zone                  | M         | 0..N        | ZoneInfo                  | Identifies resource zones where the resources are approved to be allocated by the VNFM. Absent in case of rejection, present otherwise.  |
| zoneGroup             | M         | 0..N        | ZoneGroupInfo             | Information about groups of resource zones that are related and that the NFVO has chosen to fulfil a zoneGroup constraint in the GrantVnfLifecycleOperation request. This information confirms that the NFVO has honoured the zoneGroup constraints that were passed as part of "placementConstraints" in the Grant request.   |
| addResource           | M         | 0..N        | GrantInfo                 | List of resources that are approved to be added, with one entry per resource.  |
| tempResource          | M         | 0..N        | GrantInfo                 | List of resources that are approved to be temporarily instantiated during the runtime of the lifecycle operation, with one entry per resource.   |
| removeResource        | M         | 0..N        | GrantInfo                 | List of resources that are approved to be removed, with one entry per resource.  |
| updateResource        | M         | 0..N        | GrantInfo                 | List of resources that are approved to be modified, with one entry per resource.   |
| vimAssets             | M         | 0..1        | VimAssets                 | Information about assets for the VNF that are managed by the NFVO in the VIM, such as software images and virtualised compute resource flavours.   |
| extVirtualLink        | M         | 0..N        | ExtVirtualLinkData        | Information about external VLS to connect the VNF to (see note 4).   |
| extManagedVirtualLink | M         | 0..N        | ExtManagedVirtualLinkData | Information about internal VLS that are managed by other entities than the VNFM (see notes 3 and 4).   |
| additionalParam       | M         | 0..N        | KeyValuePair              | Additional parameters passed by the NFVO, specific to the VNF and the LCM operation.   |

| Parameter | Qualifier | Cardinality | Content   | Description |
|-----------|-----------|-------------|---|-------------|
| NOTE 1:   |           |             | This interface allows to signal the use of multiple VIM connections per VNF. However, due to the partial support of this feature in the present document (see clause 6.3.2.1), the specification for managing the VNF-internal VL requirements across multiple VIMs is needed, which is not available neither in the present document, nor in other documents referenced by the present document. Therefore, it is recommended in the present document that the number of "vimConnection" attributes in the response is not greater than 1. |             |
| NOTE 2:   |           |             | Void.   |             |
| NOTE 3:   |           |             | 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 4:   |           |             | External and/or externally-managed internal VLs can be passed in VNF lifecycle management operation requests such as InstantiateVnf, ChangeVnfFlavor, ChangeVnfExtConnectivity or ChangeCurrentVnfPackage, and/or in the grant response. The NFVO may choose to override in the grant response external and/or externally-managed VL instances that have been passed previously in the associated VNF lifecycle management request, if the lifecycle management request has originated from the NFVO itself.                                |             |

### 6.3.2.4 Operation results

In case of permitting the operation, the NFVO returns to the VNFM additional information to be used in the resource management operations during the lifecycle management operation.

Once the NFVO has responded positively with a GrantVnfLifecycleOperationResponse, the VNFM executes the necessary resource management operations either towards the appropriate VIM(s) (aka VNF-related resource management in direct mode) or towards the NFVO which proxies them to the appropriate VIM(s) (aka VNF-related resource management in indirect mode).

In addition to failure situations, the NFVO can reject a GrantVnfLifecycleOperationRequest due to various reasons, such as resource unavailability or operational policy. In case of rejecting the operation or in case of failure, the NFVO returns to the VNFM appropriate error information, describing the reason of rejection or failure.

If placement constraints have been passed with the request, the NFVO shall process the constraints as below:

- If fallbackBestEffort is not present or set to "false" in placement constraints, and if and the NFVO cannot satisfy the constraints, it shall reject the request.
- If fallbackBestEffort is present and set to "true" in placement constraints, the NFVO shall process the Affinity/AntiAffinity constraint in a best effort manner, in which case, if specified resources cannot be allocated based on specified placement constraint, the NFVO looks for an alternate best effort placement for the specified resources to be granted. In the best effort anti-affinity case, the resources are spread optimally over all available instances of scope (e.g. zones), and in the best effort affinity case, they are distributed optimally over fewer possible instances of scope.

## 6.4 Virtualised Resources Management interfaces in indirect mode

### 6.4.1 Introduction

In indirect mode of VNF-related resource management, the NFVO produces towards the VNFM the virtualised resource management interfaces defined below.

These interfaces are related to the corresponding interfaces defined in ETSI GS NFV-IFA 006 [1]; however, an additional *resource provider identifier* is introduced. This identifier is used by the NFVO to determine the entity responsible for the management of the virtualised resource, the management of the virtualised resources reservation or the management of the virtualised resources quota (usually one of multiple VIMs with which the NFVO interacts). It is used by the VNFM to uniquely identify resources, resource reservations or resource quotas by means of the pair of the resource provider identifier and the actual identifier of the resource/reservation/quota.

## 6.4.2 Virtualised Compute interfaces

### 6.4.2.1 Virtualised Compute Resources Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Compute Resources Management to VNFM. This interface shall comply with the provisions in clause 7.3.1 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content `VirtualCompute` used in output parameters in clause 7.3.1 of ETSI GS NFV-IFA 006 [1] is replaced with `ComputeResourceWithRpInfo` as defined in clause 8.4.2.2 of the present document.
- For the `Terminate Virtualised Compute Resource` operation the content of both the input and output parameters is changed from `Identifier` to `ComputeResourceWithRpId` as defined in clause 8.4.2.3 of the present document.
- All operations except `Query Virtualised Compute Resource` and `Terminate Virtualised Compute Resource` have an additional input parameter, `resourceProviderId`, defined in table 6.4.2.1-1.

**Table 6.4.2.1-1: Definition of the `resourceProviderId` input parameter for compute resources**

| Parameter                       | Qualifier | Cardinality | Content    | Description   |
|---------------------------------|-----------|-------------|------------|---|
| <code>resourceProviderId</code> | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources by means of the tuple [ <code>resourceProviderId</code> , <code>computedId</code> ]. |

### 6.4.2.2 Virtualised Compute Resources Change Notification interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Compute Resources Change Notifications to be consumed by VNFM. This interface shall comply with the provisions in clause 7.3.2 of ETSI GS NFV-IFA 006 [1] and the related information elements with the following changes:

- The notification `VirtualisedResourceChangeNotification` sent by means of the `Notify` operation of clause 7.3.2.3 of ETSI GS NFV-IFA 006 [1] is replaced with the notification `VirtualisedResourceWithRpChangeNotification` defined in clause 8.4.5.2 of the present document.

### 6.4.2.3 Virtualised Compute Resources Information Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Compute Resources Information Management to VNFM. This interface shall comply with the provisions in clause 7.3.3 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content `VirtualComputeResourceInformation` used in output parameters for the `Query Virtualised Compute Resource Information` operation in clause 7.3.3.4 of ETSI GS NFV-IFA 006 [1] is replaced with `VirtualComputeResourceWithRpInfo` as defined in clause 8.4.2.4 of the present document.
- The notification `InformationChangeNotification` sent by means of the `Notify` operation of clause 7.3.3.3 of ETSI GS NFV-IFA 006 [1] is replaced with the notification `InformationWithRpChangeNotification` defined in clause 8.4.5.3 of the present document.

## 6.4.3 Virtualised Network interfaces

### 6.4.3.1 Virtualised Network Resources Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Network Resources Management to VNFM. This interface shall comply with the provisions in clause 7.4.1 of ETSI GS NFV-IFA 006 [1] with the following change(s):

- The content `VirtualNetwork` used in output parameters in clause 7.4.1 of ETSI GS NFV-IFA 006 [1] is replaced by `NetworkResourceWithRpInfo` as defined in clause 8.4.3.2 of the present document.

- For the Terminate Virtualised Network Resource operation the content of both the input and output parameter is changed from Identifier to NetworkResourceWithRpId as defined in clause 8.4.3.3 of the present document.
- All operations except Query Virtualised Network Resource and Terminate Virtualised Network Resource have an additional input parameter, resourceProviderId, defined in table 6.4.3.1-1.

**Table 6.4.3.1-1: Definition of the resourceProviderId input parameter for network resources**

| Parameter          | Qualifier | Cardinality | Content    | Description   |
|--------------------|-----------|-------------|------------|---|
| resourceProviderId | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, networkResourceId]. |

### 6.4.3.2 Virtualised Network Resources Change Notification interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Network Resources Change Notifications to be consumed by VNFM. This interface shall comply with the provisions in clause 7.4.2 of ETSI GS NFV-IFA 006 [1] and the related information elements with the following changes:

- The notification VirtualisedResourceChangeNotification sent by means of the Notify operation of clause 7.4.2.3 of ETSI GS NFV-IFA 006 [1] is replaced with the notification VirtualisedResourceWithRpChangeNotification defined in clause 8.4.5.2 of the present document.

### 6.4.3.3 Virtualised Network Resources Information Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Network Resources Information Management to VNFM. This interface shall comply with the provisions in clause 7.4.3 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content VirtualNetworkResourceInformation used in output parameters for the Query Virtualised Network Resource Information operation in clause 7.4.3.4 of ETSI GS NFV-IFA 006 [1] is replaced with VirtualNetworkResourceWithRpInfo as defined in clause 8.4.3.4 of the present document.
- The notification InformationChangeNotification sent by means of the Notify operation of clause 7.4.3.3 of ETSI GS NFV-IFA 006 [1] is replaced with the notification InformationWithRpChangeNotification defined in clause 8.4.5.3 of the present document.

## 6.4.4 Virtualised Storage interfaces

### 6.4.4.1 Virtualised Storage Resources Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Storage Resources Management to VNFM. This interface shall comply with the provisions in clause 7.5.1 of ETSI GS NFV-IFA 006 [1] with the following change(s):

- The content VirtualStorage used in output parameters in clause 7.5.1 of ETSI GS NFV-IFA 006 [1] is replaced by StorageResourceWithRpInfo as defined in clause 8.4.4.2 of the present document.
- For the Terminate Virtualised Storage Resource operation the content of both the input and output parameter is changed from Identifier to StorageResourceWithRpId as defined in clause 8.4.4.3 of the present document.
- All operations except Query Virtualised Storage Resource and Terminate Virtualised Storage Resource have an additional input parameter, resourceProviderId, defined in table 6.4.4.1-1.

**Table 6.4.4.1-1: Definition of the resourceProviderId input parameter for storage resources**

| Parameter          | Qualifier | Cardinality | Content    | Description  |
|--------------------|-----------|-------------|------------|--|
| resourceProviderId | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, storagelId]. |

#### 6.4.4.2 Virtualised Storage Resources Change Notification interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Storage Resources Change Notifications to be consumed by VNFM. This interface shall comply with the provisions in clause 7.5.2 of ETSI GS NFV-IFA 006 [1] and the related information elements with the following changes:

- The notification VirtualisedResourceChangeNotification sent by means of the Notify operation of clause 7.5.2.3 of ETSI GS NFV-IFA 006 [1] is replaced with the notification VirtualisedResourceWithRpChangeNotification defined in clause 8.4.5.2 of the present document.

#### 6.4.4.3 Virtualised Storage Resources Information Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Storage Resources Information Management to VNFM. This interface shall comply with the provisions in clause 7.5.3 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content VirtualStorageResourceInformation used in output parameters for the Query Virtualised Storage Resources Information operation in clause 7.5.3.4 of ETSI GS NFV-IFA 006 [1] is replaced with VirtualStorageResourceWithRpInfo as defined in clause 8.4.4.4 of the present document.
- The notification InformationChangeNotification sent by means of the Notify operation of clause 7.5.3.3 of ETSI GS NFV-IFA 006 [1] is replaced with the notification InformationWithRpChangeNotification defined in clause 8.4.5.3 of the present document.

#### 6.4.5 Virtualised Resource Performance Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Resource Performance Management to be consumed by VNFM. This interface shall comply with the provisions in clause 7.7 of ETSI GS NFV-IFA 006 [1] and the related information elements with the following changes:

- The operations Create PM Job and Create Threshold have an additional input element, resourceProviderId, defined in table 6.4.5-1, with the value received in the response to the GrantVnfLifecycleOperation request.
- The notification PerformanceInformationAvailableNotification notified/sent by means of the Notify operation of clause 7.7.6 of ETSI GS NFV-IFA 006 [1] is replaced with the notification PerformanceInformationWithRpAvailableNotification defined in clause 8.4.6.2 of the present document.
- The notification ThresholdCrossedNotification notified/sent by means of the Notify operation of clause 7.7.6 of ETSI GS NFV-IFA 006 [1] is replaced with the notification ThresholdCrossedWithRpNotification defined in clause 8.4.6.3 of the present document.

**Table 6.4.5-1: Definition of the resourceProviderId input parameter for virtual resource performance information**

| Parameter          | Qualifier | Cardinality | Content    | Description  |
|--------------------|-----------|-------------|------------|--|
| resourceProviderId | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the virtualised resource performance information and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, storagelId]. |

## 6.4.6 Virtualised Resource Fault Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Resource Fault Management to be consumed by VNFM. This interface shall comply with the provisions in clause 7.6 of ETSI GS NFV-IFA 006 [1] and the related information elements with the following changes:

- The content Alarm used in the output parameters of the Get Alarm List operation of clause 7.6.4 of ETSI GS NFV-IFA 006 [1] is replaced with AlarmWithRpInfo as defined in clause 8.4.7.2 of the present document in order to distinguish between alarms from different VIM instances managed by the NFVO.
- The notification AlarmNotification published/notified/sent by means of the Notify operation of clause 7.6.3 of ETSI GS NFV-IFA 006 [1] is replaced with the notification AlarmWithRpNotification defined in clause 8.4.7.3 of the present document.
- The notification AlarmClearedNotification published/notified/sent by means of the Notify operation of clause 7.6.3 of ETSI GS NFV-IFA 006 [1] is replaced with the notification AlarmClearedWithRpNotification defined in clause 8.4.7.4 of the present document.

## 6.4.7 Virtualised Resources Quota Management interfaces

### 6.4.7.1 Virtualised Compute Resources Quota Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Compute Resources Quota Management to the VNFM. This interface shall comply with the provisions in clause 7.9.1 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content VirtualComputeQuota used in output parameters in clause 7.9.1 of ETSI GS NFV-IFA 006 [1] is replaced with VirtualComputeQuotaWithRpInfo as defined in clause 8.4.8.2 of the present document.

### 6.4.7.2 Virtualised Network Resources Quota Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Network Resources Quota Management to the VNFM. This interface shall comply with the provisions in clause 7.9.2 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content VirtualNetworkQuota used in output parameters in clause 7.9.2 of ETSI GS NFV-IFA 006 [1] is replaced with VirtualNetworkQuotaWithRpInfo as defined in clause 8.4.8.3 of the present document.

### 6.4.7.3 Virtualised Storage Resources Quota Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Storage Resources Quota Management to the VNFM. This interface shall comply with the provisions in clause 7.9.3 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content VirtualStorageQuota used in output parameters in clause 7.9.3 of ETSI GS NFV-IFA 006 [1] is replaced with VirtualStorageQuotaWithRpInfo as defined in clause 8.4.8.4 of the present document.

### 6.4.7.4 Virtualised Resources Quota Change Notification interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Resources Quota Change Notification to be consumed by the VNFM. This interface shall comply with the provisions in clause 7.9.4 of ETSI GS NFV-IFA 006 [1] and the related information elements with the following changes:

- The notification VirtualisedResourceQuotaChangeNotification sent by means of the Notify operation of clause 7.9.4.3 of ETSI GS NFV-IFA 006 [1] is replaced with notification VirtualisedResourceQuotaWithRpChangeNotification defined in clause 8.4.8.5 of the present document.

## 6.4.8 Virtualised Resource Reservation interfaces

### 6.4.8.1 Virtualised Compute Resources Reservation Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Compute Resources Reservation Management to VNFM. This interface shall comply with the provisions in clause 7.8.1 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content `ReservedVirtualCompute` used in output parameters in clause 7.8.1 of ETSI GS NFV-IFA 006 [1] is replaced with `ReservedVirtualComputeWithRpInfo` as defined in clause 8.4.9.2 of the present document.

### 6.4.8.2 Virtualised Network Resources Reservation Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Network Resources Reservation Management to VNFM. This interface shall comply with the provisions in clause 7.8.2 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content `ReservedVirtualNetwork` used in output parameters in clause 7.8.2 of ETSI GS NFV-IFA 006 [1] is replaced with `ReservedVirtualNetworkWithRpInfo` as defined in clause 8.4.9.3 of the present document.

### 6.4.8.3 Virtualised Storage Resources Reservation Management interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Storage Resources Reservation Management to VNFM. This interface shall comply with the provisions in clause 7.8.3 of ETSI GS NFV-IFA 006 [1] with the following changes:

- The content `ReservedVirtualStorage` used in output parameters in clause 7.8.3 of ETSI GS NFV-IFA 006 [1] is replaced with `ReservedVirtualStorageWithRpInfo` as defined in clause 8.4.9.4 of the present document.

### 6.4.8.4 Virtualised Resources Reservation Change Notification interface

In indirect resource management mode, the NFVO produces an interface for Virtualised Resources Reservation Change Notifications to be consumed by the VNFM. This interface shall comply with the provisions in clause 7.8.4 of ETSI GS NFV-IFA 006 [1] and the related information elements with the following changes:

- The notification `VirtualisedResourceReservationChangeNotification` sent by means of the `Notify` operation of clause 7.8.4.3 of ETSI GS NFV-IFA 006 [1] is replaced with the notification `VirtualisedResourceReservationWithRpChangeNotification` defined in clause 8.4.9.5 of the present document.

## 6.5 Virtualised Resources Quota Available Notification interface

### 6.5.1 Description

This interface allows an authorized consumer FB to manage subscriptions regarding information on the availability of the virtualised resources quota(s), and to provide such notification to the subscribed consumer.

Support for this interface is optional.

The VNFM needs to issue a `Subscribe` request for `VirtualisedResourceQuotaAvailable` notifications in order to know when a quota applicable to the VNFM is available.

When a quota applicable to the consumer is available, the consumer is notified using the notification `VirtualisedResourceQuotaAvailableNotification` (see clause 8.11.2).

## 6.5.2 Subscribe operation

### 6.5.2.1 Description

This operation enables the VNFM to subscribe with a filter for the notifications related to the availability of quota on virtualised resources sent by the NFVO. Specification of filtering mechanism is part of the protocol design.

Table 6.5.2.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.5.2.1-1: Subscribe operation**

| Message           | Requirement | Direction   |
|-------------------|-------------|-------------|
| SubscribeRequest  | Mandatory   | VNFM → NFVO |
| SubscribeResponse | Mandatory   | NFVO → VNFM |

### 6.5.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.5.2.2-1.

**Table 6.5.2.2-1: Subscribe operation input parameters**

| Parameter | Qualifier | Cardinality | Content | Description   |
|-----------|-----------|-------------|---------|---|
| filter    | M         | 1           | Filter  | Input filter for selecting notifications to subscribe to. This filter can contain information about specific attributes of the virtualised resources quota. |

### 6.5.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.5.2.3-1.

**Table 6.5.2.3-1: Subscribe operation output parameters**

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

### 6.5.2.4 Operation results

After successful subscription, the VNFM is registered to receive notifications sent by the NFVO when a virtualised resources quota applicable to the VNFM is available. 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 VNFM.

## 6.5.3 Notify operation

### 6.5.3.1 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 (VNFM).

In order to receive notifications, the VNFM shall have a subscription.

Table 6.5.3.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.5.3.1-1: Notify operation**

| Message | Requirement | Direction   |
|---------|-------------|-------------|
| Notify  | Mandatory   | NFVO → VNFM |



The following notification is sent by this operation:

- VirtualisedResourceQuotaAvailableNotification. See clause 8.11.2.

## 6.5.4 Terminate Subscription operation

### 6.5.4.1 Description

This operation enables the VNFM to terminate a particular subscription.

Table 6.5.4.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.5.4.1-1: Terminate Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| TerminateSubscriptionRequest  | Mandatory   | VNFM → NFVO |
| TerminateSubscriptionResponse | Mandatory   | NFVO → VNFM |

### 6.5.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.5.4.2-1.

**Table 6.5.4.2-1: Terminate Subscription operation input parameters**

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

### 6.5.4.3 Output parameters

None.

### 6.5.4.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the VNFM 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.

## 6.5.5 Query Subscription Info operation

### 6.5.5.1 Description

This operation enables the VNFM to query information about subscriptions.

Table 6.5.5.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 6.5.5.1-1: Query Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| QuerySubscriptionInfoRequest  | Mandatory   | VNFM → NFVO |
| QuerySubscriptionInfoResponse | Mandatory   | NFVO → VNFM |

### 6.5.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.5.5.2-1.

**Table 6.5.5.2-1: 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. |

### 6.5.5.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.5.5.3-1.

**Table 6.5.5.3-1: 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. |

### 6.5.5.4 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 VNF quota availability that the VNFM has access to and that are matching the filter shall be returned.

## 6.6 VNF Snapshot Package Management interface

### 6.6.1 Description

This interface allows the VNFM to access VNF Snapshot Package information and to fetch the content of an available VNF Snapshot package from the NFVO.

### 6.6.2 Fetch VNF Snapshot Package operation

#### 6.6.2.1 Description

This operation enables the VNFM to fetch the content of a VNF Snapshot Package. The package is addressed using an identifier of information held by the NFVO about the specific VNF Snapshot Package.

Table 6.6.2.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 6.6.2.1-1: Fetch VNF Snapshot Package operation**

| Message                         | Requirement | Direction   |
|---------------------------------|-------------|-------------|
| FetchVnfSnapshotPackageRequest  | Mandatory   | VNFM → NFVO |
| FetchVnfSnapshotPackageResponse | Mandatory   | NFVO → VNFM |

#### 6.6.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.6.2.2-1.

**Table 6.6.2.2-1: Fetch VNF Snapshot Package operation input parameters**

| Parameter          | Qualifier | Cardinality | Content                                      | Description   |
|--------------------|-----------|-------------|--|---|
| vnfSnapshotPkgInfo | M         | 1           | Identifier (Reference to VnfSnapshotPkgInfo) | References the information held by the NFVO about the specific VNF Snapshot Package. This identifier was allocated by the NFVO. |

### 6.6.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.6.2.3-1.

**Table 6.6.2.3-1: Fetch VNF Snapshot Package operation output parameters**

| Parameter          | Qualifier | Cardinality | Content | Description               |
|--------------------|-----------|-------------|---------|---------------------------|
| vnfSnapshotPackage | M         | 1           | Binary  | The VNF Snapshot Package. |

### 6.6.2.4 Operation results

After successful operation, the NFVO has provided to the VNFM a copy of the requested VNF Snapshot Package.

## 6.6.3 Fetch VNF Snapshot Package Artifacts operation

### 6.6.3.1 Description

This operation enables the VNFM to fetch selected artifacts contained in a VNF Snapshot Package. Artifacts are addressed using selector information that can be obtained using the QueryVnfSnapshotPkgInfo operation. Table 6.6.3.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 6.6.3.1-1: Fetch VNF Snapshot Package Artifacts operation**

| Message                                  | Requirement | Direction   |
|--|-------------|-------------|
| FetchVnfSnapshotPackageArtifactsRequest  | Mandatory   | VNFM → NFVO |
| FetchVnfSnapshotPackageArtifactsResponse | Mandatory   | NFVO → VNFM |

### 6.6.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.6.3.2-1.

**Table 6.6.3.2-1: 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 specific 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 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. |             |  |   |

### 6.6.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.6.3.3-1.

**Table 6.6.3.3-1: Fetch VNF Snapshot Package Artifacts operation output parameters**

| Parameter                  | Qualifier   | Cardinality | Content       | Description   |
|----------------------------|---|-------------|---------------|---|
| vnfSnapshotPackageArtifact | M   | 1..N        | Not specified | A VNF Snapshot Package 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. |             |               |   |

### 6.6.3.4 Operation results

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

## 6.6.4 Query VNF Snapshot Package Information operation

### 6.6.4.1 Description

This operation enables the VNFM to query the NFVO for information about one or more VNF Snapshot Packages. Table 6.6.4.1-1 lists the information flow exchanged between the VNFM and the NFVO.

The operation allows querying specific components of the information about a VNF Snapshot Package, for instance, retrieving the `vnfSnapshotInfoId`.

NOTE: The `vnfSnapshotInfoId` is an attribute of the `VnfSnapshotPkgInfo`.

**Table 6.6.4.1-1: Query VNF Snapshot Package Information operation**

| Message                         | Requirement | Direction   |
|---------------------------------|-------------|-------------|
| QueryVnfSnapshotPkgInfoRequest  | Mandatory   | VNFM → NFVO |
| QueryVnfSnapshotPkgInfoResponse | Mandatory   | NFVO → VNFM |

### 6.6.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 6.6.4.2-1.

**Table 6.6.4.2-1: 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 <code>VnfSnapshotPkgInfo</code> . It can also be used to specify one or more VNF Snapshot Packages to be queried by providing their <code>vnfSnapshotInfoId</code> , <code>vnfcSnapshotInfoId</code> or <code>vnfSnapshotPkgInfoId</code> . See note. |
| attributeSelector  | M         | 0..N        | String  | It provides a list of attribute names of <code>VnfSnapshotPkgInfo</code> . If present, only these attributes are returned for the <code>VnfSnapshotPkgInfo</code> matching the filter. If absent, the complete <code>VnfSnapshotPkgInfo</code> is returned.  |
| NOTE: The <code>vnfSnapshotInfoId</code> , assigned by the VNFM at VNF Snapshot creation or at VNF Snapshot Package extraction, identifies the information related to a VNF Snapshot.<br>The <code>vnfSnapshotPkgInfoId</code> 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. |           |             |         |  |

### 6.6.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 6.6.4.3-1.

**Table 6.6.4.3-1: Query VNF Snapshot Package Information operation output parameters**

| Parameter   | Qualifier | Cardinality | Content                         | Description  |
|-------------|-----------|-------------|---------------------------------|--|
| queryResult | M         | 0..N        | <code>VnfSnapshotPkgInfo</code> | Details of the VNF Snapshot Packages matching the input filter. If <code>attributeSelector</code> is present, only the attributes listed in <code>attributeSelector</code> are returned for the selected entities. Cardinality is 0 if no data is matching the input filter. |

#### 6.6.4.4 Operation results

After successful operation, the NFVO has queried the 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 VNFM exposed interfaces

### 7.1 Introduction

This clause defines the interfaces exposed by the VNFM towards the NFVO over the Or-Vnfm 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 can be encoded in the URL, in the message header, in the message body or any combination thereof.

### 7.2 VNF Lifecycle Management interface

#### 7.2.1 Description

This interface allows the NFVO to invoke VNF lifecycle management operations towards the VNFM.

The following operations are defined:

- Create VNF Identifier.
- Instantiate VNF.
- Scale VNF.
- Scale VNF to Level.
- Change VNF Flavour.
- Terminate VNF.
- Delete VNF Identifier.
- Query VNF.
- Heal VNF.
- Operate VNF.
- Modify VNF Information.
- Get Operation Status.
- Change External VNF connectivity.
- Query VNF Snapshot information.
- Create VNF Snapshot.
- Revert to VNF Snapshot.
- Delete VNF Snapshot information.

- Fetch VNF state snapshot.
- Change current VNF Package.

An identifier (i.e. lifecycleOperationOccurrenceId) is generated for each VNF lifecycle operation occurrence, except for Query VNF, Create VNF Identifier, Delete VNF Identifier and Get Operation Status.

Furthermore, this interface allows the NFVO to manage subscriptions to notifications sent by the VNFM which inform about changes of a VNF instance that are related to VNF lifecycle management operation occurrences, as well as related to the creation/deletion of a VNF instance identifier and the associated instance of a VnfInfo information element. It further allows the VNFM to provide such notifications to the subscriber.

## 7.2.2 Create VNF Identifier operation

### 7.2.2.1 Description

This operation creates a VNF instance identifier, and an associated instance of a VnfInfo information element, identified by that identifier, in the NOT\_INSTANTIATED instantiation state without instantiating the VNF or doing any additional lifecycle operation(s). It allows returning right away a VNF instance identifier that can be used in subsequent lifecycle operations, like the Instantiate VNF operation.

This operation shall be supported for all VNFs.

Table 7.2.2.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.2.1-1: Create VNF Identifier operation**

| Message                     | Requirement | Direction   |
|-----------------------------|-------------|-------------|
| CreateVnfIdentifierRequest  | Mandatory   | NFVO → VNFM |
| CreateVnfIdentifierResponse | Mandatory   | VNFM → NFVO |

### 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: Create VNF Identifier operation input parameters**

| Parameter   | Qualifier | Cardinality | Content      | Description   |
|---|-----------|-------------|--------------|---|
| vnfdId  | M         | 1           | Identifier   | Identifier that identifies the VNFD which defines the VNF instance to be created. See note. |
| 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.                               |
| metadata  | M         | 0..N        | KeyValuePair | This parameter provides values for the "metadata" attribute in "VnfInfo".                   |
| 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 [3], clause 7.1.2.2. |           |             |              |   |

### 7.2.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.2.3-1.

**Table 7.2.2.3-1: Create VNF Identifier operation output parameters**

| Parameter     | Qualifier | Cardinality | Content    | Description                           |
|---------------|-----------|-------------|------------|---------------------------------------|
| vnfInstanceId | M         | 1           | Identifier | VNF instance identifier just created. |

### 7.2.2.4 Operation results

In case of success, an instance of a VnfInfo information element, in the NOT\_INSTANTIATED instantiation state, has been created and can be used in subsequent lifecycle operations and the corresponding VnfIdentifierCreationNotification has been sent. In case of failure, appropriate error information is returned.

## 7.2.3 Instantiate VNF operation

### 7.2.3.1 Description

This operation instantiates a particular DF of a VNF that has been in the NOT\_INSTANTIATED instantiation state, based on the definition in the VNFD.

This operation shall be supported for all VNFs.

Table 7.2.3.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.3.1-1: Instantiate VNF operation**

| Message                | Requirement | Direction   |
|------------------------|-------------|-------------|
| InstantiateVnfRequest  | Mandatory   | NFVO → VNFM |
| InstantiateVnfResponse | Mandatory   | VNFM → NFVO |

### 7.2.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.3.2-1.

**Table 7.2.3.2-1: Instantiate VNF operation input parameters**

| Parameter             | Qualifier | Cardinality | Content                   | Description   |
|-----------------------|-----------|-------------|---------------------------|---|
| vnfInstanceld         | M         | 1           | Identifier                | Identifier of the VNF instance.   |
| flavourId             | M         | 1           | Identifier                | Identifier of the VNF DF to be instantiated.  |
| instantiationLevelId  | M         | 0..1        | Identifier                | Identifier of the instantiation level of the DF to be instantiated. If not present, the default instantiation level as declared in the VNFD shall be instantiated.  |
| 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).  |
| vimConnectionInfo     | CM        | 0..N        | VimConnectionInfo         | Information about VIM connection(s) for managing resources for the VNF instance, or external/externally-managed virtual links.<br>This attribute shall be supported if VNF-related resource management in direct mode is applicable. In that case, this attribute shall be present if there is the need to communicate VIM connection information for external or externally-managed virtual links. |

| Parameter   | Qualifier | Cardinality | Content       | Description   |
|---|-----------|-------------|---------------|---|
| localizationLanguage  | M         | 0..1        | Not specified | Localization language of the VNF to be instantiated.<br>The localization languages supported by a VNF can be declared in the VNFD.<br>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 NFVO 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 [3]).   |
| extension   | M         | 0..N        | KeyValuePair  | This parameter provides values for the "extension" attribute in "VnfInfo" (see clause 8.5.2).   |
| vnfConfigurableProperty   | M         | 0..N        | KeyValuePair  | This parameter provides values for the VNF configurable properties attribute in the "VnfInfo" (see clause 8.5.2).   |
| <p>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].</p> <p>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 8.12.4.2).</p> |           |             |               |   |

### 7.2.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.3.3-1.

**Table 7.2.3.3-1: Instantiate VNF operation output parameters**

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

### 7.2.3.4 Operation results

In case of success, the VNF has been instantiated and initially configured, and the associated instance of a VnfInfo information element has been updated. The VNF instance is in the INSTANTIATED instantiation state. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

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

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



## 7.2.4 Scale VNF operation

### 7.2.4.1 Description

This operation provides methods to request scaling a VNF in multiple ways:

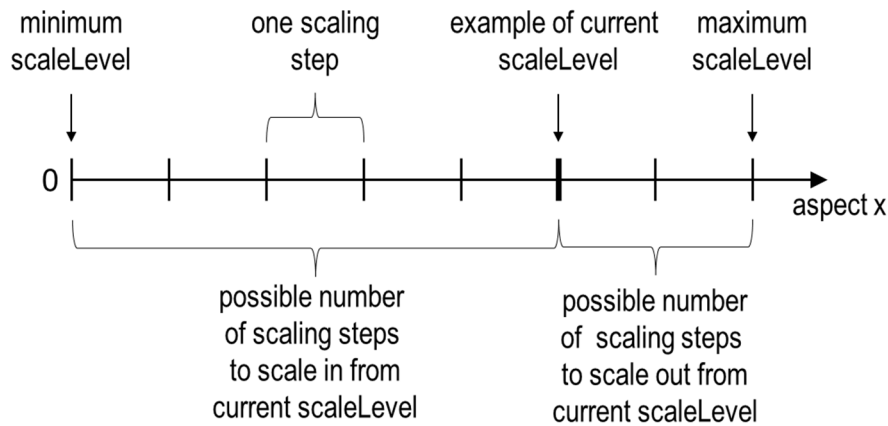
- horizontal scaling:
  - scale out: adding additional VNFC instances to the VNF to increase capacity;
  - scale in: removing VNFC instances from the VNF, in order to release unused capacity;
- vertical scaling (not supported in the present document):
  - scale up: adding further resources to existing VNFC instances, e.g. increase memory, Central Processing Unit (CPU) capacity or storage size of the virtualisation container hosting a VNFC instance, in order to increase VNF capacity;
  - scale down: removing resources from existing VNFC instances, e.g. decrease memory, CPU capacity or storage size of the virtualisation container hosting a VNFC instance, in order to release unused capacity.

Potentially, different aspects of a VNF can be scaled independently. For example, a VNF could be designed to provide static capacity such as database nodes and dynamic capacity such as query processing nodes. Such a VNF might be scaled w.r.t. two separate aspects: the 'static capacity' aspect can be scaled by adding VNFCs from VNF Deployment Units (VDUs) defining database nodes, and the 'dynamic capacity' aspect can be scaled by adding VNFCs from VDUs defining query processing nodes.

In complex VNF designs, scaling a VNF often requires adding/removing a number of related VNFC instances of several different types, possibly based on multiple VDUs. For example, in a high availability configuration, it might be required to add in each scaling step a pair of VNFC instances, one in active and one in standby configuration.

The `ScaleVnfRequest` in the interface allows the consumer to specify the scaling aspect. The scaling aspects valid for a particular VNF are defined in the VNFD. After receiving a scale request, the VNFM will figure out the necessary set of VNFCs and the related set of resources based on VNF-specific rules, for instance using the lifecycle management script associated to the Scale VNF event.

When scaling a VNF for a particular aspect, the number of scaling steps to apply to that aspect can be provided as a parameter. A scaling step is the smallest unit by which a particular aspect of a VNF can be scaled, and is mapped by the VNFM to the addition (or removal) of a certain number of resources, based on one or more VDUs. For each scaling aspect, the maximum scale level is defined in the VNFD. The minimum scale level is assumed as zero; the maximum scale level corresponds to the maximum number of steps that can be performed within this aspect, starting at the minimum scale level (i.e. zero). At each point in time between the completed VNF instantiation and the VNF termination, the "size" of the VNF w.r.t. a particular aspect can be expressed by the current scale level w.r.t. that aspect, and can be obtained, among other information, by invoking the "QueryVNF" operation. When the VNF is instantiated, the current scale level is initialized with values that are defined as part of the instantiation level in the VNFD for the associated aspect. Figure 7.2.4.1-1 illustrates the concepts described above.



**Figure 7.2.4.1-1: Illustrating the concepts of scaleLevel and scaling steps for a particular scaling aspect**

The VNFM will then communicate information about the necessary resource changes via the GrantVnfLifecycleOperationRequest to the NFVO.

It depends on the VNF capabilities, and is declared in the VNFD, whether and how this operation is supported for a particular VNF.

Table 7.2.4.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.4.1-1: Scale VNF operation**

| Message          | Requirement | Direction   |
|------------------|-------------|-------------|
| ScaleVnfRequest  | Mandatory   | NFVO → VNFM |
| ScaleVnfResponse | Mandatory   | VNFM → NFVO |

## 7.2.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.4.2-1.

**Table 7.2.4.2-1: Scale VNF operation input parameters**

| Parameter       | Qualifier | Cardinality | Content      | Description  |
|-----------------|-----------|-------------|--------------|--|
| vnfInstanceId   | M         | 1           | Identifier   | Identifier of the VNF instance to which this scaling request is related.   |
| type            | M         | 1           | Enum         | Defines the type of the scale operation requested. The set of types actually supported depends on the capabilities of the VNF being managed, as declared in the VNFD.<br>VALUES:<br><ul style="list-style-type: none"> <li>SCALE_OUT</li> <li>SCALE_IN</li> </ul> See note 1.  |
| aspectId        | M         | 1           | Identifier   | Identifies 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 to be executed as part of this ScaleVnf operation. It shall be a positive number. Defaults to 1.<br>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.<br>See note 2. |
| additionalParam | M         | 0..N        | KeyValuePair | Additional parameters passed by the NFVO as input to the scaling process, specific to the VNF being scaled as declared in the VNFD (see clause 7.1.5.4 in ETSI GS NFV-IFA 011 [3]).  |

| Parameter   | Qualifier | Cardinality | Content | Description |
|---|-----------|-------------|---------|-------------|
| NOTE 1: ETSI GS NFV-IFA 010 [2] 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: A scaling step is the smallest unit by which a VNF can be scaled w.r.t a particular scaling aspect.   |           |             |         |             |

### 7.2.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.4.3-1.

**Table 7.2.4.3-1: Scale VNF operation output parameters**

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

### 7.2.4.4 Operation results

In case of success, the VNF has been scaled according to the request, and the associated instance of a VnfInfo information element has been updated. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

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

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

## 7.2.5 Scale VNF to Level operation

### 7.2.5.1 Description

This operation scales an instantiated VNF of a particular DF 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. Typically, the result of this operation is adding and/or removing Network Functions Virtualisation Infrastructure (NFVI) resources to/from the VNF.

It depends on the VNF capabilities, and is declared in the VNFD, whether this operation is supported for a particular VNF.

Table 7.2.5.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.5.1-1: Scale VNF to Level operation**

| Message                 | Requirement | Direction   |
|-------------------------|-------------|-------------|
| ScaleVnfToLevelRequest  | Mandatory   | NFVO → VNFM |
| ScaleVnfToLevelResponse | Mandatory   | VNFM → NFVO |

### 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: Scale VNF to Level operation input parameters**

| Parameter            | Qualifier | Cardinality | Content      | Description   |
|----------------------|-----------|-------------|--------------|---|
| vnfInstanceId        | M         | 1           | Identifier   | Identifier of the VNF instance to which this scaling request is related.  |
| instantiationLevelId | M         | 0..1        | Identifier   | Identifier of the target instantiation level of the current DF to which the VNF is requested to be scaled. Either instantiationLevelId or scaleInfo but not both shall be present.  |
| scaleInfo            | M         | 0..N        | ScaleInfo    | For each scaling aspect of the current DF, defines the target scale level to which the VNF is to be scaled.<br>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.<br>Either instantiationLevelId or scaleInfo but not both shall be present. |
| additionalParam      | M         | 0..N        | KeyValuePair | Additional parameters passed by the NFVO as input to the scaling process, specific to the VNF being scaled as declared in the VNFD (see clause 7.1.5.5 in ETSI GS NFV-IFA 011 [3]).   |

### 7.2.5.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.5.3-1.

**Table 7.2.5.3-1: Scale VNF to Level operation output parameters**

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

### 7.2.5.4 Operation results

In case of success, the VNF has been scaled according to the request, and the associated instance of a VnfInfo information element has been updated. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

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

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

## 7.2.6 Change VNF Flavour operation

### 7.2.6.1 Description

This operation changes the DF of a VNF instance.

It depends on the VNF capabilities, and is declared in the VNFD, whether this operation is supported for a particular VNF. This operation may be service-disruptive.

Table 7.2.6.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.6.1-1: Change VNF Flavour operation**

| Message                  | Requirement | Direction   |
|--------------------------|-------------|-------------|
| ChangeVnfFlavourRequest  | Mandatory   | NFVO → VNFM |
| ChangeVnfFlavourResponse | Mandatory   | VNFM → NFVO |

## 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: Change VNF Flavour operation input parameters**

| Parameter   | Qualifier | Cardinality | Content                   | Description   |
|---|-----------|-------------|---------------------------|---|
| vnfInstanceId   | M         | 1           | Identifier                | Identifier of the VNF instance to be modified.  |
| newFlavourId  | M         | 1           | Identifier                | Identifier of the new VNF DF to apply to this VNF instance.   |
| instantiationLevelId  | M         | 0..1        | Identifier                | Identifier of the instantiation level of the DF to be used. If not present, the default instantiation level as declared in the VNFD shall be used.  |
| 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).  |
| vimConnectionInfo   | CM        | 0..N        | VimConnectionInfo         | Information about VIM connection(s) for managing resources for the VNF instance, or external/externally-managed virtual links.<br>This attribute shall be supported and present if VNF-related resource management in direct mode is applicable. In that case, this attribute shall be present if there is the need to communicate VIM connection information for external or externally-managed virtual links. |
| additionalParam   | M         | 0..N        | KeyValuePair              | Additional parameters passed by the NFVO 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 [3]).  |
| extension   | M         | 0..N        | KeyValuePair              | This parameter provides changes to the values for the "extension" attribute in "VnfInfo".   |
| vnfConfigurableProperty   | M         | 0..N        | KeyValuePair              | This parameter provides changes to the values for the "vnfConfigurableProperty" attribute in "VnfInfo".   |
| 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 8.12.4.2).   |           |             |                           |   |

### 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: Change VNF Flavour operation output parameters**

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

### 7.2.6.4 Operation results

In case of success, the VNF has been modified to use the new DF and initially configured, and the associated instance of a VnfInfo information element has been updated. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

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

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

## 7.2.7 Terminate VNF operation

### 7.2.7.1 Description

This operation terminates a VNF instance that has been in the INSTANTIATED instantiation state.

A VNF can be terminated gracefully or forcefully. Graceful termination means that the VNFM arranges to take the VNF out of service, e.g. by asking the VNF's EM to take the VNF out of service, and only after that shuts down the VNF and releases the resources. Forceful termination means that the VNFM immediately shuts down the VNF and releases the resources. A time interval can be specified for taking the VNF out of service, after which the VNF is shut down if taking it out of service has not completed.

Terminating a VNF instance does not delete the instance of the VnfInfo information element. This operation shall be supported for all VNFs.

Table 7.2.7.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.7.1-1: Terminate VNF operation**

| Message              | Requirement | Direction   |
|----------------------|-------------|-------------|
| TerminateVnfRequest  | Mandatory   | NFVO → VNFM |
| TerminateVnfResponse | Mandatory   | VNFM → NFVO |

### 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: Terminate VNF operation input parameters

| Parameter  | Qualifier | Cardinality | Content      | Description  |
|--|-----------|-------------|--------------|--|
| vnfInstanceId  | M         | 1           | Identifier   | Identifier of the VNF instance to be terminated.   |
| terminationType  | M         | 1           | Enum         | <p>Signals whether forceful or graceful termination is requested.</p> <p>VALUES:</p> <ul style="list-style-type: none"> <li>• FORCEFUL</li> <li>• GRACEFUL</li> </ul> <p>In case of forceful termination, the VNF is shut down immediately, and resources are released (see note 1).</p> <p>In case of graceful termination, 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 was successful, or after a timeout, the VNFM shuts down the VNF and releases the resources.</p> |
| gracefulTerminationTimeout   | M         | 0..1        | TimeDuration | <p>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.</p> <p>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 shutting down the VNF and releasing the resources (see note 2).</p> <p>Minimum timeout or timeout range are specified by the VNF Provider (e.g. defined in the VNFD or communicated by other means).</p> <p>Not relevant in case of forceful termination.</p>                            |
| additionalParam  | M         | 0..N        | KeyValuePair | Additional parameters passed by the NFVO as input to the Terminate VNF operation, specific to the VNF being terminated as declared in the VNFD (see clause 7.1.5.7 in ETSI GS NFV-IFA 011 [3]).  |
| NOTE 1: If the VNF is still in service, this can adversely impact network service, and therefore, operator policies apply to determine if forceful termination is allowed in the particular situation. |           |             |              |  |
| NOTE 2: This implies that no VNF shutdown and resource release will be attempted if taking the VNF out of service fails or hangs.  |           |             |              |  |

### 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: Terminate VNF operation output parameters

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

### 7.2.7.4 Operation results

In case of success, the VNF instance has been terminated and resources used by the VNF have been released, and the associated instance of a VnfInfo information element has been updated. The VNF instance is in the NOT\_INSTANTIATED instantiation state. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

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

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

## 7.2.8 Delete VNF Identifier operation

### 7.2.8.1 Description

This operation deletes a VNF instance identifier and the associated instance of a VnfInfo information element in the NOT\_INSTANTIATED instantiation state.

This operation shall be supported for all VNFs.

Table 7.2.8.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.8.1-1: Delete VNF Identifier operation**

| Message                     | Requirement | Direction   |
|-----------------------------|-------------|-------------|
| DeleteVnfIdentifierRequest  | Mandatory   | NFVO → VNFM |
| DeleteVnfIdentifierResponse | Mandatory   | VNFM → NFVO |

### 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: Delete VNF Identifier operation input parameters**

| Parameter     | Qualifier | Cardinality | Content    | Description                           |
|---------------|-----------|-------------|------------|---------------------------------------|
| vnfInstanceId | M         | 1           | Identifier | VNF instance identifier to be deleted |

### 7.2.8.3 Output parameters

No output parameter.

### 7.2.8.4 Operation results

In case of success, the VNF instance identifier and the associated instance of the VnfInfo information element has been deleted and can no longer be used; and the corresponding VnfIdentifierDeletionNotification has been sent. If the VNF instance was not terminated (i.e. the VNF is in INSTANTIATED instantiation state), the operation shall be rejected.

In case of failure, appropriate error information is returned.

## 7.2.9 Query VNF operation

### 7.2.9.1 Description

This operation provides information about VNF instances. The applicable VNF instances can be chosen based on filtering criteria, and the information can be restricted to selected attributes.



This operation shall be supported for all VNFs.

Table 7.2.9.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.9.1-1: Query VNF operation**

| Message          | Requirement | Direction   |
|------------------|-------------|-------------|
| QueryVnfRequest  | Mandatory   | NFVO → VNFM |
| QueryVnfResponse | Mandatory   | VNFM → NFVO |

### 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: Query VNF operation input parameters**

| Parameter         | Qualifier | Cardinality | Content | Description   |
|-------------------|-----------|-------------|---------|---|
| filter            | M         | 1           | Filter  | Filter to select the VNF instance(s) about which information is queried.  |
| attributeSelector | M         | 0..N        | String  | Provides a list of attribute names. If present, only these attributes are returned for the VNF instance(s) matching the filter.<br>If absent, the complete information is returned for the VNF instance(s) matching the filter. |

### 7.2.9.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.9.3-1.

**Table 7.2.9.3-1: Query VNF operation output parameters**

| Parameter   | Qualifier | Cardinality | Content | Description   |
|---|-----------|-------------|---------|---|
| vnfInfo   | M         | 0..N        | VnfInfo | The information items about the selected VNF instance(s) that are returned.<br>If attributeSelector is present, only the attributes listed in attributeSelector are returned for the selected VNF instance(s).<br>See note. |
| <b>NOTE:</b> The lower cardinality is 0 since there may be no matches to the provided filter. |           |             |         |   |

### 7.2.9.4 Operation results

In case of success, information related to the VNF instances that match the filter is returned. In case of failure, appropriate error information is returned.

## 7.2.10 Heal VNF operation

### 7.2.10.1 Description

This operation enables the NFVO to request a VNFM to perform a VNF healing procedure.

It depends on the VNF capabilities, and is declared in the VNFD, whether this operation is supported for a particular VNF.

Table 7.2.10.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.10.1-1: Heal VNF operation**

| Message         | Requirement | Direction   |
|-----------------|-------------|-------------|
| HealVnfRequest  | Mandatory   | NFVO → VNFM |
| HealVnfResponse | Mandatory   | VNFM → NFVO |

## 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: Heal VNF operation input parameters**

| Parameter       | Qualifier | Cardinality | Content      | Description  |
|-----------------|-----------|-------------|--------------|--|
| vnfInstanceId   | M         | 1           | Identifier   | Identifies the VNF instance 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 NFVO 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 [3]).<br>EXAMPLE: Input parameters to VNF-specific healing procedures. |

## 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: Heal VNF operation output parameters**

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

## 7.2.10.4 Operation results

In case of success, the VNF has been healed, and the associated instance of a VnfInfo information element has been updated. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

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

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

## 7.2.11 Operate VNF operation

### 7.2.11.1 Description

This operation enables requesting to change the state of a VNF instance, including starting and stopping the VNF instance.

NOTE 1: These operations are complementary to instantiating and terminating a VNF.

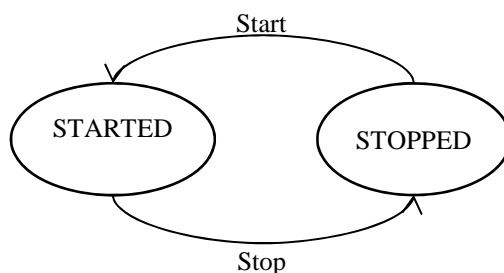
NOTE 2: In the present document, only starting and stopping the VNF instance(s) are supported. Extension of this operation to support other VNF state changes is part of the protocol design.

A VNF instance can be in the following states:

- **STARTED:** the VNF instance is up and running.
- **STOPPED:** the VNF instance has been shut down. A VNF instance is stopped if all its VNFC instances are also stopped.

In the state STOPPED, the virtualised container(s), where the VNFC instance(s) of the VNF run, are shut down but not terminated. In addition, if the workflow requires a graceful stop, as part of this process the VNFM (producer of the interface) will interact with VNF/EM to gracefully stop the VNF application. Once a VNF is instantiated, i.e. all instantiation steps have been completed, the VNF instance is in the state STARTED.

Figure 7.2.11.1-1 illustrates the VNF operate state diagram. The desired change of state is indicated as an input in the OperateVnfRequest operation.



**Figure 7.2.11.1-1: Operate VNF state diagram**

It depends on the VNF capabilities, and is declared in the VNFD, whether this operation is supported for a particular VNF.

Table 7.2.11.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.11.1-1: Operate VNF operation**

| Message            | Requirement | Direction   |
|--------------------|-------------|-------------|
| OperateVnfRequest  | Mandatory   | NFVO → VNFM |
| OperateVnfResponse | Mandatory   | VNFM → NFVO |

## 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: Operate VNF operation input parameters

| Parameter  | 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.<br>VALUES:<br><ul style="list-style-type: none"> <li>START</li> <li>STOP</li> </ul>  |
| stopType   | M         | 0..1        | Enum         | It signals whether forceful or graceful stop is requested.<br>VALUES:<br><ul style="list-style-type: none"> <li>FORCEFUL</li> <li>GRACEFUL</li> </ul> <p>In case of forceful stop, the VNF is stopped immediately. Note that if the VNF is still in service, this may adversely impact network service, and therefore, operator policies apply to determine if forceful stop is allowed in the particular situation.</p> <p>In case of graceful stop, 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.</p> <p>Only applicable when changing state to stop.</p> |
| 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.<br><br>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).<br><br>Minimum timeout or timeout range are specified by the VNF vendor (e.g. defined in the VNFD or communicated by other means).<br><br>The parameter is not relevant in case of forceful stop.  |
| additionalParam  | M         | 0..N        | KeyValuePair | Additional parameters passed by the NFVO 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 [3]).  |
| NOTE: This implies that no VNF stop will be attempted if taking the VNF out of service fails or hangs. |           |             |              |  |

### 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: Operate VNF operation output parameters

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

### 7.2.11.4 Operation results

In case of success, the state of the VNF has been changed, and the associated instance of a VnfInfo information element has been updated. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

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

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

## 7.2.12 Modify VNF Information operation

### 7.2.12.1 Description

This operation allows updating information about a VNF instance.

This operation shall be supported for all VNFs.

Table 7.2.12.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.12.1-1: Modify VNF Information operation**

| Message               | Requirement | Direction   |
|-----------------------|-------------|-------------|
| ModifyVnfInfoRequest  | Mandatory   | NFVO → VNFM |
| ModifyVnfInfoResponse | Mandatory   | VNFM → NFVO |

### 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: Modify VNF Information operation input parameters**

| Parameter     | Qualifier | Cardinality | Content      | Description  |
|---------------|-----------|-------------|--------------|--|
| vnfInstanceId | M         | 1           | Identifier   | Identifier of the VNF instance for which the writeable 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. |

### 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: Modify VNF Information operation output parameters**

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

### 7.2.12.4 Operation results

In case of success:

- if the operation handles changes to the VNF configurable properties, the configuration in the VNF has been modified according to the input parameters specified in the operation;
- if the operation handles other changes to the VNF instance information, the VNF information has been changed according to the input parameters specified in the operation.

In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification. In particular, error information shall indicate the reason why the requested attribute has not been updated, e.g. changing the value of the attribute is not supported, input attribute name is not recognized, etc.

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

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

## 7.2.13 Get Operation Status operation

### 7.2.13.1 Description

This operation provides the status of a VNF lifecycle management operation. This means, it is not a VNF lifecycle management operation itself, but an operation on VNF lifecycle management operations. Therefore, this operation shall be supported for all VNFs.

Table 7.2.13.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.13.1-1: Get Operation Status operation**

| Message                    | Requirement | Direction   |
|----------------------------|-------------|-------------|
| GetOperationStatusRequest  | Mandatory   | NFVO → VNFM |
| GetOperationStatusResponse | Mandatory   | VNFM → NFVO |

### 7.2.13.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.13.2-1.

**Table 7.2.13.2-1: Get Operation Status operation input parameters**

| Parameter                      | Qualifier | Cardinality | Content    | Description   |
|--------------------------------|-----------|-------------|------------|---|
| lifecycleOperationOccurrenceId | M         | 1           | Identifier | Identifier of the VNF lifecycle operation occurrence. |

### 7.2.13.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.13.3-1.

**Table 7.2.13.3-1: Get Operation Status operation output parameters**

| Parameter       | Qualifier | Cardinality | Content | Description  |
|-----------------|-----------|-------------|---------|--|
| operationStatus | M         | 1           | Enum    | Indicates the operation status (which includes, for example: Processing, Successfully done, Failed, but can also include operation-specific states).<br>VALUES:<br><ul style="list-style-type: none"> <li>• STARTING</li> <li>• PROCESSING</li> <li>• COMPLETED</li> <li>• FAILED_TEMP</li> <li>• FAILED</li> <li>• ROLLING_BACK</li> <li>• ROLLED_BACK</li> <li>• Etc.</li> </ul> |

### 7.2.13.4 Operation results

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

## 7.2.14 Subscribe operation

### 7.2.14.1 Description

This operation enables the NFVO to subscribe with a filter for the notifications sent by the VNFM which are related to VNF lifecycle management operation occurrences, as well as creation/deletion of VNF instance identifiers and the associated VnfInfo information element instances.

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

Table 7.2.14.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.14.1-1: Subscribe operation**

| Message           | Requirement | Direction   |
|-------------------|-------------|-------------|
| SubscribeRequest  | Mandatory   | NFVO → VNFM |
| SubscribeResponse | Mandatory   | VNFM → NFVO |

### 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: Subscribe operation input parameters**

| Parameter  | Qualifier | Cardinality | Content | Description  |
|--|-----------|-------------|---------|--|
| filter   | M         | 1           | Filter  | Input filter for selecting e.g. the VNF instances of interest and the specific types of changes. See note. |
| NOTE: When subscribing for notifications regarding the creation of VNF identifiers and the associated VNF information object instances, selecting the VNF instances in the filter is not possible. |           |             |         |  |

### 7.2.14.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.14.3-1.

**Table 7.2.14.3-1: Subscribe operation output parameters**

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

### 7.2.14.4 Operation results

After successful subscription, the consumer (NFVO) is registered to receive notifications related to VNF lifecycle management operation occurrences, as well as creation/deletion of VNF instance identifiers and the associated VnfInfo 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.

## 7.2.15 Notify operation

### 7.2.15.1 Description

This operation notifies a subscriber about events related to VNF lifecycle operation occurrences, as well as creation/deletion of VNF instance identifiers and the associated VnfInfo information element instances.

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

Table 7.2.15.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.15.1-1: Notify operation**

| Message | Requirement | Direction   |
|---------|-------------|-------------|
| Notify  | Mandatory   | VNFM → NFVO |

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

- VnfLcmOperationOccurrenceNotification (see clause 8.6.2).
- VnfIdentifierCreationNotification (see clause 8.6.7).
- VnfIdentifierDeletionNotification (see clause 8.6.8).

## 7.2.16 Terminate Subscription operation

### 7.2.16.1 Description

This operation enables the NFVO to terminate a particular subscription.

Table 7.2.16.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.2.16.1-1: Terminate Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| TerminateSubscriptionRequest  | Mandatory   | NFVO → VNFM |
| TerminateSubscriptionResponse | Mandatory   | VNFM → NFVO |

### 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: Terminate Subscription operation input parameters**

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



### 7.2.16.3 Output parameters

None.

### 7.2.16.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the NFVO 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.

## 7.2.17 Query Subscription Info operation

### 7.2.17.1 Description

This operation enables the NFVO to query information about subscriptions.

Table 7.2.17.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.2.17.1-1: Query Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| QuerySubscriptionInfoRequest  | Mandatory   | NFVO → VNFM |
| QuerySubscriptionInfoResponse | Mandatory   | VNFM → NFVO |

### 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: 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. |

### 7.2.17.3 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: 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. |

### 7.2.17.4 Operation results

After successful operation, the VNFM 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 lifecycle management that the NFVO has access to and that are matching the filter shall be returned.

## 7.2.18 Change External VNF Connectivity operation

### 7.2.18.1 Description

This operation enables changing the external connectivity of a VNF instance. 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 [3]. 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 [3], and connect them to a particular external VL. Creation of the parent port with this operation is not supported.

VNFs shall support this operation. This operation may be service-disruptive.

Table 7.2.18.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.2.18.1-1: Change External VNF Connectivity**

| Message                          | Requirement | Direction   |
|----------------------------------|-------------|-------------|
| ChangeExtVnfConnectivityRequest  | Mandatory   | NFVO → VNFM |
| ChangeExtVnfConnectivityResponse | Mandatory   | VNFM → NFVO |

### 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. The parameters passed for this operation override those passed at instantiation time.

**Table 7.2.18.2-1: Change External VNF Connectivity operation input parameters**

| Parameter       | 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 NFVO as input to the Change External VNF Connectivity operation, specific to the VNF of which the external VLs are being changed as declared in the VNFD (see clause 7.1.5.10 in ETSI GS NFV-IFA 011 [3]). |

| Parameter         | Qualifier | Cardinality | Content           | Description  |
|-------------------|-----------|-------------|-------------------|--|
| vimConnectionInfo | CM        | 0..N        | VimConnectionInfo | Information about VIM connection(s) for managing resources for the VNF instance, or external virtual links.<br><br>This attribute shall be supported if VNF-related resource management in direct mode is applicable. In that case, this attribute shall be present if there is the need to communicate VIM connection information for external virtual links. |

The following behaviour applies for the changes that can be performed with this operation:

- To change the connection of external CP instances based on certain external CPDs from a "source" external VL to a different "target" external VL, the identifier of the "target" external VL shall be sent in the "extVirtualLinkId" attribute of the "extVirtualLink" parameter, and the "extCp" attributes of that parameter shall refer via the "cpdId" attribute to the external CPDs of the corresponding external connection point instances that are to be reconnected to the target external VL.

NOTE: For CP instances that are not part of a trunk, this means that all CP instances based on a given external CPD will be reconnected. See clause A.3 in annex A for an illustration. For CP instances that are part of a trunk the change of connectivity can be requested individually per CP instance.

- To change the connectivity parameters of the external CPs connected to a particular external VL, including changing addresses, the identifier of that external VL shall be sent in the "extVirtualLinkId" attribute of the "extVirtualLink" parameter, and the "extCp" attribute of that parameter shall contain at least those entries with modified parameters.

### 7.2.18.3 Output parameters

None.

### 7.2.18.4 Operation results

In the case of success, the connectivity of the VNF has been changed according to the input parameters, and the associated instance of a VnfInfo information element has been updated. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

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

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

## 7.2.19 Query Snapshot Information operation

### 7.2.19.1 Description

This operation enables the NFVO to query the VNFM for information it has stored about one or more VNF Snapshots. Table 7.2.19.1-1 lists the information flow exchanged between the VNFM and the NFVO.

The operation allows querying specific components of the information stored in the VNFM about a VNF Snapshot, for instance, retrieving the vnfSnapshotInfoId.

This operation shall be supported by the VNF if the Create Snapshot Operation is supported.

**Table 7.2.19.1-1: Query Snapshot Information operation**

| Message                   | Requirement | Direction   |
|---------------------------|-------------|-------------|
| QuerySnapshotInfoRequest  | Mandatory   | NFVO → VNFM |
| QuerySnapshotInfoResponse | Mandatory   | VNFM → NFVO |

### 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: Query Snapshot Information operation input parameters**

| Parameter   | Qualifier | Cardinality | Content | Description   |
|---|-----------|-------------|---------|---|
| filter  | M         | 1           | Filter  | Filter defining the VNF Snapshot on which the query applies, based on attributes of the VnfSnapshotInfo.<br>It can also be used to specify one or more VNF Snapshots to be queried by providing their vnfSnapshotInfoId or vnfInstanceId. See note. |
| attributeSelector   | M         | 0..N        | String  | It provides a list of attribute names of VnfSnapshotInfo. If present, only these attributes are returned for the VnfSnapshotInfo matching the filter. If absent, the complete VnfSnapshotInfo is returned.  |
| NOTE: The vnfSnapshotInfoId, assigned at VNF Snapshot creation or at VNF Snapshot Package extraction, identifies the information related to a VNF Snapshot. |           |             |         |   |

### 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: Query Snapshot Information operation output parameters**

| Parameter       | Qualifier | Cardinality | Content         | Description   |
|-----------------|-----------|-------------|-----------------|---|
| vnfSnapshotInfo | M         | 0..N        | VnfSnapshotInfo | Details of the VNF Snapshots available to the NFVO matching the input filter. If attributeSelector is present, only the attributes listed in attributeSelector are returned for the selected entities. The lower cardinality is 0 since there may be no matches to the provided filter. |

### 7.2.19.4 Operation results

After successful operation, the VNFM has queried the internal VNF Snapshot 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 that the consumer has access to and that are matching the filter shall be returned.

## 7.2.20 Create Snapshot operation

### 7.2.20.1 Description

This operation enables the NFVO to request the creation of a VNF Snapshot. The VNF instance to be snapshotted is addressed using an identifier held by the VNFM about a specific VNF instance.

It depends on the VNF capabilities, and is declared in the VNFD (refer to the "supportedOperation" attribute in the VnfDf information element; see clause 7.1.8.2 in ETSI GS NFV-IFA 011 [3]), whether this operation is supported for a particular VNF.

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

**Table 7.2.20.1-1: Create Snapshot operation**

| Message                | Requirement | Direction   |
|------------------------|-------------|-------------|
| CreateSnapshotRequest  | Mandatory   | NFVO → VNFM |
| CreateSnapshotResponse | Mandatory   | VNFM → NFVO |

### 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: Create Snapshot operation input parameters**

| Parameter       | Qualifier | Cardinality | Content      | Description  |
|-----------------|-----------|-------------|--------------|--|
| vnfInstanceId   | M         | 1           | Identifier   | Identifier of the VNF instance to be snapshotted.  |
| additionalParam | M         | 0..N        | KeyValuePair | Additional parameters passed by the NFVO as input to the 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 [3]). |
| userDefinedData | O         | 0..N        | KeyValuePair | User defined data for the VNF Snapshot.  |

### 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: Create Snapshot operation output parameters**

| Parameter       | Qualifier | Cardinality | Content                                      | Description   |
|-----------------|-----------|-------------|--|---|
| vnfSnapshotInfo | M         | 1           | Identifier<br>(Reference to VnfSnapshotInfo) | Identifier of information held by the VNFM about the specific VNF Snapshot. This identifier was allocated by the VNFM if the create operation was successful. |

### 7.2.20.4 Operation results

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

Before the VNF Snapshot creation starts and ends, the VNF/EM have been notified about the operation via VnfLcmOperationOccurrenceNotification in case they have subscribed for this type of notifications.

After successful operation, the VNFM has created a VNF Snapshot from the specified VNF instance, including information associated with this VNF Snapshot. Once created, the VNF Snapshot is known to the VNFM. It is enabled to be queried for its associated information, it is enabled to be reverted to, and it is enabled to create a VNF Snapshot Package from it.

## 7.2.21 Revert-to Snapshot operation

### 7.2.21.1 Description

This operation enables the NFVO to request the reversion of a VNF instance to a VNF Snapshot. The VNF Snapshot to be reverted to is addressed using an identifier held by the VNFM about a specific VNF Snapshot.

It depends on the VNF capabilities, and is declared in the VNFD (refer to the "supportedOperation" attribute in the VnfDf information element; see clause 7.1.8.2 in ETSI GS NFV-IFA 011 [3]), whether this operation is supported for a particular VNF. This operation may be service-disruptive.

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

**Table 7.2.21.1-1: Revert-to Snapshot operation**

| Message                  | Requirement | Direction   |
|--------------------------|-------------|-------------|
| RevertToSnapshotRequest  | Mandatory   | NFVO → VNFM |
| RevertToSnapshotResponse | Mandatory   | VNFM → NFVO |

### 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: Revert-to Snapshot operation input parameters**

| Parameter         | Qualifier | Cardinality | Content                                      | Description   |
|-------------------|-----------|-------------|--|---|
| vnfInstancelid    | M         | 1           | Identifier                                   | Identifier of the VNF instance to be reverted.  |
| vnfSnapshotInfold | M         | 1           | Identifier<br>(Reference to VnfSnapshotInfo) | 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 NFVO as input to the snapshot reversion process, specific for the VNF instance being "reverted" to snapshot as declared in the VNFD (see clause 7.1.5.12 in ETSI GS NFV-IFA 011 [3]). |

### 7.2.21.3 Output parameters

No output parameter.

### 7.2.21.4 Operation results

The result of the operation indicates if the reversion to the VNF Snapshot has been successful or not with a standard success/error result.

Before the VNF Snapshot reversion starts and ends, the VNF/EM have been notified about the operation via VnfLcmOperationOccurrenceNotification in case they have subscribed for this type of notifications.

## 7.2.22 Delete Snapshot Information operation

### 7.2.22.1 Description

This operation enables the NFVO to request the deletion of the held information associated to a VNF Snapshot. The VNF Snapshot information to be deleted is addressed using an identifier of information held by the VNFM about a specific VNF Snapshot.

It depends on the VNF capabilities, and is declared in the VNFD (refer to the "supportedOperation" attribute in the VnfDf information element; see clause 7.1.8.2 in ETSI GS NFV-IFA 011 [3]), whether this operation is supported for a particular VNF.

Table 7.2.22.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.2.22.1-1: Delete Snapshot Information operation**

| Message                    | Requirement | Direction   |
|----------------------------|-------------|-------------|
| DeleteSnapshotInfoRequest  | Mandatory   | NFVO → VNFM |
| DeleteSnapshotInfoResponse | Mandatory   | VNFM → NFVO |

### 7.2.22.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.22.2-1.

**Table 7.2.22.2-1: Delete Snapshot Information operation input parameters**

| Parameter         | Qualifier | Cardinality | Content   | Description  |
|-------------------|-----------|-------------|---|--|
| vnfSnapshotInfoId | M         | 1           | Identifier<br>(Reference to<br>VnfSnapshotInfo) | Identifier of information held by the VNFM about a specific VNF Snapshot. This identifier was allocated by the VNFM. |

### 7.2.22.3 Output parameters

No output parameter.

### 7.2.22.4 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 VNFM has deleted the held information associated to the specified VNF Snapshot. After deletion of the held information associated to a VNF Snapshot, this VNF Snapshot is not known any longer to the VNFM and it is not possible to revert to this VNF Snapshot or to create a VNF Snapshot Package from it.

## 7.2.23 Change current VNF package operation

### 7.2.23.1 Description

This operation enables the NFVO to request the VNFM to change the current VNF Package, i.e. the VNF package on which a VNF instance is based.

Refer to ETSI GS NFV-REL 006 [i.10] for more information related to the types of changes to the current VNF Package (aka "VNF software modification"). Clause B.3 of the present document illustrates the variants of changes to the current VNF Package and information flow procedures.

This operation encompasses the following scenarios:

- Changes of the VNF virtualised resources, such as requirements, composition and structure between the VNF versions, without changing the VNF software version.
- Changes of both the VNF software version and the VNF virtualised resources. This case includes replacing the VNF software version by means of virtualised resources management, such as terminating the current virtualised resource instances running the current software version and instantiating new virtualised resource instances with the destination VNF software version. The new virtualised resource instances may have the same characteristics as the current virtualised resource instances.
- Changes related to the VNFD, such as correction of bugs in the VNFD, changes in the naming scheme of VNFD components (e.g. name of the VDU, vduId), and adding/removing descriptors of VNF Package changes (VnfPackageChangeInfo).

**NOTE:** For software updates that are executed by functional entities outside NFV-MANO and that require synchronization of the information held by the NFV-MANO entities with a new VNF package that reflects the same changes, a separate procedure using the Modify VNF Information operation has been defined, as illustrated in clause B.2. This procedure assumes certain restrictions on the characteristics of the new VNF package, as defined in note 4 in table 8.5.2.2-1.

As part of changing the current VNF Package, the VNFM shall be capable to add temporary virtualised resources used in the modification process, e.g. virtualised resources for a VNFC which will be responsible for handling or supporting the change of the current VNF Package process. The need for temporary virtualised resources shall be indicated as "tempResource" to the NFVO during the VNF LCM operation granting exchange. In addition, the VNFM shall be capable to add and remove virtualised resources as required for the "change of current VNF Package" process. The need for addition and removal of existing virtualised resources shall be indicated as "addResource" and "removeResource" in the VNF LCM operation granting exchange.

All VNFs shall support this operation. This operation may be service-disruptive. It is declared in the VNFD (refer to the "selector" attribute in the " VnfPackageChangeInfo" information element; see clause 7.1.15.2.2 in ETSI GS NFV-IFA 011 [3]), whether a change from a particular "source" VNF package to a particular "destination" VNF package is possible.

Table 7.2.23.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.2.23.1-1: Change current VNF Package operation**

| Message                         | Requirement | Direction   |
|---------------------------------|-------------|-------------|
| ChangeCurrentVnfPackageRequest  | Mandatory   | NFVO → VNFM |
| ChangeCurrentVnfPackageResponse | Mandatory   | VNFM → NFVO |

### 7.2.23.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.23.2-1.

**Table 7.2.23.2-1: Change current VNF Package operation input parameters**

| Parameter             | Qualifier | Cardinality | Content                   | Description  |
|-----------------------|-----------|-------------|---------------------------|--|
| vnfInstanceId         | M         | 1           | Identifier                | Identifier of the VNF instance to which this change request is related.  |
| vnfdId                | M         | 1           | Identifier                | Identifier of the VNFD which defines the destination VNF Package for the change.   |
| extVirtualLink        | M         | 0..N        | ExtVirtualLinkData        | Information about external VLS.  |
| extManagedVirtualLink | M         | 0..N        | ExtManagedVirtualLinkData | Information about internal VLS that are managed by other entities than the VNFM (see notes 1 and 2).   |
| vimConnectionInfo     | CM        | 0..N        | VimConnectionInfo         | Information about VIM connection(s) for managing resources for the VNF instance, or external/externally-managed virtual links.<br>This attribute shall be supported if VNF-related resource management in direct mode is applicable. In that case, this attribute shall be present if there is the need to communicate modified VIM connection information for external or externally-managed virtual links. |
| additionalParam       | M         | 0..N        | KeyValuePair              | Additional parameters passed by the NFVO 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 [3]).  |
| extension             | M         | 0..N        | KeyValuePair              | This parameter provides changes to the values for the "extension" attribute in "VnfInfo", including new values for extensions that are declared in the VNFDs of both the source and the destination VNF Packages (see clause 7.1.14.2 in ETSI GS NFV-IFA 011 [3]) and values for new extensions declared in the VNFD of the destination VNF Package.   |



| Parameter   | Qualifier | Cardinality | Content      | Description  |
|---|-----------|-------------|--------------|--|
| vnfConfigurableProperties   | M         | 0..N        | KeyValuePair | This parameter provides changes to the values for the "vnfConfigurableProperties" attribute in "VnfInfo", including new values for configurable properties that are declared in the VNFDs of both the source and the destination VNF Packages (see clause 7.1.12.2 in ETSI GS NFV-IFA 011 [3]) and values for new configurable properties declared in the VNFD of the destination VNF Package. |
| 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 8.12.4.2).   |           |             |              |  |

### 7.2.23.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.23.3-1.

**Table 7.2.23.3-1: Change current VNF Package operation output parameters**

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

### 7.2.23.4 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 requested change of the current VNF Package has been completed on the VNF instance, and the associated instance of the VnfInfo information element has been updated. In case of failure, appropriate error information is provided in the "result" LCM Operation Occurrence Notification.

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

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

## 7.2.24 Fetch VNF state snapshot

### 7.2.24.1 Description

As part of a VNF snapshot creation, VNF-specific state data associated to the VNF snapshot might be created by the VNFM. Such data can be used during VNF snapshot reversion, root cause analysis, etc. and might need to be also compiled by the NFVO into a VNF snapshot package.

This operation enables the NFVO to fetch the content of a VNF state snapshot from the VNFM.

Table 7.2.24.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.2.24.1-1: Fetch VNF state snapshot operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| FetchVnfStateSnapshotRequest  | Mandatory   | NFVO → VNFM |
| FetchVnfStateSnapshotResponse | Mandatory   | VNFM → NFVO |

### 7.2.24.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.2.24.2-1.

**Table 7.2.24.2-1: Fetch VNF state snapshot operation input parameters**

| Parameter       | Qualifier | Cardinality | Content                                   | Description  |
|-----------------|-----------|-------------|---|--|
| vnfSnapshotInfo | M         | 1           | Identifier (Reference to VnfSnapshotInfo) | References information held by the VNFM about a specific VNF Snapshot. |

### 7.2.24.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.2.24.3-1.

**Table 7.2.24.3-1: Fetch VNF state snapshot operation output parameters**

| Parameter        | Qualifier | Cardinality | Content       | Description         |
|------------------|-----------|-------------|---------------|---------------------|
| vnfStateSnapshot | M         | 1           | Not specified | VNF state snapshot. |

### 7.2.24.4 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 fetched the requested VNF state snapshot from the VNFM. In case of failure, appropriate error information is returned.

## 7.3 Void

## 7.4 VNF Performance Management interface

### 7.4.1 Description

This interface allows providing performance management (measurement results collection and notifications) related to VNFs. Performance information on a given VNF related measured object instance (see note 1) results from performance information of the virtualised resources that is collected from the VIM and mapped to this VNF related measured object instance.

NOTE 1: The VNF related measured object instance is the instance of one of the measured object type(s) for which the performance measurements applicable to Or-Vnfm reference point are defined in clause 7.2 of ETSI GS NFV-IFA 027 [5].

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

When new performance information is available, the consumer is notified using the notification PerformanceInformationAvailableNotification (see clause 8.7.8). The details of the performance measurements are provided using the PerformanceReport information element (see clause 8.7.5).

NOTE 2: 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 NFVO:

- 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.
- Terminate Subscription operation.
- Query Subscription Info operation.

## 7.4.2 Create PM Job operation

### 7.4.2.1 Description

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

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

Table 7.4.2.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.4.2.1-1: Create PM Job operation**

| Message             | Requirement | Direction   |
|---------------------|-------------|-------------|
| CreatePmJobRequest  | Mandatory   | NFVO → VNFM |
| CreatePmJobResponse | Mandatory   | VNFM → NFVO |

### 7.4.2.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.2.2-1.

**Table 7.4.2.2-1: Create PM Job operation input parameters**

| Parameter         | Qualifier | Cardinality | Content         | Description   |
|-------------------|-----------|-------------|-----------------|---|
| objectSelector    | M         | 1           | ObjectSelection | Defines the VNF related measured object(s) for which performance information is requested 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 group) shall be present. |

| Parameter              | Qualifier  | Cardinality | Content       | Description   |
|------------------------|--|-------------|---------------|---|
| 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.<br>At least one of the two attributes (performance metric or group) shall be present. |
| collectionPeriod       | M  | 1           | Not specified | Specifies the periodicity at which the VNFM will collect performance information (see note).  |
| reportingPeriod        | M  | 1           | Not specified | Specifies the periodicity at which the VNFM will report to the NFVO about performance information (see note).   |
| reportingBoundary      | O  | 0..1        | Not specified | Identifies a boundary after which the reporting will stop.<br>The boundary shall allow a single reporting as well as periodic reporting up to the boundary.   |
| NOTE:                  | At the end of each reportingPeriod, the VNFM will inform NFVO 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 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. |             |               |   |

### 7.4.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.2.3-1.

**Table 7.4.2.3-1: Create PM Job operation output parameters**

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

### 7.4.2.4 Operation results

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

The pmJobId is returned when the operations has been successful.

## 7.4.3 Delete PM Jobs operation

### 7.4.3.1 Description

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

Table 7.4.3.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.4.3.1-1: Delete PM Jobs operation**

| Message              | Requirement | Direction   |
|----------------------|-------------|-------------|
| DeletePmJobsRequest  | Mandatory   | NFVO → VNFM |
| DeletePmJobsResponse | Mandatory   | VNFM → NFVO |

### 7.4.3.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.3.2-1.

**Table 7.4.3.2-1: Delete PM Jobs operation input parameters**

| Parameter  | Qualifier | Cardinality | Content    | Description                               |
|--|-----------|-------------|------------|---|
| pmJobId  | M         | 1..N        | Identifier | Identifiers of the PM jobs to be deleted. |
| NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to delete multiple PM Jobs in one request, or as a series of requests that delete one PM Job at a time. |           |             |            |   |

### 7.4.3.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.3.3-1.

**Table 7.4.3.3-1: Delete PM Jobs operation output parameters**

| Parameter  | Qualifier | Cardinality | Content    | Description                                      |
|--|-----------|-------------|------------|--|
| deletedPmJobId   | M         | 1..N        | Identifier | Identifiers of the PM jobs successfully deleted. |
| NOTE: It is up to the protocol design stage to determine whether this operation will be modelled as a "bulk" operation that allows to delete multiple PM Jobs in one request, or as a series of requests that delete one PM Job at a time. |           |             |            |  |

### 7.4.3.4 Operation results

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

## 7.4.4 Subscribe operation

### 7.4.4.1 Description

This operation enables the NFVOs to subscribe with a filter for the notifications related to performance information with the VNFM.

NOTE 1: Specification of 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 7.4.4.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.4.4.1-1: Subscribe operation**

| Message           | Requirement | Direction   |
|-------------------|-------------|-------------|
| SubscribeRequest  | Mandatory   | NFVO → VNFM |
| SubscribeResponse | Mandatory   | VNFM → NFVO |

### 7.4.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.4.2-1.

**Table 7.4.4.2-1: Subscribe operation input parameters**

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

### 7.4.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.4.3-1.

**Table 7.4.4.3-1: Subscribe operation output parameters**

| Parameter      | Qualifier | Cardinality | Content    | Description                              |
|----------------|-----------|-------------|------------|--|
| subscriptionId | M         | 1           | Identifier | Identifier of the subscription returned. |

#### 7.4.4.4 Operation results

As a result of this operation, the VNFM shall indicate to the NFVO 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.

### 7.4.5 Notify operation

#### 7.4.5.1 Description

This operation distributes notifications to subscribers. It is a one-way operation issued by the VNFM that cannot be invoked as an operation by the consumer (NFVO). In order to receive notifications, the NFVO shall have a subscription.

Table 7.4.5.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.4.5.1-1: Notify operation**

| Message | Requirement | Direction   |
|---------|-------------|-------------|
| Notify  | Mandatory   | VNFM → NFVO |

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

- PerformanceInformationAvailableNotification (see clause 8.7.8).
- ThresholdCrossedNotification (see clause 8.7.9).

### 7.4.6 Query PM Job operation

#### 7.4.6.1 Description

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

This operation is not returning performance reports.

Table 7.4.6.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.4.6.1-1: Query PM Job operation**

| Message            | Requirement | Direction   |
|--------------------|-------------|-------------|
| QueryPmJobRequest  | Mandatory   | NFVO → VNFM |
| QueryPmJobResponse | Mandatory   | VNFM → NFVO |

#### 7.4.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.6.2-1.

**Table 7.4.6.2-1: 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. |

### 7.4.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.6.3-1.

**Table 7.4.6.3-1: Query PM Job operation output parameters**

| Parameter | Qualifier | Cardinality | Content | Description                                   |
|-----------|-----------|-------------|---------|---|
| pmJob     | M         | 0..N        | PmJob   | Details of PM jobs matching the input filter. |

### 7.4.6.4 Operation results

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

## 7.4.7 Create Threshold operation

### 7.4.7.1 Description

This operation will allow the NFVO to create a threshold to specify threshold levels on specified performance metric and VNF 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 7.4.7.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.4.7.1-1: Create Threshold operation**

| Message                 | Requirement | Direction   |
|-------------------------|-------------|-------------|
| CreateThresholdRequest  | Mandatory   | NFVO → VNFM |
| CreateThresholdResponse | Mandatory   | VNFM → NFVO |

### 7.4.7.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.7.2-1.

**Table 7.4.7.2-1: Create Threshold operation input parameters**

| Parameter         | Qualifier | Cardinality | Content         | Description   |
|-------------------|-----------|-------------|-----------------|---|
| objectSelector    | M         | 1           | ObjectSelection | Defines the VNF related measured object(s) 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.<br>VALUES:<br><ul style="list-style-type: none"> <li>• SIMPLE: Single-valued static threshold</li> <li>• Etc.</li> </ul> |
| thresholdDetails  | M         | 1           | Not specified   | Details of the threshold: value to be crossed, and direction in which it is crossed, details on the notification to be generated, etc.  |

### 7.4.7.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.7.3-1.

**Table 7.4.7.3-1: Create Threshold operation output parameters**

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

#### 7.4.7.4 Operation results

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

The thresholdId is returned when the operations has been successful.

### 7.4.8 Delete Thresholds operation

#### 7.4.8.1 Description

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

Table 7.4.8.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.4.8.1-1: Delete Thresholds operation**

| Message                  | Requirement | Direction   |
|--------------------------|-------------|-------------|
| DeleteThresholdsRequest  | Mandatory   | NFVO → VNFM |
| DeleteThresholdsResponse | Mandatory   | VNFM → NFVO |

#### 7.4.8.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.8.2-1.

**Table 7.4.8.2-1: Delete Thresholds operation input parameters**

| Parameter  | Qualifier | Cardinality | Content    | Description                                  |
|--|-----------|-------------|------------|--|
| thresholdId  | M         | 1..N        | Identifier | Identifiers of the thresholds to be deleted. |
| NOTE: It is up to the protocol design stage to determine whether this operation will be 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. |           |             |            |  |

#### 7.4.8.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.8.3-1.

**Table 7.4.8.3-1: Delete Thresholds operation output parameters**

| Parameter  | Qualifier | Cardinality | Content    | Description  |
|--|-----------|-------------|------------|--|
| deletedThresholdId   | M         | 1..N        | Identifier | Identifiers of the thresholds that have been deleted successfully. |
| NOTE: It is up to the protocol design stage to determine whether this operation will be 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. |           |             |            |  |

#### 7.4.8.4 Operation results

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



## 7.4.9 Query Threshold operation

### 7.4.9.1 Description

This operation will allow the NFVO to query the details of an existing threshold.

Table 7.4.9.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.4.9.1-1: Query Threshold operation**

| Message                | Requirement | Direction   |
|------------------------|-------------|-------------|
| QueryThresholdRequest  | Mandatory   | NFVO → VNFM |
| QueryThresholdResponse | Mandatory   | VNFM → NFVO |

### 7.4.9.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.9.2-1.

**Table 7.4.9.2-1: Query 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. |

### 7.4.9.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.9.3-1.

**Table 7.4.9.3-1: Query Threshold operation output parameters**

| Parameter | Qualifier | Cardinality | Content   | Description  |
|-----------|-----------|-------------|-----------|--|
| threshold | M         | 0..N        | Threshold | List of threshold details matching the input filter. |

### 7.4.9.4 Operation results

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

## 7.4.10 Terminate Subscription operation

### 7.4.10.1 Description

This operation enables the NFVO to terminate a particular subscription.

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

Table 7.4.10.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.4.10.1-1: Terminate Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| TerminateSubscriptionRequest  | Mandatory   | NFVO → VNFM |
| TerminateSubscriptionResponse | Mandatory   | VNFM → NFVO |

### 7.4.10.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.10.2-1.

**Table 7.4.10.2-1: Terminate Subscription operation input parameters**

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

### 7.4.10.3 Output parameters

None.

### 7.4.10.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the NFVO 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.

## 7.4.11 Query Subscription Info operation

### 7.4.11.1 Description

This operation enables the NFVO 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 PM jobs and Thresholds.

Table 7.4.11.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.4.11.1-1: Query Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| QuerySubscriptionInfoRequest  | Mandatory   | NFVO → VNFM |
| QuerySubscriptionInfoResponse | Mandatory   | VNFM → NFVO |

### 7.4.11.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.4.11.2-1.

**Table 7.4.11.2-1: 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. |

### 7.4.11.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.4.11.3-1.

**Table 7.4.11.3-1: 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. |

#### 7.4.11.4 Operation results

After successful operation, the VNFM 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 fault management that the NFVO has access to and that are matching the filter shall be returned.

## 7.5 VNF Fault Management interface

### 7.5.1 Description

This interface shall allow the VNFM to provide alarms related to the VNFs visible to the consumer.

Virtualised resource alarms collected by the VNFM will be filtered, correlated and modified by the VNFM and mapped to the corresponding VNF instance, resulting in alarms on the corresponding VNF.

The fault management interface shall support the following operations:

- Subscribe operation (Subscription of NFVOs with the VNFM for the notifications related to the alarms).
- Notify operation (Notifications of alarms or alarm state change from VNFM to NFVO).
- Get alarm list operation (Accessing active alarms by the NFVO).
- Acknowledge alarms operation (Acknowledging alarms by the NFVO).
- Terminate Subscription operation.
- Query Subscription Info operation.

### 7.5.2 Subscribe operation

#### 7.5.2.1 Description

This operation enables the NFVO to subscribe with a filter for the notifications related to VNF alarms sent by the VNFM.

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

Table 7.5.2.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.5.2.1-1: Subscribe operation**

| Message           | Requirement | Direction   |
|-------------------|-------------|-------------|
| SubscribeRequest  | Mandatory   | NFVO → VNFM |
| SubscribeResponse | Mandatory   | VNFM → NFVO |

#### 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: Subscribe operation input parameters**

| Parameter | Qualifier | Cardinality | Content | Description   |
|-----------|-----------|-------------|---------|---|
| filter    | M         | 1           | Filter  | Input filter for selecting VNFs and related alarm notifications.<br><br>This filter can contain information to select VNF instances, information to select notification types as defined in clause 7.5.3.1, and additional filter criteria on further attributes of the Alarm information element defined in clause 8.8.4 as chosen in the protocol design stage. |

### 7.5.2.3 Output parameters

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

**Table 7.5.2.3-1: Subscribe operation output parameters**

| Parameter      | Qualifier | Cardinality | Content    | Description                              |
|----------------|-----------|-------------|------------|--|
| subscriptionId | M         | 1           | Identifier | Identifier of the subscription returned. |

### 7.5.2.4 Operation results

As a result of this operation, the VNFM shall indicate to the NFVO 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.

## 7.5.3 Notify operation

### 7.5.3.1 Description

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

In order to receive notifications, the NFVO shall have a subscription.

Table 7.5.3.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.5.3.1-1: Notify operation**

| Message | Requirement | Direction   |
|---------|-------------|-------------|
| Notify  | Mandatory   | VNFM → NFVO |

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

- AlarmNotification (see clause 8.8.2).
- AlarmClearedNotification (see clause 8.8.3).
- AlarmListRebuiltNotification (see clause 8.8.6).

## 7.5.4 Get Alarm List operation

### 7.5.4.1 Description

This operation enables the NFVOs to query the active alarms from the VNFM.

Table 7.5.4.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.5.4.1-1: Get Alarm List operation**

| Message              | Requirement | Direction   |
|----------------------|-------------|-------------|
| GetAlarmListRequest  | Mandatory   | NFVO → VNFM |
| GetAlarmListResponse | Mandatory   | VNFM → NFVO |

### 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: 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 VNF Identifiers, fault type, severity and cause. |

### 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: Get Alarm List operation output parameters**

| Parameter | Qualifier | Cardinality | Content | Description  |
|-----------|-----------|-------------|---------|--|
| alarm     | M         | 0..N        | Alarm   | Information about alarms including alarmId, affected VNF identifier, 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). |

### 7.5.4.4 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 are delivered to the NFVO.

## 7.5.5 Terminate Subscription operation

### 7.5.5.1 Description

This operation enables the NFVO to terminate a particular subscription.

Table 7.5.5.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.5.5.1-1: Terminate Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| TerminateSubscriptionRequest  | Mandatory   | NFVO → VNFM |
| TerminateSubscriptionResponse | Mandatory   | VNFM → NFVO |

### 7.5.5.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.5.5.2-1.

**Table 7.5.5.2-1: Terminate Subscription operation input parameters**

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

### 7.5.5.3 Output parameters

None.

### 7.5.5.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the NFVO 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.

## 7.5.6 Query Subscription Info operation

### 7.5.6.1 Description

This operation enables the NFVO to query information about subscriptions.

Table 7.5.6.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.5.6.1-1: Query Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| QuerySubscriptionInfoRequest  | Mandatory   | NFVO → VNFM |
| QuerySubscriptionInfoResponse | Mandatory   | VNFM → NFVO |

### 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 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. |

### 7.5.6.3 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 Subscription Info operation output parameters**

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

### 7.5.6.4 Operation results

After successful operation, the VNFM 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 fault management that the NFVO has access to and that are matching the filter shall be returned.

## 7.5.7 Acknowledge alarms operation

### 7.5.7.1 Description

This operation enables the NFVO to acknowledge alarms at VNFM.

Table 7.5.7.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.5.7.1-1: Acknowledge alarms operation**

| Message                   | Requirement | Direction   |
|---------------------------|-------------|-------------|
| AcknowledgeAlarmsRequest  | Mandatory   | NFVO → VNFM |
| AcknowledgeAlarmsResponse | Mandatory   | VNFM → NFVO |

### 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: 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 up to the protocol design stage to determine whether this operation will be 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. |           |             |                                 |   |

### 7.5.7.3 Output parameters

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

**Table 7.5.7.3-1: 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 up to the protocol design stage to determine whether this operation will be 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. |           |             |                                 |  |

### 7.5.7.4 Operation results

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

## 7.6 Void

## 7.7 VNF Indicator interface

### 7.7.1 Description

This interface allows the VNFM to provide information on value changes of VNF related indicators. VNF related indicators are declared in the VNFD. This interface is originally produced by the EM and/or VNF on the Ve-Vnfm-em and/or Ve-Vnfm-vnf reference point respectively (see ETSI GS NFV-IFA 008 [i.5]) and is re-exposed by the VNFM.

The following operations are defined for this interface:

- Subscribe.
- Notify.
- Get Indicator Value.
- Terminate Subscription.

- Query Subscription Info.

## 7.7.2 Subscribe operation

### 7.7.2.1 Description

This operation enables the NFVO to subscribe with a filter for the notifications related to VNF indicator value changes sent by the VNFM.

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

Table 7.7.2.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.7.2.1-1: Subscribe operation**

| Message           | Requirement | Direction   |
|-------------------|-------------|-------------|
| SubscribeRequest  | Mandatory   | NFVO → VNFM |
| SubscribeResponse | Mandatory   | VNFM → NFVO |

### 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: Subscribe operation input parameters**

| Parameter | Qualifier | Cardinality | Content | Description   |
|-----------|-----------|-------------|---------|---|
| filter    | M         | 1           | Filter  | Input filter for selecting VNFs and related indicators. |

### 7.7.2.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.7.2.3-1.

**Table 7.7.2.3-1: Subscribe operation input parameters**

| Parameter      | Qualifier | Cardinality | Content    | Description                              |
|----------------|-----------|-------------|------------|--|
| subscriptionId | M         | 1           | Identifier | Identifier of the subscription returned. |

### 7.7.2.4 Operation results

As a result of this operation, the VNFM shall indicate to the NFVO 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.

## 7.7.3 Notify operation

### 7.7.3.1 Description

This operation distributes notifications to subscribers. It is a one-way operation issued by the VNFM towards the NFVO that cannot be invoked as an operation by the consumer (NFVO). In order to receive notifications, the NFVO shall have a subscription.

Table 7.7.3.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.7.3.1-1: Notify operation**

| Message | Requirement | Direction   |
|---------|-------------|-------------|
| Notify  | Mandatory   | VNFM → NFVO |



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

- IndicatorValueChangeNotification (see clause 8.10.2).
- SupportedIndicatorsChangeNotification (see clause 8.10.4).

## 7.7.4 Get Indicator Value operation

### 7.7.4.1 Description

This operation enables NFVO to request from the VNFM information about available indicators and their actual values.

Table 7.7.4.1-1 lists the information flow exchanged between the VNFM and the NFVO.

**Table 7.7.4.1-1: Get Indicator Value operation**

| Message                   | Requirement | Direction   |
|---------------------------|-------------|-------------|
| GetIndicatorValueRequest  | Mandatory   | NFVO → VNFM |
| GetIndicatorValueResponse | Mandatory   | VNFM → NFVO |

### 7.7.4.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.7.4.2-1.

**Table 7.7.4.2-1: Get Indicator Value operation input parameters**

| Parameter | Qualifier | Cardinality | Content | Description   |
|-----------|-----------|-------------|---------|---|
| filter    | M         | 1           | Filter  | Input filter for selecting VNFs and related indicators. |

### 7.7.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.7.4.3-1.

**Table 7.7.4.3-1: Get Indicator Value operation output parameters**

| Parameter            | Qualifier | Cardinality | Content              | Description  |
|----------------------|-----------|-------------|----------------------|--|
| indicatorInformation | M         | 0..N        | IndicatorInformation | The requested indicator values as complex structures having the VNF Instance ID, Indicator and the value of the Indicator. |

### 7.7.4.4 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 indicators matching the filter will be delivered to the NFVO.

## 7.7.5 Terminate Subscription operation

### 7.7.5.1 Description

This operation enables the NFVO to terminate a particular subscription.

Table 7.7.5.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.7.5.1-1: Terminate Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| TerminateSubscriptionRequest  | Mandatory   | NFVO → VNFM |
| TerminateSubscriptionResponse | Mandatory   | VNFM → NFVO |

### 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: Terminate Subscription operation input parameters**

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

### 7.7.5.3 Output parameters

None.

### 7.7.5.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the NFVO 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.

## 7.7.6 Query Subscription Info operation

### 7.7.6.1 Description

This operation enables the NFVO to query information about subscriptions.

Table 7.7.6.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.7.6.1-1: Query Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| QuerySubscriptionInfoRequest  | Mandatory   | NFVO → VNFM |
| QuerySubscriptionInfoResponse | Mandatory   | VNFM → NFVO |

### 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 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. |

### 7.7.6.3 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 Subscription Info operation output parameters**

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

### 7.7.6.4 Operation results

After successful operation, the VNFM 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 indicator value changes that the NFVO has access to and that are matching the filter shall be returned.

## 7.8 Policy Management interface

### 7.8.1 Description

This interface allows the NFVO to invoke policy management operations towards the VNFM.

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 NFVO to manage subscriptions to notifications sent by the VNFM which inform about changes of a policy and about any detected policy conflicts. It allows the VNFM to provide such notifications to the subscriber (e.g. NFVO).

### 7.8.2 Transfer Policy operation

#### 7.8.2.1 Description

This operation enables the NFVO to transfer a NFV-MANO policy to the VNFM. Table 7.8.2.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.2.1-1: Transfer Policy operation**

| Message                | Requirement | Direction   |
|------------------------|-------------|-------------|
| TransferPolicyRequest  | Mandatory   | NFVO → VNFM |
| TransferPolicyResponse | Mandatory   | VNFM → NFVO |

#### 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: 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.                         |
| 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 NFVO 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. |           |             |               |  |

### 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: Transfer Policy operation output parameters**

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

### 7.8.2.4 Operation results

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

## 7.8.3 Delete Policy operation

### 7.8.3.1 Description

This operation enables the NFVO to delete one or multiple NFV-MANO policy(ies) from the VNFM. Table 7.8.3.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.3.1-1: Delete Policy operation**

| Message              | Requirement | Direction   |
|----------------------|-------------|-------------|
| DeletePolicyRequest  | Mandatory   | NFVO → VNFM |
| DeletePolicyResponse | Mandatory   | VNFM → NFVO |

### 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: 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 up to the protocol design stage to determine whether this operation will be 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. |           |             |                                     |                                      |

### 7.8.3.3 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: 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. |

### 7.8.3.4 Operation results

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

## 7.8.4 Query Policy operation

### 7.8.4.1 Description

This operation enables the NFVO to query the information from the VNFM on one or multiple NFV-MANO policy(ies). Table 7.8.4.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.4.1-1: Query Policy operation**

| Message             | Requirement | Direction   |
|---------------------|-------------|-------------|
| QueryPolicyRequest  | Mandatory   | NFVO → VNFM |
| QueryPolicyResponse | Mandatory   | VNFM → NFVO |

### 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: 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.                                 |

### 7.8.4.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.4.3-1.

**Table 7.8.4.3-1: Query Policy operation output parameters**

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

### 7.8.4.4 Operation results

After success operation, the VNFM 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.

## 7.8.5 Activate Policy operation

### 7.8.5.1 Description

This operation enables the NFVO to activate one or multiple NFV-MANO policy(ies) in the VNFM. Table 7.8.5.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.5.1-1: Activate Policy operation**

| Message                | Requirement | Direction   |
|------------------------|-------------|-------------|
| ActivatePolicyRequest  | Mandatory   | NFVO → VNFM |
| ActivatePolicyResponse | Mandatory   | VNFM → NFVO |

### 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: 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 up to the protocol design stage to determine whether this operation will be 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. |           |             |                                     |  |

### 7.8.5.3 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: 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). |

### 7.8.5.4 Operation results

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

## 7.8.6 Deactivate Policy operation

### 7.8.6.1 Description

This operation enables the NFVO to deactivate one or multiple NFV-MANO policy(ies) in the VNFM. Table 7.8.6.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.6.1-1: Deactivate Policy operation**

| Message                  | Requirement | Direction   |
|--------------------------|-------------|-------------|
| DeactivatePolicyRequest  | Mandatory   | NFVO → VNFM |
| DeactivatePolicyResponse | Mandatory   | VNFM → NFVO |

### 7.8.6.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.6.2-1.

**Table 7.8.6.2-1: 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. |           |             |                                     |  |

### 7.8.6.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.6.3-1.

**Table 7.8.6.3-1: 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). |

### 7.8.6.4 Operation results

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

## 7.8.7 Subscribe operation

### 7.8.7.1 Description

This operation enables the NFVO to subscribe with a filter for the notifications sent by the VNFM 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, associating policy and disassociating policy.

Table 7.8.7.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.7.1-1: Subscribe operation**

| Message           | Requirement | Direction   |
|-------------------|-------------|-------------|
| SubscribeRequest  | Mandatory   | NFVO → VNFM |
| SubscribeResponse | Mandatory   | VNFM → NFVO |

### 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: 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. |

### 7.8.7.3 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: Subscribe operation output parameters**

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

### 7.8.7.4 Operation results

After successful subscription, the consumer (NFVO) 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.

## 7.8.8 Notify operation

### 7.8.8.1 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 (VNFM) that cannot be invoked as an operation by the consumer (NFVO). In order to receive notifications, the consumer (NFVO) has to perform an explicit Subscribe operation beforehand.

Table 7.8.8.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.8.1-1: Notify operation**

| Message | Requirement | Direction   |
|---------|-------------|-------------|
| Notify  | Mandatory   | VNFM → NFVO |

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

- PolicyChangeNotification. See clause 8.13.3.
- PolicyConflictNotification. See clause 8.13.4.



## 7.8.9 Terminate Subscription operation

### 7.8.9.1 Description

This operation enables the NFVO to terminate a particular subscription.

Table 7.8.9.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.9.1-1: Terminate Subscription operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| TerminateSubscriptionRequest  | Mandatory   | NFVO → VNFM |
| TerminateSubscriptionResponse | Mandatory   | VNFM → NFVO |

### 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: Terminate Subscription operation input parameters**

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

### 7.8.9.3 Output parameters

None.

### 7.8.9.4 Operation results

After successful termination of a subscription, the identified subscription does not exist anymore, and the NFVO 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.

## 7.8.10 Query Subscription Info operation

### 7.8.10.1 Description

This operation enables the NFVO to query information about subscriptions.

Table 7.8.10.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.10.1-1: Query Subscription Info operation**

| Message                       | Requirement | Direction   |
|-------------------------------|-------------|-------------|
| QuerySubscriptionInfoRequest  | Mandatory   | NFVO → VNFM |
| QuerySubscriptionInfoResponse | Mandatory   | VNFM → NFVO |

### 7.8.10.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.10.2-1.

**Table 7.8.10.2-1: 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. |

### 7.8.10.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.10.3-1.

**Table 7.8.10.3-1: 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. |

### 7.8.10.4 Operation results

After successful operation, the VNFM 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 NFVO has access to and that are matching the filter shall be returned.

## 7.8.11 Associate Policy operation

### 7.8.11.1 Description

This operation enables the NFVO to associate a MANO policy to one or multiple VNF instances in the VNFM.

Table 7.8.11.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.11.1-1: Associate Policy operation**

| Message                 | Requirement | Direction   |
|-------------------------|-------------|-------------|
| AssociatePolicyRequest  | Mandatory   | NFVO → VNFM |
| AssociatePolicyResponse | Mandatory   | VNFM → NFVO |

### 7.8.11.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.11.2-1.

**Table 7.8.11.2-1: Associate Policy operation input parameters**

| Parameter  | Qualifier | Cardinality | Content                              | Description  |
|--|-----------|-------------|--------------------------------------|--|
| policyInfoId   | M         | 1           | Identifier (Reference to PolicyInfo) | Identifier of policy information.                                      |
| vnfInstanceld  | M         | 1..N        | Identifier                           | Identifier(s) of the VNF instance(s) to associate policy to. 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 associate a policy to multiple VNF instances in one request, or as a series of requests that associate the policy to one VNF instance at a time. |           |             |                                      |  |

### 7.8.11.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.11.3-1.

**Table 7.8.11.3-1: Associate Policy operation output parameters**

| Parameter     | Qualifier | Cardinality | Content    | Description   |
|---------------|-----------|-------------|------------|---|
| vnfInstanceld | M         | 0..N        | Identifier | Identifier(s) of the VNF instance(s) to which the policy has been associated. |

### 7.8.11.4 Operation results

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

## 7.8.12 Disassociate Policy operation

### 7.8.12.1 Description

This operation enables the NFVO to disassociate a MANO policy from one or multiple VNF instances in the VNFM.

Table 7.8.12.1-1 lists the information flow exchanged between the NFVO and the VNFM.

**Table 7.8.12.1-1: Disassociate Policy operation**

| Message                    | Requirement | Direction   |
|----------------------------|-------------|-------------|
| DisassociatePolicyRequest  | Mandatory   | NFVO → VNFM |
| DisassociatePolicyResponse | Mandatory   | VNFM → NFVO |

### 7.8.12.2 Input parameters

The input parameters sent when invoking the operation shall follow the indications provided in table 7.8.12.2-1.

**Table 7.8.12.2-1: Disassociate Policy operation input parameters**

| Parameter  | Qualifier | Cardinality | Content                                    | Description   |
|--|-----------|-------------|--|---|
| policyInfoId   | M         | 1           | Identifier<br>(Reference to<br>PolicyInfo) | Identifier of policy information.   |
| vnfInstanceId  | M         | 1..N        | Identifier                                 | Identifier(s) of the VNF instance(s) to disassociate policy from. 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 disassociate a policy from multiple VNF instances in one request, or as a series of requests that disassociate the policy from one VNF instance at a time. |           |             |  |   |

### 7.8.12.3 Output parameters

The output parameters returned by the operation shall follow the indications provided in table 7.8.12.3-1.

**Table 7.8.12.3-1: Disassociate Policy operation output parameters**

| Parameter     | Qualifier | Cardinality | Content    | Description  |
|---------------|-----------|-------------|------------|--|
| vnfInstanceId | M         | 0..N        | Identifier | Identifier(s) of the VNF instance(s) from which the policy has been disassociated. |

### 7.8.12.4 Operation results

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

## 7.9 Void

## 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.
- Version.
- KeyValuePair.

### 8.2 Information elements and notifications related to VNF Package Management

#### 8.2.1 Introduction

This clause defines information elements related to VNF Package Management.

#### 8.2.2 VnfPkgInfo information element

##### 8.2.2.1 Description

This information element provides the details of a VNF Package.

NOTE: The definition below is aligned with the definition of the VnfPkgInfo information element in ETSI GS NFV-IFA 013 [i.8].

##### 8.2.2.2 Attributes

The VnfPkgInfo information element shall follow the indications provided in table 8.2.2.2-1.

**Table 8.2.2.2-1: 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. |
| vnfId              | M         | 0..1        | Identifier | Identifier of the onboarded 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.                           |

| Attribute          | Qualifier | Cardinality | Content                        | Description   |
|--------------------|-----------|-------------|--------------------------------|---|
| 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 notes 2 and 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        | VnfPackageSoftwareImageInfo    | Information about VNF Package artifacts that are software images. See note 3.   |
| additionalArtifact | M         | 0..N        | VnfPackageArtifactInformation  | 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.<br>VALUES:<br><ul style="list-style-type: none"> <li>• CREATED</li> <li>• UPLOADING</li> <li>• PROCESSING</li> <li>• ONBOARDED</li> </ul> |
| operationalState   | M         | 1           | Enum                           | Operational state of the VNF Package.<br>VALUES:<br><ul style="list-style-type: none"> <li>• ENABLED</li> <li>• DISABLED</li> </ul>   |
| usageState         | M         | 1           | Enum                           | Usage state of the VNF Package.<br>VALUES:<br><ul style="list-style-type: none"> <li>• IN_USE</li> <li>• NOT_IN_USE</li> </ul>  |
| userDefinedData    | O         | 0..N        | KeyValuePair                   | User defined data for the VNF Package.  |

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 [3], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package.

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.

## 8.2.3 Vnfd information element

### 8.2.3.1 Description

This information element provides the details of the VNFD.

### 8.2.3.2 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 [3], clause 7.1.2.

## 8.2.4 VnfPackageOnBoardingNotification

### 8.2.4.1 Description

This notification indicates that 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.

### 8.2.4.2 Trigger Conditions

- New VNF Package on-boarded.

### 8.2.4.3 Attributes

The VnfPackageOnBoardingNotification shall follow the indications provided in table 8.2.4.3-1.

**Table 8.2.4.3-1: Attributes of the VnfPackageOnBoardingNotification**

| Attribute   | Qualifier | Cardinality | Content    | Description  |
|---|-----------|-------------|------------|--|
| onboardedVnfPkgInfold   | M         | 1           | Identifier | Identifier of the 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 [3], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package. |           |             |            |  |

## 8.2.5 VnfPackageChangeNotification

### 8.2.5.1 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.

### 8.2.5.2 Trigger Conditions

- Change of the operational state of an on-boarded VNF Package.
- Deletion of an on-boarded VNF Package.

### 8.2.5.3 Attributes

The VnfPackageChangeNotification shall follow the indications provided in table 8.2.5.3-1.

**Table 8.2.5.3-1: Attributes of the VnfPackageChangeNotification**

| Attribute   | Qualifier | Cardinality | Content    | Description  |
|---|-----------|-------------|------------|--|
| onboardedVnfPkgInfold   | 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.<br>VALUES: <ul style="list-style-type: none"> <li>• OP_STATE_CHANGE: change of operational state of an on-boarded VNF Package</li> <li>• PKG_DELETE: deletion of a VNF Package</li> </ul> |
| operationalState  | M         | 0..1        | Enum       | New operational state of the VNF Package.<br>VALUES: <ul style="list-style-type: none"> <li>• ENABLED</li> <li>• DISABLED</li> </ul>   |
| 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 [3], clause 7.1.2.2. This information is copied from the VNFD of the on-boarded VNF Package. |           |             |            |  |

## 8.2.6 Void

## 8.2.7 VnfPackageSoftwareImageInfo information element

### 8.2.7.1 Description

This information element represents Software Image Information.

### 8.2.7.2 Attributes

The VnfPackageSoftwareImageInfo information element shall follow the indications provided in table 8.2.7.2-1.

**Table 8.2.7.2-1: 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 time when this software image was created.   |
| 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. |

## 8.2.8 VnfPackageArtifactInformation information element

### 8.2.8.1 Description

This information element represents an artifact other than a Software Image which is contained in the VNF Package.

### 8.2.8.2 Attributes

The VnfPackageArtifactInformation information element shall follow the indications provided in table 8.2.8.2-1.

**Table 8.2.8.2-1: Attributes of the VnfPackageArtifactInformation 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 Package.                            |
| 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. |

## 8.2.9 Void

## 8.3 Information elements related to VNF Lifecycle Operation Granting

### 8.3.1 Introduction

This clause defines information elements related to VNF Lifecycle Operation Granting.

### 8.3.2 ResourceDefinition information element

#### 8.3.2.1 Description

This information element provides information of an existing or proposed resource used by the VNF.

#### 8.3.2.2 Attributes

The ResourceDefinition information element shall follow the indications provided in table 8.3.2.2-1.

**Table 8.3.2.2-1: Attributes of the ResourceDefinition information element**

| Attribute            | Qualifier | Cardinality | Content   | Description  |
|----------------------|-----------|-------------|---|--|
| resourceDefinitionId | M         | 1           | Identifier  | Identifier of this ResourceDefinition information element, unique at least within the scope of the grant request.  |
| type                 | M         | 1           | Enum  | Type of the resource definition referenced.<br>VALUES:<br><ul style="list-style-type: none"> <li>• COMPUTE</li> <li>• VL</li> <li>• LINKPORT</li> <li>• STORAGE</li> <li>• Etc.</li> </ul>   |
| vduld                | M         | 0..1        | Identifier (Reference to Vdu)   | Reference to the related Vdu applicable to this resource in the VNFD.<br>Shall only be present if a VDU is applicable to this resource in the VNFD.  |
| vnfdId               | M         | 0..1        | Identifier  | Identifier of the VNFD to which resourceTemplateId and vduld refer.<br>Shall be present if at least one of resourceTemplateId and vduld is present and the operation to be granted changes the current VNF Package. May be absent otherwise. |
| resourceTemplateId   | M         | 1           | Identifier (Reference to VnfVirtualLinkDesc, VirtualComputeDesc, VnfExtCpd or VirtualStorageDesc) | Reference to a resource template (VnfVirtualLinkDesc, VirtualComputeDesc, VnfExtCpd, VirtualStorageDesc) in the VNFD.  |
| resourceHandle       | M         | 0..1        | ResourceHandle  | Resource information for an existing resource. Shall be present for resources that are planned to be deleted or modified. Shall be absent otherwise.   |
| snapshotResDef       | M         | 0..1        | SnapshotResourceDefinition  | Information to identify a snapshot resource. Shall only be present if the operation to be granted concerns to creating a VNF snapshot from the VNF or to reverting the VNF to a VNF snapshot.  |



### 8.3.3 GrantInfo information element

#### 8.3.3.1 Description

This information element contains information about a Compute, storage or network resource whose addition/update/deletion was granted in a GrantVnfLifecycleOperationResponse.

#### 8.3.3.2 Attributes

The GrantInfo information element shall follow the indications provided in table 8.3.3.2-1.

**Table 8.3.3.2-1: Attributes of the GrantInfo information element**

| Attribute            | Qualifier | Cardinality | Content   | Description  |
|----------------------|-----------|-------------|---|--|
| resourceDefinitionId | M         | 1           | Identifier (Reference to ResourceDefinition)  | Identifier of the related ResourceDefinition information element from the grant request.   |
| reservationId        | M         | 0..1        | Identifier (Reference to ReservedVirtualCompute, ReservedVirtualNetwork, ReservedVirtualStorage, or ReservedComputeHosts) | The reservation identifier applicable to the VNFC/VirtualLink/VirtualStorage/ compute hosts. It shall be present for new resources when policy is GRANT_RESERVE and an applicable reservation exists; shall not be present otherwise.  |
| vimConnectionId      | CM        | 0..1        | Identifier (Reference to VimConnectionInfo)   | Reference to the identifier of the VimConnectionInfo information element defining the VIM connection to be used to manage this resource. Shall be present for new resources, and shall be absent for resources that have already been allocated.<br><br>This attribute shall be supported when VNF-related Resource Management in direct mode is applicable. |
| resourceProviderId   | CM        | 0..1        | Identifier  | Identifies the entity responsible for the management of the virtualised resource.<br><br>Shall be present for new resources, and shall be absent for resources that have already been allocated.<br><br>This attribute shall be supported when VNF-related Resource Management in indirect mode is applicable.   |
| zoneId               | M         | 0..1        | Identifier (Reference to ZoneInfo)  | Reference to the identifier of the ZoneInfo information element defining the resource zone into which this resource is to be placed.<br><br>Shall be present for new resources, and shall be absent for resources that have already been allocated.  |

| Attribute       | Qualifier | Cardinality | Content    | Description   |
|-----------------|-----------|-------------|------------|---|
| resourceGroupId | M         | 0..1        | Identifier | <p>Identifier of the "infrastructure resource group", logical grouping of virtual resources assigned to a tenant within an Infrastructure Domain, to be provided when allocating the resource.</p> <p>If the VIM connection referenced by "vimConnectionId" applies to multiple infrastructure resource groups, this attribute shall be present for new resources.</p> <p>If the VIM connection referenced by "vimConnectionId" applies to a single infrastructure resource group, this attribute may be present for new resources.</p> <p>This attribute shall be absent for resources that have already been allocated.</p> |

### 8.3.4 ZoneInfo information element

#### 8.3.4.1 Description

This information element provides information regarding a resource zone.

#### 8.3.4.2 Attributes

The ZoneInfo information element shall follow the indications provided in table 8.3.4.2-1.

**Table 8.3.4.2-1: Attributes of the ZoneInfo information element**

| Attribute          | Qualifier | Cardinality | Content                                     | Description  |
|--------------------|-----------|-------------|---|--|
| zoneInfoId         | M         | 1           | Identifier                                  | The identifier of this ZoneInfo instance, for the purpose of referencing it from other information elements.   |
| zoneId             | M         | 1           | Identifier                                  | The identifier of the resource zone, as managed by the resource management layer (typically, the VIM).   |
| vimConnectionId    | CM        | 1           | Identifier (Reference to VimConnectionInfo) | <p>The identifier of the connection to the VIM that manages the resource zone.</p> <p>This attribute shall be supported when VNF-related Resource Management in direct mode is applicable.</p> |
| resourceProviderId | CM        | 1           | Identifier                                  | <p>Identifies the entity responsible for the management the resource zone.</p> <p>This attribute shall be supported when VNF-related Resource Management in indirect mode is applicable.</p>   |

## 8.3.5 ZoneGroupInfo information element

### 8.3.5.1 Description

This information element provides information regarding a resource zone group. A resource zone group is a group of one or more related resource zones which can be used in resource placement constraints. To fulfil such constraint, the NFVO may decide to place a resource into any zone that belongs to a particular group.

**NOTE:** A resource zone group can be used to support overflow from one resource zone into another, in case a particular deployment supports only non-elastic resource zones.

### 8.3.5.2 Attributes

The ZoneGroupInfo information element shall follow the indications provided in table 8.3.5.2-1.

**Table 8.3.5.2-1: Attributes of the ZoneGroupInfo information element**

| Attribute | Qualifier | Cardinality | Content                            | Description   |
|-----------|-----------|-------------|------------------------------------|---|
| zonelid   | M         | 1..N        | Identifier (Reference to ZoneInfo) | References of identifiers of ZoneInfo instances, each of which provides information about a resource zone that belongs to this group. |

## 8.3.6 PlacementConstraint information element

### 8.3.6.1 Description

This information element provides information regarding a resource placement constraint. A set of such constraints may be sent by the VNFM to the NFVO to influence the resource placement decisions made by the NFVO as part of the granting process. A placement constraint defines a condition to the placement of new resources, considering other new resources as well as existing resources.

**EXAMPLE:** The following rules influence the placement of a set of resources such that they are placed in the same Network Function Virtualisation Infrastructure Point of Presence (NFVI-PoP) but in different resource zones:

```
{type="affinity"; scope="NFVI-PoP"; {resource1,resource2}}
{type="anti-affinity"; scope="Zone"; {resource1,resource2}}
```

### 8.3.6.2 Attributes

The PlacementConstraint information element shall follow the indications provided in table 8.3.6.2-1.

**Table 8.3.6.2-1: Attributes of the PlacementConstraint information element**

| Attribute              | Qualifier | Cardinality | Content               | Description  |
|------------------------|-----------|-------------|-----------------------|--|
| affinityOrAntiAffinity | M         | 1           | Enum                  | The type of the constraint.<br>VALUES:<br><ul style="list-style-type: none"> <li>AFFINITY</li> <li>ANTI_AFFINITY</li> </ul>  |
| scope                  | M         | 1           | Enum                  | The scope of the placement constraint indicating the category of the "place" where the constraint applies.<br>VALUES:<br><ul style="list-style-type: none"> <li>NFVI_POP</li> <li>ZONE</li> <li>ZONE_GROUP</li> <li>NFVI_NODE</li> </ul> |
| resource               | M         | 2..N        | ConstraintResourceRef | References to resources in the constraint rule.  |

| Attribute          | Qualifier | Cardinality | Content | Description  |
|--------------------|-----------|-------------|---------|--|
| fallbackBestEffort | M         | 0..1        | Boolean | Indication if the constraint is handled with fall back best effort. Default value is "false".<br><br>If set to true, the Affinity/Anti_Affinity placement constraint need not be fully satisfied, i.e. if the allocation cannot be honoured with the placement constraint, the request is processed in a best effort manner. |

## 8.3.7 VimConstraint information element

### 8.3.7.1 Description

This information element provides information regarding a VIM selection constraint. A set of such constraints may be sent by the VNFM to the NFVO to influence the VIM selection decisions made by the NFVO as part of the granting process.

### 8.3.7.2 Attributes

The VimConstraint information element shall follow the indications provided in table 8.3.7.2-1.

**Table 8.3.7.2-1: Attributes of the VimConstraint information element**

| Attribute         | Qualifier | Cardinality | Content               | Description   |
|-------------------|-----------|-------------|-----------------------|---|
| sameResourceGroup | M         | 0..1        | Boolean               | If present and set to true, this signals that the constraint applies not only to the same VIM connection, but also to the same infrastructure resource group.   |
| resource          | M         | 2..N        | ConstraintResourceRef | References to resources in the constraint rule.<br><br>The NFVO shall ensure that all resources in this list are managed through the same VIM connection. If "sameResourceGroup" is set to true, the NFVO shall further ensure that all resources in this list are part of the same infrastructure resource group in that VIM connection. |

## 8.3.8 ConstraintResourceRef information element

### 8.3.8.1 Description

This information element references a resource either by its VIM-level identifier for existing resources, or by the identifier of a resourceDefinition information element in the grant request for new resources.

### 8.3.8.2 Attributes

The ConstraintResourceRef information element shall follow the indications provided in table 8.3.8.2-1.

**Table 8.3.8.2-1: Attributes of the ConstraintResourceRef information element**

| Attribute          | Qualifier | Cardinality | Content                                     | Description  |
|--------------------|-----------|-------------|---|--|
| idType             | M         | 1           | Enum  | The type of the identifier.<br>VALUES: <ul style="list-style-type: none"> <li>RES_MGMT: Resource-management-level identifier; this identifier is managed by the VIM in direct mode and is managed by the NFVO in indirect mode</li> <li>GRANT: Reference to identifier in the ResourceDefinition in the grant request</li> </ul> |
| resourceId         | M         | 1           | Identifier                                  | An actual resource-management-level identifier (idType=RES_MGMT), or an identifier that references the ResourceDefinition in the related grant request (idType=GRANT).   |
| vimConnectionId    | CM        | 0..1        | Identifier (Reference to VimConnectionInfo) | Identifier of the VIM Connection. It shall only be present when idType = RES_MGMT.<br>It shall be supported when VNF-related resource management in direct mode is applicable.   |
| resourceProviderId | CM        | 0..1        | Identifier                                  | Identifier of the resource provider. It shall only be present when idType = RES_MGMT.<br>It shall be supported when VNF-related resource management in indirect mode is applicable.  |

### 8.3.9 VimAssets information element

#### 8.3.9.1 Description

This information element contains references to the asset which are defined in VNFD and managed in the VIM by the NFVO, such as compute resource flavours and/or software images.

#### 8.3.9.2 Attributes

The VimAssets information element shall follow the indications provided in table 8.3.9.2-1.

**Table 8.3.9.2-1: Attributes of the VimAssets information element**

| Attribute              | Qualifier | Cardinality | Content                   | Description  |
|------------------------|-----------|-------------|---------------------------|--|
| computeResourceFlavour | M         | 0..N        | VimComputeResourceFlavour | Mappings between virtual compute descriptors defined in the VNFD and compute resource flavours managed in the VIM. |
| softwareImage          | M         | 0..N        | VimSoftwareImage          | Mappings between software images defined in the VNFD and software images managed in the VIM.                       |
| snapshotResource       | M         | 0..N        | VimSnapshotResource       | Mappings between snapshot resources defined in the VNF snapshot package and resources managed in the VIM.          |

## 8.3.10 VimComputeResourceFlavour information element

### 8.3.10.1 Description

If the VIM requires the use of virtual compute resource flavours during compute resource instantiation, it is assumed that such flavours are selected or created by the NFVO based on the information in the VirtualComputeDesc information elements defined in the VNFD.

This information element defines the mapping between a VirtualComputeDesc in the VNFD and the corresponding compute resource flavour managed by the NFVO in the VIM.

### 8.3.10.2 Attributes

The VimComputeResourceFlavour information element shall follow the indications provided in table 8.3.10.2-1.

**Table 8.3.10.2-1: Attributes of the VimComputeResourceFlavour information element**

| Attribute                | Qualifier | Cardinality | Content                                      | Description  |
|--------------------------|-----------|-------------|--|--|
| vimConnectionId          | CM        | 0..1        | Identifier (Reference to VimConnectionInfo)  | Identifier of the VIM connection to access the flavour referenced in this information element. 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 virtualised resource.<br><br>Shall be supported and present if VNF-related resource management in indirect mode is applicable.     |
| vnfdVirtualComputeDescId | M         | 1           | Identifier (Reference to VirtualComputeDesc) | Identifier which references the VirtualComputeDesc in the VNFD that maps to this flavour.  |
| vimFlavourId             | M         | 1           | Identifier                                   | Identifier of the compute resource flavour in the resource management layer (i.e. VIM).  |

## 8.3.11 VimSoftwareImage information element

### 8.3.11.1 Description

This information element contains a mapping between a software image definition the VNFD and the corresponding software image managed by the NFVO in the VIM which is needed during compute resource instantiation.

### 8.3.11.2 Attributes

The VimSoftwareImage information element shall follow the indications provided in table 8.3.11.2-1.

**Table 8.3.11.2-1: Attributes of the VimSoftwareImage information element**

| Attribute       | Qualifier | Cardinality | Content                                     | Description   |
|-----------------|-----------|-------------|---|---|
| vimConnectionId | CM        | 0..1        | Identifier (Reference to VimConnectionInfo) | Identifier of the VIM connection to access the software image referenced in this information element. Shall be supported and present if VNF-related resource management in direct mode is applicable. |

| Attribute           | Qualifier | Cardinality | Content                                | Description   |
|---------------------|-----------|-------------|--|---|
| resourceProviderId  | CM        | 0..1        | Identifier                             | Identifier used by NFVO to determine the entity responsible for the management of the VIM asset.<br>Shall be supported and present if VNF-related resource management in indirect mode is applicable. |
| vnfdSoftwareImageld | M         | 1           | Identifier (Reference to SwlImageDesc) | Identifier of the software image descriptor in the VNFD.  |
| vimSoftwareImageld  | M         | 1           | Identifier                             | Identifier of the software image in the resource management layer (i.e. VIM).   |

## 8.3.12 VimSnapshotResource information element

### 8.3.12.1 Description

This information element contains a mapping between a snapshot resource definition related to a VNF snapshot and the corresponding resource managed by the NFVO in the VIM which is needed during the revert to VNF snapshot operation.

### 8.3.12.2 Attributes

The VimSnapshotResource information element shall follow the indications provided in table 8.3.12.2-1.

**Table 8.3.12.2-1: Attributes of the VimSnapshotResource information element**

| Attribute             | Qualifier | Cardinality | Content                                     | Description  |
|-----------------------|-----------|-------------|---|--|
| vimConnectionId       | CM        | 0..1        | Identifier (Reference to VimConnectionInfo) | References the VIM connection to access the software image referenced in this structure.<br><br>The applicable "VimConnectionInfo" structure, which is referenced by vimConnectionId, can be obtained from the "vimConnectionInfo" attribute of the "VnfInfo" structure.<br><br>CONDITION: This attribute shall only 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 virtualised resource.<br>CONDITION: This attribute shall only be supported and present if VNF-related resource management in indirect mode is applicable.  |
| vnfSnapshotId         | M         | 1           | Identifier                                  | Identifier of the VNF snapshot related to the snapshot resource.   |
| vnfcSnapshotId        | M         | 1           | Identifier                                  | Identifier of the information about a specific VNFC snapshot (refer to "VnfcSnapshotInfo") of the VNF snapshot.  |
| storageSnapshotId     | M         | 0..1        | Identifier                                  | Identifier of the virtual storage resource that has been snapshotted as referred in the VNFC snapshot information.<br><br>Shall only be present if the snapshot resource in the VIM is a storage resource (as indicated by the "type=STORAGE" in the parent resource definition).  |
| vimSnapshotResourceId | M         | 1           | Identifier                                  | Identifier of the snapshot resource in the resource management layer (i.e. VIM).   |

### 8.3.13 SnapshotResourceDefinition information element

#### 8.3.13.1 Description

This information element provides information related to a snapshot resource.

#### 8.3.13.2 Attributes

The SnapshotResourceDefinition information element shall follow the indications provided in table 8.3.13.2-1.

**Table 8.3.13.2-1: Attributes of the SnapshotResourceDefinition information element**

| Attribute   | Qualifier | Cardinality | Content        | Description  |
|---|-----------|-------------|----------------|--|
| vnfSnapshotId   | M         | 1           | Identifier     | Identifier of the VNF snapshot related to the resource change for the VNF instance.<br><br>Shall only be present if the operation to be granted concerns to creating a VNF snapshot from the VNF or to reverting the VNF to a VNF snapshot.  |
| vnfcSnapshotId  | M         | 0..1        | Identifier     | Identifier of the information about a specific VNFC snapshot (refer to "VnfcSnapshotInfo") of the VNF snapshot.<br><br>Shall only be present if the operation to be granted concerns to reverting the VNF to a VNF snapshot, and the resource is planned to be added based on the VNFC snapshot and the type of resource is "COMPUTE" or "STORAGE". See note 1 and note 2. |
| storageSnapshotId   | M         | 0..1        | Identifier     | Identifier of a snapshotted storage resource associated to the VNFC snapshot.<br><br>Shall only be present if the operation to be granted concerns to reverting the VNF to a VNF snapshot, and the storage resource is planned to be added based on the VNFC snapshot and the type of resource is "STORAGE". See note 2.   |
| snapshotResource  | M         | 0..1        | ResourceHandle | Resource information for an existing snapshot resource.<br><br>Shall only be present if the operation to be granted concerns to reverting the VNF to a VNF snapshot and the resource is planned to be added based on an existing VNF snapshot that has been created by the VNFM. Shall be absent otherwise. See note 2.  |
| <p>NOTE 1: If present, the value of the "vduld" (for a related VDU) in the "VnfcResourceInfo" referred by the "vnfcInfoId" of the "VnfcSnapshotInfo" shall match the value of the "vduld" in the resource definition that is signalled in the granting request.</p> <p>NOTE 2: For snapshot resource definitions from a VNF snapshot package, only the "vnfcSnapshotId" and "storageSnapshotId" (in case of a storage type of resource) are applicable. If the snapshot resource definition generated as part of a VNF snapshot created by the VNFM (that is, not extracted from a VNF snapshot package), the "snapshotResource" is applicable. This is a similar specification as the one defined with the "vduld", "resourceTemplateId" and "resourceHandle" attributes provided in the ResourceDefinition, but in this case applicable to resources that are defined from VNF snapshots instead of VNFD.</p> |           |             |                |  |



## 8.4 Information elements and notifications related to Virtualised Resources Management in indirect mode

### 8.4.1 Introduction

This clause defines information elements related to Virtualised Resources Management. These information elements shall be supported when VNF-related resource management in indirect mode is applicable.

### 8.4.2 Information elements related to Virtualised Compute

#### 8.4.2.1 Introduction

The clauses below define information elements related to the management of virtualised compute resources and virtualised compute resources information.

#### 8.4.2.2 ComputeResourceWithRpInfo information element

##### 8.4.2.2.1 Description

The ComputeResourceWithRpInfo information element encapsulates data of an instantiated virtualised compute resource in indirect mode.

##### 8.4.2.2.2 Attributes

The ComputeResourceWithRpInfo information element shall comply with the provisions in clause 8.4.3.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.2.2.2-1. All attributes of the VirtualCompute are also attributes of the ComputeResourceWithRpInfo.

**Table 8.4.2.2.2-1: Attributes of the ComputeResourceWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description   |
|------------------------|-----------|-------------|------------|---|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, computeId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualCompute.   |

#### 8.4.2.3 ComputeResourceWithRpId information element

##### 8.4.2.3.1 Description

This information element defines the identity of a virtualised compute resource in indirect mode.

##### 8.4.2.3.2 Attributes

The ComputeResourceWithRpId information element shall follow the indications provided in table 8.4.2.3.2-1.

**Table 8.4.2.3.2-1: Attributes of the ComputeResourceWithRpId information element**

| Attribute          | Qualifier | Cardinality | Content    | Description   |
|--------------------|-----------|-------------|------------|---|
| resourceProviderId | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, computeId]. |
| computeId          | M         | 1           | Identifier | Identifier of the compute resource within the VIM.  |

#### 8.4.2.4 VirtualComputeResourceWithRpInfo information element

##### 8.4.2.4.1 Description

The VirtualComputeResourceWithRpInfo information element defines the characteristics of a consumable virtualised compute resources in indirect mode.

##### 8.4.2.4.2 Attributes

The VirtualComputeResourceWithRpInfo information element shall comply with the provisions in clause 8.3.3.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.2.4.2-1. All attributes of the VirtualComputeResourceInformation are also attributes of the VirtualComputeResourceWithRpInfo.

**Table 8.4.2.4.2-1: Attributes of the VirtualComputeResourceWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the consumable virtualised resource and is used by the VNFM to uniquely identify consumable compute type resources by means of the tuple [resourceProviderId, computeResourceTypeId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualComputeResourceInformation.   |

#### 8.4.3 Information elements related to Virtualised Network

##### 8.4.3.1 Introduction

The clauses below define information elements related to the management of virtualised network resources and virtualised network resources information.

##### 8.4.3.2 NetworkResourceWithRpInfo information element

###### 8.4.3.2.1 Description

The NetworkResourceWithRpInfo information element encapsulates data of an instantiated virtualised network resource in indirect mode.

###### 8.4.3.2.2 Attributes

The NetworkResourceWithRpInfo information element shall comply with the provisions in clause 8.4.5.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.3.2.2-1. All attributes of the VirtualNetwork are also attributes of the NetworkResourceWithRpInfo.

**Table 8.4.3.2.2-1: Attributes of the NetworkResourceWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description   |
|------------------------|-----------|-------------|------------|---|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, networkResourceId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualNetwork.   |

##### 8.4.3.3 NetworkResourceWithRpId information element

###### 8.4.3.3.1 Description

This information element defines the identity of a virtualised network resource in indirect mode.

### 8.4.3.3.2 Attributes

The NetworkResourceWithRpId information element shall follow the indications provided in table 8.4.3.3.2-1.

**Table 8.4.3.3.2-1: Attributes of the NetworkResourceWithRpId information element**

| Attribute          | Qualifier | Cardinality | Content    | Description   |
|--------------------|-----------|-------------|------------|---|
| resourceProviderId | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, networkResourceId]. |
| networkResourceId  | M         | 1           | Identifier | Identifier of the network resource within the VIM.  |

### 8.4.3.4 VirtualNetworkResourceWithRpInfo information element

#### 8.4.3.4.1 Description

The VirtualNetworkResourceWithRpInfo information element defines the characteristics of a consumable virtualised network resource in indirect mode.

#### 8.4.3.4.2 Attributes

The VirtualNetworkResourceWithRpInfo information element shall comply with the provisions in clause 8.3.5 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.3.4.2-1. All attributes of the VirtualNetworkResourceInformation are also attributes of the VirtualNetworkResourceWithRpInfo.

**Table 8.4.3.4.2-1: Attributes of the VirtualNetworkResourceWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the consumable virtualised resource and is used by the VNFM to uniquely identify consumable network type resources by means of the tuple [resourceProviderId, networkResourceTypeId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualNetworkResourceInformation.   |

## 8.4.4 Information elements related to Virtualised Storage

### 8.4.4.1 Introduction

The clauses below define information elements related to the management of virtualised storage resources and virtualised storage resources information.

### 8.4.4.2 StorageResourceWithRpInfo information element

#### 8.4.4.2.1 Description

The StorageResourceWithRpInfo information element encapsulates data of an instantiated virtualised storage resource.

#### 8.4.4.2.2 Attributes

The StorageResourceWithRpInfo information element shall comply with the provisions in clause 8.4.7.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.4.2.2-1. All attributes of the VirtualStorage are also attributes of the StorageResourceWithRpInfo.

**Table 8.4.4.2.2-1: Attributes of the StorageResourceWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description   |
|------------------------|-----------|-------------|------------|---|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, storageId]. |
| (inherited attributes) |           |             |            | All attributes inherited from StorageResourceWithRpInfo.  |

### 8.4.4.3 StorageResourceWithRpId information element

#### 8.4.4.3.1 Description

This information element defines the identity of a virtualised storage resource in indirect mode.

#### 8.4.4.3.2 Attributes

The StorageResourceWithRpId information element shall follow the indications provided in table 8.4.4.3.2-1.

**Table 8.4.4.3.2-1: Attributes of the StorageResourceWithRpId information element**

| Attribute          | Qualifier | Cardinality | Content    | Description   |
|--------------------|-----------|-------------|------------|---|
| resourceProviderId | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, storageId]. |
| storageId          | M         | 1           | Identifier | Identifier of the storage resource within the VIM.  |

### 8.4.4.4 VirtualStorageResourceWithRpInfo information element

#### 8.4.4.4.1 Description

The VirtualStorageResourceWithRpInfo information element defines the characteristics of a consumable virtualised storage resource in indirect mode.

#### 8.4.4.4.2 Attributes

The VirtualStorageResourceWithRpInfo information element shall comply with the provisions in clause 8.3.4 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.4.4.2-1. All attributes of the VirtualStorageResourceInformation are also attributes of the VirtualStorageResourceWithRpInfo.

**Table 8.4.4.4.2-1: Attributes of the VirtualStorageResourceWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the consumable virtualised resource and is used by the VNFM to uniquely identify consumable storage type resources by means of the tuple [resourceProviderId, storageResourceTypeId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualStorageResourceInformation.   |

## 8.4.5 Notifications related to changes of virtualised resources

### 8.4.5.1 Introduction

The clauses below define notifications related to changes of virtualised resources.

## 8.4.5.2 VirtualisedResourceWithRpChangeNotification

### 8.4.5.2.1 Description

This notification informs the receiver of changes in the virtualised resources that are allocated and is applicable in the indirect mode of VNF-related resource reservation management.

Support of this notification is mandatory.

### 8.4.5.2.2 Trigger conditions

This notification is triggered with the same trigger conditions applicable to the VirtualisedResourceChangeNotification in clause 8.4.9 of ETSI GS NFV-IFA 006 [1].

### 8.4.5.2.3 Attributes

The VirtualisedResourceWithRpChangeNotification shall comply with the indications in clause 8.4.9 of ETSI GS NFV-IFA 006 [1] with additional attributes of the notification according to table 8.4.5.2.3-1. All attributes of the VirtualisedResourceChangeNotification are also attributes of the VirtualisedResourceWithRpChangeNotification.

**Table 8.4.5.2.3-1: Attributes of the VirtualisedResourceWithRpChangeNotification**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the change notification and is used by the VNFM to uniquely identify the resource by means of the tuple [resourceProviderId, resourceTypeId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualisedResourceChangeNotification.   |

## 8.4.5.3 InformationWithRpChangeNotification

### 8.4.5.3.1 Description

This notification informs the receiver that information related to consumable virtualised resources is changed and is applicable in the indirect mode.

Support of this notification is mandatory.

### 8.4.5.3.2 Trigger conditions

This notification is triggered with the same trigger conditions applicable to the InformationChangeNotification in clause 8.3.2 of ETSI GS NFV-IFA 006 [1].

### 8.4.5.3.3 Attributes

The InformationWithRpChangeNotification shall comply with the indications in clause 8.3.2 of ETSI GS NFV-IFA 006 [1] with additional attributes of the notification according to table 8.4.5.3.3-1. All attributes of the InformationChangeNotification are also attributes of the InformationWithRpChangeNotification.

**Table 8.4.5.3.3-1: Attributes of the InformationWithRpChangeNotification**

| Attribute              | Qualifier | Cardinality | Content    | Description   |
|------------------------|-----------|-------------|------------|---|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the change notification and is used by the VNFM to uniquely identify the consumable resource by means of the tuple [resourceProviderId, resourceTypeId]. |
| (inherited attributes) |           |             |            | All attributes inherited from InformationChangeNotification.  |

## 8.4.6 Notifications related to Virtualised Resource Performance Management

### 8.4.6.1 Introduction

The clauses below define notifications related of virtualised resource performance management.

### 8.4.6.2 PerformanceInformationWithRpAvailableNotification

#### 8.4.6.2.1 Description

This notification informs the receiver that performance information is available and is applicable in the indirect mode of VNF-related resource reservation management.

Support of this notification is mandatory.

#### 8.4.6.2.2 Trigger conditions

This notification is triggered with the same trigger conditions applicable to the PerformanceInformationAvailableNotification in clause 8.5.8 of ETSI GS NFV-IFA 006 [1].

#### 8.4.6.2.3 Attributes

The PerformanceInformationWithRpAvailableNotification shall comply with the indications in clause 8.5.8 of ETSI GS NFV-IFA 006 [1] with additional attributes of the notification according to table 8.4.6.2.3-1. All attributes of the PerformanceInformationAvailableNotification are also attributes of the PerformanceInformationWithRpAvailableNotification.

**Table 8.4.6.2.3-1: Attributes of the PerformanceInformationWithRpAvailableNotification**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the resources and is used by the VNFM to uniquely identify the resources for which information is available by means of the tuple [resourceProviderId, objectInstanceId]. |
| (inherited attributes) |           |             |            | All attributes inherited from PerformanceInformationAvailableNotification.   |

### 8.4.6.3 ThresholdCrossedWithRpNotification

#### 8.4.6.3.1 Description

This notification informs the receiver that a threshold value has been crossed and is applicable in the indirect mode of VNF-related resource reservation management.

Support of this notification is mandatory.

#### 8.4.6.3.2 Trigger conditions

This notification is triggered with the same trigger conditions applicable to the PerformanceInformationAvailableNotification in clause 8.5.9 of ETSI GS NFV-IFA 006 [1].

#### 8.4.6.3.3 Attributes

The ThresholdCrossedWithRpNotification shall comply with the indications in clause 8.5.9 of ETSI GS NFV-IFA 006 [1] with additional attributes of the notification according to table 8.4.6.3.3-1. All attributes of the ThresholdCrossedNotification are also attributes of the ThresholdCrossedWithRpNotification.

**Table 8.4.6.3.3-1: Attributes of the ThresholdCrossedWithRpNotification**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the resources and is used by the VNFM to uniquely identify the resources for which the threshold is crossed by means of the tuple [resourceProviderId, objectInstanceId]. |
| (inherited attributes) |           |             |            | All attributes inherited from ThresholdCrossedNotification.  |

## 8.4.7 Information elements and notifications related to Virtualised Resource Fault Management

### 8.4.7.1 Introduction

The clauses below define notifications related to virtualised resources fault management.

### 8.4.7.2 AlarmWithRpInfo information element

#### 8.4.7.2.1 Description

The AlarmWithRpInfo information element encapsulates data of a virtualised resource alarm in indirect mode.

#### 8.4.7.2.2 Attributes

The AlarmWithRpInfo information element shall comply with the provisions in clause 8.6.4 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.7.2.2-1. All attributes of the Alarm are also attributes of the AlarmWithRpInfo.

**Table 8.4.7.2.2-1: Attributes of the AlarmWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description   |
|------------------------|-----------|-------------|------------|---|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for issuing the alarm, and is used by the VNFM to uniquely identify resources by means of the tuple [resourceProviderId, managedObjectId]. |
| (inherited attributes) |           |             |            | All attributes inherited from Alarm.  |

### 8.4.7.3 AlarmWithRpNotification

#### 8.4.7.3.1 Description

This notification encapsulates information on an alarm and is applicable in the indirect mode of VNF-related resource reservation management.

Support of this notification is mandatory.

#### 8.4.7.3.2 Trigger conditions

This notification is triggered with the same trigger conditions applicable to the AlarmNotification in clause 8.6.2 of ETSI GS NFV-IFA 006 [1].

### 8.4.7.3.3 Attributes

The AlarmWithRpNotification shall comply with the indications in clause 8.6.2 of ETSI GS NFV-IFA 006 [1] with additional attributes of the notification according to table 8.4.7.3.3-1. All attributes of the AlarmNotification are also attributes of the AlarmWithRpNotification.

**Table 8.4.7.3.3-1: Attributes of the AlarmWithRpNotification**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the alarm and is used by the VNFM to uniquely identify the alarm by means of the tuple [resourceProviderId, alarmId]. |
| (inherited attributes) |           |             |            | All attributes inherited from AlarmNotification.   |

### 8.4.7.4 AlarmClearedWithRpNotification

#### 8.4.7.4.1 Description

This notification encapsulates information on a cleared alarm and is applicable in the indirect mode of VNF-related resource reservation management.

Support of this notification is mandatory.

#### 8.4.7.4.2 Trigger conditions

This notification is triggered with the same trigger conditions applicable to the AlarmClearedNotification in clause 8.6.3 of ETSI GS NFV-IFA 006 [1].

#### 8.4.7.4.3 Attributes

The AlarmClearedWithRpNotification shall comply with the indications in clause 8.6.3 of ETSI GS NFV-IFA 006 [1] with additional attributes of the notification according to table 8.4.7.4.3-1. All attributes of the AlarmClearedNotification are also attributes of the AlarmClearedWithRpNotification.

**Table 8.4.7.4.3-1: Attributes of the AlarmClearedWithRpNotification**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the alarm and is used by the VNFM to uniquely identify the alarm by means of the tuple [resourceProviderId, alarmId]. |
| (inherited attributes) |           |             |            | All attributes inherited from AlarmClearedNotification.  |

## 8.4.8 Information elements and notifications related to Virtualised Resources Quota

### 8.4.8.1 Introduction

The clauses below define information elements and notifications related to the management of virtualised resources quota.



## 8.4.8.2 VirtualComputeQuotaWithRpInfo information element

### 8.4.8.2.1 Description

The VirtualComputeQuotaWithRpInfo information element encapsulates information about a quota for virtualised compute resources.

### 8.4.8.2.2 Attributes

The VirtualComputeQuotaWithRpInfo information element shall comply with the provisions in clause 8.8.2.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.8.2.2-1. All attributes of the VirtualComputeQuota are also attributes of the VirtualComputeQuotaWithRpInfo.

**Table 8.4.8.2.2-1: Attributes of the VirtualComputeQuotaWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by the NFVO to determine the entity responsible for the management of the virtualised resources quota and is used by the VNFM to uniquely identify resources quota by means of the tuple [resourceProviderId, resourceGroupId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualComputeQuota.   |

## 8.4.8.3 VirtualNetworkQuotaWithRpInfo information element

### 8.4.8.3.1 Description

The VirtualNetworkQuotaWithRpInfo information element encapsulates information about a quota for virtualised network resources.

### 8.4.8.3.2 Attributes

The VirtualNetworkQuotaWithRpInfo information element shall comply with the provisions in clause 8.8.3.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.8.3.2-1. All attributes of the VirtualNetworkQuota are also attributes of the VirtualNetworkQuotaWithRpInfo.

**Table 8.4.8.3.2-1: Attributes of the VirtualNetworkQuotaWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by the NFVO to determine the entity responsible for the management of the virtualised resources quota and is used by the VNFM to uniquely identify resources quota by means of the tuple [resourceProviderId, resourceGroupId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualNetworkQuota.   |

## 8.4.8.4 VirtualStorageQuotaWithRpInfo information element

### 8.4.8.4.1 Description

The VirtualStorageQuotaWithRpInfo information element encapsulates information about a quota for virtualised storage resources.

### 8.4.8.4.2 Attributes

The VirtualStorageQuotaWithRpInfo information element shall comply with the provisions in clause 8.8.4.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.8.4.2-1. All attributes of the VirtualStorageQuota are also attributes of the VirtualStorageQuotaWithRpInfo.

**Table 8.4.8.4.2-1: Attributes of the VirtualStorageQuotaWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by the NFVO to determine the entity responsible for the management of the virtualised resources quota and is used by the VNFM to uniquely identify resources quota by means of the tuple [resourceProviderId, resourceGroupId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualStorageQuota.   |

## 8.4.8.5 VirtualisedResourceQuotaWithRpChangeNotification

### 8.4.8.5.1 Description

This notification indicates a change in a virtualised resource quota and is applicable in the indirect mode of resource quota management. Support of this notification is mandatory.

### 8.4.8.5.2 Trigger conditions

This notification is triggered with the same trigger conditions applicable to the VirtualisedResourceQuotaChangeNotification in clause 8.8.5.2 of ETSI GS NFV-IFA 006 [1].

### 8.4.8.5.3 Attributes

The VirtualisedResourceQuotaWithRpChangeNotification shall comply with the provisions in clause 8.8.5 of ETSI GS NFV-IFA 006 [1] with additional attributes of the notification according to table 8.4.8.5.3-1. All attributes of the VirtualisedResourceQuotaChangeNotification are also attributes of the VirtualisedResourceQuotaWithRpChangeNotification.

**Table 8.4.8.5.3-1: Attributes of the VirtualisedResourceQuotaWithRpChangeNotification**

| Attribute              | Qualifier | Cardinality | Content    | Description  |
|------------------------|-----------|-------------|------------|--|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the change notification and is used by the VNFM to uniquely identify the resource quota by means of the tuple [resourceProviderId, resourceTypeId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualisedResourceQuotaChangeNotification.  |

## 8.4.9 Information elements and notifications related to Virtualised Resources Reservation

### 8.4.9.1 Introduction

The clauses below define information elements and notifications related to the management of virtualised resources reservations.

### 8.4.9.2 ReservedVirtualComputeWithRpInfo information element

#### 8.4.9.2.1 Description

The ReservedVirtualComputeWithRpInfo information element encapsulates information about a reservation for virtualised compute resources.

### 8.4.9.2.2 Attributes

The ReservedVirtualComputeWithRpInfo information element shall comply with the provisions in clause 8.7.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.9.2.2-1. All attributes of the ReservedVirtualCompute are also attributes of the ReservedVirtualComputeWithRpInfo.

**Table 8.4.9.2.2-1: Attributes of the ReservedVirtualComputeWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description   |
|------------------------|-----------|-------------|------------|---|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources reservation by means of the tuple [resourceProviderId, reservationId]. |
| (inherited attributes) |           |             |            | All attributes inherited from ReservedVirtualCompute.   |

### 8.4.9.3 ReservedVirtualNetworkWithRpInfo information element

#### 8.4.9.3.1 Description

The ReservedVirtualNetworkWithRpInfo information element encapsulates information about a reservation for virtualised network resources.

#### 8.4.9.3.2 Attributes

The ReservedVirtualNetworkWithRpInfo information element shall comply with the provisions in clause 8.7.4.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.9.3.2-1. All attributes of the ReservedVirtualNetwork are also attributes of the ReservedVirtualNetworkWithRpInfo.

**Table 8.4.9.3.2-1: Attributes of the ReservedVirtualNetworkWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description   |
|------------------------|-----------|-------------|------------|---|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources reservation by means of the tuple [resourceProviderId, reservationId]. |
| (inherited attributes) |           |             |            | All attributes inherited from ReservedVirtualNetwork.   |

### 8.4.9.4 ReservedVirtualStorageWithRpInfo information element

#### 8.4.9.4.1 Description

The ReservedVirtualStorageWithRpInfo information element encapsulates information about a reservation for virtualised storage resources.

#### 8.4.9.4.2 Attributes

The ReservedVirtualStorageWithRpInfo information element shall comply with the provisions in clause 8.7.6.2 of ETSI GS NFV-IFA 006 [1] with additional attributes provided in table 8.4.9.4.2-1. All attributes of the ReservedVirtualStorage are also attributes of the ReservedVirtualStorageWithRpInfo.

**Table 8.4.9.4.2-1: Attributes of the ReservedVirtualStorageWithRpInfo information element**

| Attribute              | Qualifier | Cardinality | Content    | Description   |
|------------------------|-----------|-------------|------------|---|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the management of the Virtualised resource and is used by the VNFM to uniquely identify resources reservation by means of the tuple [resourceProviderId, reservationId]. |
| (inherited attributes) |           |             |            | All attributes inherited from ReservedVirtualStorage.   |

## 8.4.9.5 VirtualisedResourceReservationWithRpChangeNotification

### 8.4.9.5.1 Description

This notification indicates a change in a virtualised resource reservation and is applicable in the indirect mode of VNF-related resource reservation management.

Support of this notification is mandatory.

### 8.4.9.5.2 Trigger conditions

This notification is triggered with the same trigger conditions applicable to the VirtualisedResourceReservationChangeNotification in clause 8.7.7.2 of ETSI GS NFV-IFA 006 [1].

### 8.4.9.5.3 Attributes

The VirtualisedResourceReservationWithRpChangeNotification shall comply with the provisions in clause 8.7.7 of ETSI GS NFV-IFA 006 [1] with additional attributes of the notification according to table 8.4.9.5.3-1. All attributes of the VirtualisedResourceReservationChangeNotification are also attributes of the VirtualisedResourceReservationWithRpChangeNotification.

**Table 8.4.9.5.3-1: Attributes of the VirtualisedResourceReservationWithRpChangeNotification**

| Attribute              | Qualifier | Cardinality | Content    | Description   |
|------------------------|-----------|-------------|------------|---|
| resourceProviderId     | M         | 1           | Identifier | It is used by NFVO to determine the entity responsible for the change notification and is used by the VNFM to uniquely identify the resource reservation by means of the tuple [resourceProviderId, reservationId]. |
| (inherited attributes) |           |             |            | All attributes inherited from VirtualisedResourceReservationChangeNotification.   |

## 8.5 Information elements related to VNF Lifecycle Management

### 8.5.1 Introduction

This clause defines information elements related to VNF Lifecycle Management.

### 8.5.2 VnfInfo information element

#### 8.5.2.1 Description

The VnfInfo information element provides run-time information about a VNF instance.

**NOTE:** In ETSI GS NFV-MAN 001 [i.7], the concept of the VNF record (VNFR) was introduced which is a model for the totality of information managed by the VNFM regarding a running VNF instance. VNFR is not used in the present document.

#### 8.5.2.2 Attributes

The VnfInfo information element shall follow the indications provided in table 8.5.2.2-1.

**Table 8.5.2.2-1: 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.  |

| Attribute               | Qualifier | Cardinality | Content                           | Description  |
|-------------------------|-----------|-------------|-----------------------------------|--|
| vnfInstanceDescription  | M         | 0..1        | String                            | Human-readable description of the VNF instance.<br>See note 1.   |
| vnfdId                  | M         | 1           | Identifier<br>(Reference to Vnfd) | Identifier of the VNFD on which the VNF instance is based. See notes 1, 2 and 4.   |
| vnfProvider             | M         | 1           | String                            | See note 3.  |
| vnfProductName          | M         | 1           | String                            | See note 3.  |
| vnfSoftwareVersion      | M         | 1           | Version                           | See note 3.  |
| vnfdVersion             | M         | 1           | Version                           | See note 3.  |
| vnfConfigurableProperty | M         | 0..N        | KeyValuePair                      | Additional VNF-specific attributes that provide the current values of the configurable properties of the VNF instance.<br><br>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 shall be declared in the VNFD (see clause 7.1.12 in ETSI GS NFV-IFA 011 [3]).<br>See notes 1 and 5.  |
| vimConnectionInfo       | CM        | 0..N        | VimConnectionInfo                 | Information about VIM connection(s) for managing resources for the VNF instance. Shall be supported and present if VNF-related resource management in direct mode is applicable.<br>See note 1.  |
| instantiationState      | M         | 1           | Enum                              | The instantiation state of the VNF instance.<br>VALUES: <ul style="list-style-type: none"> <li>NOT_INSTANTIATED: VNF instance is terminated or not instantiated, and the identifier of the VNF instance exists)</li> <li>INSTANTIATED: VNF instance is instantiated</li> </ul>   |
| instantiatedVnfInfo     | M         | 0..1        | InstantiatedVnfInfo               | Information specific to an instantiated VNF instance.<br>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.<br><br>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.<br><br>Modifying the values of these attributes has no effect on the VNF instance, it only affects the information represented in VnfInfo.<br><br>Metadata that the VNF provider foresees shall be declared in the VNFD (see clause 7.1.14.2 in ETSI GS NFV-IFA 011 [3]). The VNFM shall accept requests to write metadata that are not declared in the VNFD. See note 1. |

| Attribute  | Qualifier | Cardinality | Content      | Description   |
|--|-----------|-------------|--------------|---|
| extension  | M         | 0..N        | KeyValuePair | <p>Additional VNF-specific attributes that affect the lifecycle management of this VNF instance.</p> <p>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.</p> <p>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.</p> <p>All extensions that are allowed for the VNF shall be declared in the VNFD (see clause 7.1.14.2 in ETSI GS NFV-IFA 011 [3]). See note 1.</p> |
| <p>NOTE 1: This attribute in the VnfInfo shall be writable through the Modify VNF information operation (refer to clause 7.2.12).</p> <p>NOTE 2: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way.</p> <p>NOTE 3: See ETSI GS NFV-IFA 011 [3], 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.</p> <p>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.</p> <p>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.</p> |           |             |              |   |

## 8.5.3 InstantiatedVnfInfo information element

### 8.5.3.1 Description

This information element provides run-time information specific to an instantiated VNF instance.

Annex A 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.

### 8.5.3.2 Attributes

The InstantiatedVnfInfo information element shall follow the indications provided in table 8.5.3.2-1.

**Table 8.5.3.2-1: 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                            | <p>The state of the VNF instance.</p> <p>VALUES:</p> <ul style="list-style-type: none"> <li>• STARTED</li> <li>• STOPPED</li> </ul> |

| Attribute  | Qualifier | Cardinality | Content                    | Description  |
|--|-----------|-------------|----------------------------|--|
| scaleStatus  | M         | 0..N        | ScaleInfo                  | Scale status of the VNF, one entry per aspect. Shall be present if the VNF supports scaling.<br><br>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.<br><br>Represents for every scaling aspect how "big" the VNF can be scaled w.r.t. that aspect. See note 2. |
| extCplInfo   | M         | 1..N        | VnfExtCplInfo              | External CPs exposed by the VNF instance.  |
| 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).<br><br>The localization languages supported by a VNF can be declared in the VNFD, and localization language selection can take place at instantiation time.            |
| 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.   |
| <p>NOTE 1: The VnfDf information element is defined in ETSI GS NFV-IFA 011 [3], clause 7.1.8.2.</p> <p>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.</p> <p>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.</p> <p>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 8.5.10).</p> |           |             |                            |  |

## 8.5.4 VnfcResourceInfo information element

### 8.5.4.1 Description

This information element provides information on virtualised compute and storage resources used by a VNFC in a VNF instance.

### 8.5.4.2 Attributes

The VnfcResourceInfo information element shall follow the indications provided in table 8.5.4.2-1.

**Table 8.5.4.2-1: Attributes of the VnfcResourceInfo information element**

| Attribute         | Qualifier | Cardinality | Content  | Description  |
|-------------------|-----------|-------------|--|--|
| vnfcInstanceld    | M         | 1           | Identifier   | Identifier of this VNFC instance.  |
| vduld             | M         | 1           | Identifier (Reference to Vdu)                        | Reference to the applicable Vdu information element in the VNFD.   |
| vnfdld            | M         | 0..1        | Identifier (Reference to Vnfd)                       | Reference to the VNFD.<br><br>Shall be present in case the value differs from the vnfdld 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.<br><br>Detailed information is (for new and modified resources) or has been (for removed resources) available from the VIM.                                      |
| storageResourceId | M         | 0..N        | Identifier (Reference to VirtualStorageResourceInfo) | Reference(s) to the VirtualStorage resource(s).<br><br>Information about the resource(s) is available from the VIM.  |
| reservationId     | M         | 0..1        | Identifier   | The reservation identifier applicable to the resource. It shall be present when an applicable reservation exists.  |
| vnfcCplInfo       | M         | 0..N        | VnfcCplInfo  | CP(s) of the VNFC instance.<br>Shall be present when that particular CP of the VNFC instance is associated to an external CP of the VNF instance.<br>May be present otherwise.                             |
| metadata          | M         | 0..N        | KeyValuePair   | Metadata about this resource.  |

## 8.5.5 VnfVirtualLinkResourceInfo information element

### 8.5.5.1 Description

This information element provides information on virtualised network resources used by an internal VL instance in a VNF.

### 8.5.5.2 Attributes

The VnfVirtualLinkResourceInfo information element shall follow the indications provided in table 8.5.5.2-1.



**Table 8.5.2-1: Attributes of the VnfVirtualLinkResourceInfo information element**

| Attribute              | Qualifier | Cardinality | Content                                      | Description  |
|------------------------|-----------|-------------|--|--|
| virtualLinkInstancelId | M         | 1           | Identifier                                   | Identifier of 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)               | Reference to the VNFD.<br><br>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.<br><br>Detailed information is (for new and modified resources) or has been (for removed resources) available from the VIM.                                      |
| 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.<br>Shall be present when the linkPort is used for external connectivity by the VNF (refer to VnfLinkPortInfo in clause 8.5.11).<br>May be present otherwise.                       |
| metadata               | M         | 0..N        | KeyValuePair                                 | Metadata about this resource.  |

## 8.5.6 VirtualStorageResourceInfo information element

### 8.5.6.1 Description

This information element provides information on virtualised storage resources used by a storage instance in a VNF.

### 8.5.6.2 Attributes

The VirtualStorageResourceInfo information element shall follow the indications provided in table 8.5.6.2-1.

**Table 8.5.6.2-1: Attributes of the VirtualStorageResourceInfo information element**

| Attribute                 | Qualifier | Cardinality | Content                                      | Description  |
|---------------------------|-----------|-------------|--|--|
| virtualStorageInstancelId | 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)               | Reference to the VNFD.<br><br>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.<br><br>Detailed information is (for new and modified resources) or has been (for removed resources) available from the VIM.                                      |
| 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.  |

## 8.5.7 ResourceHandle information element

### 8.5.7.1 Description

This information element provides information that allows addressing a resource that is used by a VNF instance.

Information about the resource is available from the corresponding Virtualised Compute/Storage/Network Resource Management interfaces. Table 8.5.7.1-1 shows the relationship between the resourceId attribute of ResourceHandle specified in the present document and the resource identifiers used in the aforementioned interfaces specified in ETSI GS NFV-IFA 005 [i.4] and ETSI GS NFV-IFA 006 [1].

**Table 8.5.7.1-1: Relationship between resource identifiers**

| Attribute in Or-Vnfm ref. point | Type, Interface, information element and attribute in ETSI GS NFV-IFA 005 [i.4] and ETSI GS NFV-IFA 006 [1] |   |                                   |
|---------------------------------|---|---|-----------------------------------|
|                                 | Type  | Interface                               | Information element and attribute |
| ResourceHandle:resourceId       | Compute   | Virtualised Compute Resource Management | VirtualCompute:computeld          |
|                                 | Storage   | Virtualised Storage Resource Management | VirtualStorage:storageld          |
|                                 | Network   | Virtualised Network Resource Management | VirtualNetwork:networkResourceld  |

### 8.5.7.2 Attributes

The ResourceHandle information element shall follow the indications provided in table 8.5.7.2-1.

**Table 8.5.7.2-1: Attributes of the ResourceHandle information element**

| Attribute   | Qualifier | Cardinality | Content                                     | Description  |
|---|-----------|-------------|---|--|
| vimConnectionId   | CM        | 0..1        | Identifier (Reference to VimConnectionInfo) | Reference to the identifier of the VimConnectionInfo information element defining the VIM Connection to manage this resource. This attribute shall be supported when VNF-related Resource Management in direct mode is applicable. |
| resourceProviderId  | CM        | 0..1        | Identifier                                  | Identifies the entity responsible for the management of the virtualised resource. 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 or the resource provider.   |
| vimLevelResourceType  | M         | 0..1        | Not specified                               | Type of the resource in the scope of the VIM or the resource provider. See note.   |
| NOTE: The value set of the "vimLevelResourceType" attribute is within the scope of the VIM or the resource provider and can be used as information that complements the ResourceHandle. This value set is different from the value set of the "type" attribute in the ResourceDefinition (refer to clause 8.3.2). |           |             |   |  |

## 8.5.8 ScaleInfo information element

### 8.5.8.1 Description

This information element provides information about the scale level of a VNF instance w.r.t. one scaling aspect.

### 8.5.8.2 Attributes

The ScaleInfo information element shall follow the indications provided in table 8.5.8.2-1.

**Table 8.5.8.2-1: 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)          | Reference to the VNFD.<br><br>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.<br><br>Minimum value 0, maximum value maxScaleLevel as declared in the VNFD (see ETSI GS NFV-IFA 011 [3], clause 7.1.10.2.2).   |

## 8.5.9 ExtVirtualLinkInfo information element

### 8.5.9.1 Description

This information element provides a reference to an external VL.

### 8.5.9.2 Attributes

The ExtVirtualLinkInfo information element shall follow the indications provided in table 8.5.9.2-1.

**Table 8.5.9.2-1: 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.   |

## 8.5.10 ExtManagedVirtualLinkInfo information element

### 8.5.10.1 Description

This information element provides a reference to an externally-managed internal VL.

### 8.5.10.2 Attributes

The ExtManagedVirtualLinkInfo information element shall follow the indications provided in table 8.5.10.2-1.

**Table 8.5.10.2-1: 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.  |

| Attribute                        | Qualifier | Cardinality | Content                        | Description  |
|----------------------------------|-----------|-------------|--------------------------------|--|
| vnfdId                           | M         | 0..1        | Identifier (Reference to Vnfd) | Reference to the VNFD.<br><br>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 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 virtualLinkId shall refer to the same extManagedMultisiteVirtualLinkId. |

## 8.5.11 VnfLinkPortInfo information element

### 8.5.11.1 Description

This information element provides information about a port of a VNF's internal VL. See also VnfVirtualLinkResourceInfo in clause 8.5.5.

### 8.5.11.2 Attributes

The attributes of the VnfLinkPortInfo information element shall follow the indications provided in table 8.5.11.2-1.

**Table 8.5.11.2-1: 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.  |
| associatedExtCpld   | M         | 0..1        | Identifier (Reference to VnfExtCplInfo) | 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. |
| vnfcCplInstanceId   | M         | 0..1        | Identifier (Reference to VnfcCplInfo)   | VNFC CP of the VNF connected to this link port. May be present. See notes 1, 3 and 4.  |
| 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. |           |             |   |  |
| NOTE 2: A VnfLinkPort does not terminate on an external CP, as external CPs are connected to external VLs.  |           |             |   |  |
| NOTE 3: Either associatedExtCpld or vnfcCplInstanceId is present, but not both shall be present for a VnfLinkPortInfo.                                    |           |             |   |  |

| Attribute  | Qualifier | Cardinality | Content | Description |
|--|-----------|-------------|---------|-------------|
| NOTE 4: The attributes "associatedExtCpld" and "vnfcCplInstanceId" model two separate associations in the information model. To represent these in the data model during the protocol design stage, an alternative representation of these associations could be chosen as well. |           |             |         |             |

## 8.5.12 VnfExtCplInfo information element

### 8.5.12.1 Description

This information element provides information related to an external CP.

### 8.5.12.2 Attributes

The VnfExtCplInfo information element shall follow the indications provided in table 8.5.12.2-1.

**Table 8.5.12.2-1: Attributes of the VnfExtCplInfo information element**

| Attribute  | Qualifier | Cardinality | Content   | Description  |
|--|-----------|-------------|---|--|
| cpInstanceId   | M         | 1           | Identifier  | Identifier of this external CP instance and of this VnfExtCplInfo information element.   |
| cpId   | M         | 1           | Identifier<br>(Reference to VnfExtCpd)                  | Identifier of the external Connection Point Descriptor (CPD), VnfExtCpd, in the VNFD.  |
| vnfdId   | M         | 0..1        | Identifier<br>(Reference to Vnfd)                       | Reference to the VNFD.<br><br>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.   |
| associatedVnfcCpld   | M         | 0..1        | Identifier<br>(Reference to VnfcCplInfo)                | Identifier of the VnfcCp that is directly exposed as this VnfExtCp. Shall be present if the cpId of this VnfExtCp has an intCpd attribute. See note.   |
| associatedVnfVirtualLinkId   | M         | 0..1        | Identifier<br>(reference to VnfVirtualLinkResourceInfo) | Identifier of the Vnf VL that this VnfExtCP maps to. Shall be present if the cpId of this VnfExtCp has an intVirtualLinkDesc attribute. See note.  |
| extLinkPortId  | M         | 0..1        | Identifier<br>(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.  |
| metadata   | M         | 0..N        | KeyValuePair  | Metadata about this external CP.   |
| NOTE: The attributes associatedVnfcCpld and associatedVnfVirtualLinkId are exclusive. One and only one shall be present. |           |             |   |  |

## 8.5.13 ExtLinkPortInfo information element

### 8.5.13.1 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.

### 8.5.13.2 Attributes

The attributes of the ExtLinkPortInfo information element shall follow the indications provided in table 8.5.13.2-1.

**Table 8.5.13.2-1: 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.                          |

NOTE: There shall be at most one link port associated with any external connection point instance.

## 8.5.14 VnfcCpInfo information element

### 8.5.14.1 Description

This information element provides information related to a CP of a VNFC.

### 8.5.14.2 Attributes

The VnfcCpInfo information element shall follow the indications provided in table 8.5.14.2-1.

**Table 8.5.14.2-1: Attributes of the VnfcCpInfo information element**

| Attribute      | Qualifier | Cardinality | Content                                   | Description   |
|----------------|-----------|-------------|---|---|
| cpInstanceId   | M         | 1           | Identifier                                | Identifier of this VnfcCpInfo information element.  |
| cpId           | M         | 1           | Identifier (Reference to VduCpd)          | Identifier of the VDU CPD, cpId, 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.  |

## 8.5.15 CpProtocolInfo information element

### 8.5.15.1 Description

This information element describes and associates the protocol layer that a CP uses together with other protocol-related information, like addresses.

### 8.5.15.2 Attributes

The CpProtocolInfo information element shall follow the indications provided in table 8.5.15.2-1.

**Table 8.5.15.2-1: 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.<br>VALUES: <ul style="list-style-type: none"> <li>• IP_OVER_ETHERNET</li> <li>• Etc.</li> </ul> |
| address       | M         | 1..N        | Not specified. | List of network addresses that have been configured (statically or dynamically) on the link port that connects the CP to a VL. See note 2.                                   |

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.

## 8.5.16 VnfSnapshotInfo information element

### 8.5.16.1 Description

This information element provides the details of a VNF Snapshot, which the VNFM creates and stores as part of the ongoing VNF Lifecycle Management operations related to VNF Snapshots.

### 8.5.16.2 Attributes

The VnfSnapshotInfo information element shall follow the indications provided in table 8.5.16.2-1.

**Table 8.5.16.2-1: 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.<br>Cardinality is 0 when the VNF Snapshot creation has not yet completed and shall be 1 afterwards.  |
| 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 1 and 2.   |
| vnfInfo              | M         | 1           | VnfInfo                        | VnfInfo of the snapshotted VNF instance.   |
| vnfcSnapshotInfo     | M         | 1..N        | VnfcSnapshotInfo               | Information about VNFC Snapshots constituting this VNF Snapshot.   |
| vnfStateSnapshotInfo | M         | 0..1        | VnfStateSnapshotInfo           | Information about VNF-specific state snapshot data.<br>This attribute shall not be present before the VNF snapshot has been completed. Otherwise, this attribute shall be present if the VNF snapshot has associated additional VNF-specific state data. |
| userDefinedData      | O         | 0..N        | KeyValuePair                   | User defined data for the VNF Snapshot.  |

NOTE 1: This identifier, which is managed by the VNF provider, identifies the VNF Package and the VNFD in a globally unique way.  
NOTE 2: See ETSI GS NFV-IFA 011 [3], 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.

## 8.5.17 VnfcSnapshotInfo information element

### 8.5.17.1 Description

This information element provides the details of a VNFC Snapshot, which the VNFM creates and stores as part of the ongoing VNF Lifecycle Management operations related to VNF Snapshots.

### 8.5.17.2 Attributes

The VnfcSnapshotInfo information element shall follow the indications provided in table 8.5.17.2-1.

**Table 8.5.17.2-1: 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 VNFC Snapshot creation has been started.   |
| createdAt   | M         | 0..1        | DateTime                                   | Timestamp indicating when the VNFC Snapshot creation has been completed.<br>Cardinality is 0 when the VNF Snapshot creation has not yet completed and shall be 1 afterwards. |
| 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.  |
| storageSnapshotResource   | M         | 0..N        | StorageSnapshotResource                    | Mapping of the storage resources associated to the VNFC with the storage snapshot resources.   |
| userDefinedData   | O         | 0..N        | KeyValuePair                               | User defined data for the VNFC Snapshot.   |
| NOTE: 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". |           |             |  |  |

## 8.5.18 StorageSnapshotResource information element

### 8.5.18.1 Description

This information element provides a mapping of the storage resources associated to the VNFC with the storage snapshot resources.



## 8.5.18.2 Attributes

The StorageSnapshotResource information element shall follow the indications provided in table 8.5.18.2-1.

**Table 8.5.18.2-1: 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". |           |             |  |   |

## 8.6 Information elements and notifications related to VNF Lifecycle Changes

### 8.6.1 Introduction

This clause defines notifications related to VNF lifecycle changes and update of VNF information.

### 8.6.2 VnfLcmOperationOccurrenceNotification

#### 8.6.2.1 Description

This notification informs the receiver of changes in the VNF lifecycle caused by a VNF lifecycle management operation occurrence. The support of the notification is mandatory.

#### 8.6.2.2 Trigger conditions

This notification is produced when there is a change in the VNF lifecycle caused by a VNF lifecycle management operation occurrence, including:

- Instantiation of the VNF.
- Scaling of the VNF instance (including auto-scaling).
- Healing of the VNF instance (including auto-healing).
- Change of the state of the VNF instance (i.e. Operate VNF).
- Change of the DF of the VNF instance.
- Changing the external connectivity of the VNF instance.
- Termination of the VNF instance.
- Modification of VNF instance information and/or VNF configurable properties explicitly through Modify VNF Information operation.
- Create a VNF Snapshot.
- Revert to a VNF Snapshot.
- Change of current VNF Package.

If this is a notification about the start of an LCM operation occurrence, the notification shall be sent before any action (including sending the grant request) is taken, however, after acknowledging the LCM operation request to the consumer.

If this is a notification about the result of an LCM operation occurrence, the notification shall be sent after all other actions of the LCM operation have been executed.

### 8.6.2.3 Attributes

The VnfLcmOperationOccurrenceNotification shall follow the indications provided in table 8.6.2.3-1.

**Table 8.6.2.3-1: Attributes of the VnfLcmOperationOccurrenceNotification**

| Attribute                      | Qualifier | Cardinality | Content             | Description  |
|--------------------------------|-----------|-------------|---------------------|--|
| status                         | M         | 1           | Enum                | Indicates whether this notification reports about the start of a lifecycle management operation occurrence or the result of a lifecycle management operation occurrence.<br>VALUES: <ul style="list-style-type: none"> <li>START: Informs about the start of the VNF LCM operation occurrence</li> <li>RESULT: Informs about the final or intermediate result of the VNF LCM operation occurrence</li> </ul> |
| vnfInstanceId                  | M         | 1           | Identifier          | The identifier of the VNF instance affected.   |
| operation                      | M         | 1           | String              | The lifecycle management operation.  |
| isAutomaticInvocation          | M         | 1           | Boolean             | Set to true if this VNF LCM operation occurrence has been triggered by an automated procedure inside the VNFM (i.e. ScaleVnf / ScaleVnfToLevel triggered by auto-scale, or HealVnf triggered by auto-heal).<br><br>Set to false otherwise.   |
| lifecycleOperationOccurrenceId | M         | 1           | Identifier          | The identifier of the VNF lifecycle management operation occurrence associated to the notification.  |
| affectedVnfc                   | M         | 0..N        | AffectedVnfc        | Information about VNFC instances that were affected during the execution of the lifecycle management operation, if this notification represents the result of a lifecycle management operation occurrence.   |
| affectedVirtualLink            | M         | 0..N        | AffectedVirtualLink | Information about VL instances that were affected during the execution of the lifecycle management operation, if this notification represents the result of a lifecycle management operation occurrence.   |

| Attribute  | Qualifier | Cardinality | Content                | Description   |
|--|-----------|-------------|------------------------|---|
| affectedVirtualStorage   | M         | 0..N        | AffectedVirtualStorage | Information about virtualised storage instances that were affected during the execution of the lifecycle management operation, if this notification represents the result of a lifecycle management operation occurrence. |
| changedInfo  | M         | 0..1        | Not specified          | Information about the changed VNF information, including changed VNF configurable properties, if this notification represents the result of a lifecycle management operation occurrence.                                  |
| changedExtConnectivity   | M         | 0..N        | ExtVirtualLinkInfo     | Information about changed external connectivity, if this notification represents the result of a lifecycle management operation occurrence.   |
| NOTE: If this notification represents the result of a lifecycle management operation occurrence that was not successful, the notification shall contain appropriate error information. |           |             |                        |   |

### 8.6.3 AffectedVnfc information element

#### 8.6.3.1 Description

This information element provides information about added, deleted, modified and temporary VNFCs.

#### 8.6.3.2 Attributes

The AffectedVnfc information element shall follow the indications provided in table 8.6.3.2-1.

**Table 8.6.3.2-1: Attributes of the AffectedVnfc information element**

| Attribute       | Qualifier | Cardinality | Content  | Description  |
|-----------------|-----------|-------------|--|--|
| vnfcInstancelId | M         | 1           | Identifier<br>(Reference to<br>VnfcResourceInfo) | Identifier of the VNFC instance.   |
| vduld           | M         | 1           | Identifier<br>(Reference to Vdu)                 | Identifier of the VDU in the VNFD.   |
| vnfdId          | M         | 0..1        | Identifier<br>(Reference to<br>Vnfd)             | Reference to the VNFD.<br><br>Shall be present in case of a "change current VNF Package" to identify whether the affected VNFC instance is associated to a VDU which is referred from the source or destination VNFD.  |
| changeType      | M         | 1           | Enum   | Signals the type of change.<br>VALUES:<br><ul style="list-style-type: none"> <li>• ADDED</li> <li>• REMOVED</li> <li>• MODIFIED</li> <li>• TEMPORARY</li> </ul><br>For a temporary resource, an AffectedVnfc IE exists as long as the temporary resource exists. |

| Attribute                 | Qualifier | Cardinality | Content   | Description  |
|---------------------------|-----------|-------------|---|--|
| computeResource           | M         | 1           | ResourceHandle  | Reference to the VirtualCompute resource.<br>Detailed information is (for new and modified resources) or has been (for removed resources) available from the VIM.  |
| metadata                  | M         | 0..N        | KeyValuePair  | Metadata about this resource.<br><br>The content of this attribute shall be a copy of the content of the "metadata" attribute of the VnfcResourceInfo information element.   |
| affectedVnfcCpInstances   | M         | 0..N        | Identifier<br>(Reference to VnfcCpInfo)                 | Identifiers of CP(s) of the VNFC instance that were affected by the change.<br><br>Shall be present for those affected CPs of the VNFC instance that are associated to an external CP of the VNF instance.<br><br>May be present for further affected CPs of the VNFC instance.                                    |
| addedStorageResourceIds   | M         | 0..N        | Identifier<br>(Reference to VirtualStorageResourceInfo) | Reference(s) to VirtualStorage resource(s) that were added.<br><br>Each value refers to a VirtualStorageResourceInfo item in the VnfInfo that was added to the VNFC.<br><br>It shall be provided if at least one storage resource was added to the VNFC.   |
| removedStorageResourceIds | M         | 0..N        | Identifier<br>(Reference to VirtualStorageResourceInfo) | Reference(s) to VirtualStorage resource(s) that were removed.<br><br>The value contains the identifier of a VirtualStorageResourceInfo item that has been removed from the VNFC, and might no longer exist in the VnfInfo.<br><br>It shall be provided if at least one storage resource was removed from the VNFC. |

## 8.6.4 AffectedVirtualLink information element

### 8.6.4.1 Description

This information element provides information about added, deleted, modified and temporary VLS, as well as about link port changes.

### 8.6.4.2 Attributes

The AffectedVirtualLink information element shall follow the indications provided in table 8.6.4.2-1.

Table 8.6.4.2-1: Attributes of the AffectedVirtualLink information element

| Attribute             | Qualifier | Cardinality | Content  | Description   |
|-----------------------|-----------|-------------|--|---|
| virtualLinkInstanceId | M         | 1           | Identifier (Reference to VnfVirtualLinkResourceInfo) | Identifier of the VL instance.  |
| vnfVirtualLinkDescId  | M         | 1           | Identifier (Reference to VnfVirtualLinkDesc)         | Identifier of the VLD in the VNFD.  |
| vnfdId                | M         | 0..1        | Identifier (Reference to Vnfd)                       | Reference to the VNFD.<br><br>Shall be present in case of a "change current VNF Package" to identify whether the affected VL instance is associated to a VLD which is referred from the source or destination VNFD.   |
| changeType            | M         | 1           | Enum   | Signals the type of change including, not limited to, changes made to the characteristics of the existing VL, new VL added, existing VL removed, temporary VL exists, link port added, link port removed.<br>VALUES: <ul style="list-style-type: none"> <li>• ADDED</li> <li>• REMOVED</li> <li>• MODIFIED</li> <li>• TEMPORARY</li> <li>• LINK_PORT_ADDED</li> <li>• LINK_PORT_REMOVED</li> <li>• etc.</li> </ul> For a temporary resource, an AffectedVirtualLink IE exists as long as the temporary resource exists. |
| networkResource       | M         | 1           | ResourceHandle                                       | Reference to the VirtualNetwork resource.<br><br>Detailed information is (for new and modified resources) or has been (for removed resources) available from the VIM.   |
| metadata              | M         | 0..N        | KeyValuePair   | Metadata about this resource.<br><br>The content of this attribute shall be a copy of the content of the "metadata" attribute of the VnfVirtualLinkResourceInfo information element.  |

## 8.6.5 AffectedVirtualStorage information element

### 8.6.5.1 Description

This information element provides information about added, deleted, modified and temporary virtual storage resources.

### 8.6.5.2 Attributes

The AffectedVirtualStorage information element shall follow the indications provided in table 8.6.5.2-1.

**Table 8.6.5.2-1: Attributes of the AffectedVirtualStorage information element**

| Attribute                 | Qualifier | Cardinality | Content  | Description   |
|---------------------------|-----------|-------------|--|---|
| virtualStorageInstancelId | M         | 1           | Identifier (Reference to VirtualStorageResourceInfo) | Identifier of the virtual storage instance.   |
| virtualStorageDescId      | M         | 1           | Identifier (Reference to VirtualStorageDesc)         | Identifier of the VirtualStorageDesc in the VNFD.   |
| vnfdId                    | M         | 0..1        | Identifier (Reference to Vnfd)                       | Reference to the VNFD.<br><br>Shall be present in case of a "change current VNF Package" to identify whether the affected virtual storage instance is associated to a VirtualStorageDesc which is referred from the source or destination VNFD.                         |
| changeType                | M         | 1           | Enum   | Signals the type of change.<br>VALUES:<br><ul style="list-style-type: none"> <li>• ADDED</li> <li>• REMOVED</li> <li>• MODIFIED</li> <li>• TEMPORARY</li> </ul> For a temporary resource, an AffectedVirtualStorage IE exists as long as the temporary resource exists. |
| storageResource           | M         | 1           | ResourceHandle                                       | Reference to the VirtualStorage resource.<br>Detailed information is (for new and modified resources) or has been (for removed resources) available from the VIM.   |
| metadata                  | M         | 0..N        | KeyValuePair   | Metadata about this resource.<br><br>The content of this attribute shall be a copy of the content of the "metadata" attribute of the VirtualStorageResourceInfo information element.  |

## 8.6.6 Void

## 8.6.7 VnfIdentifierCreationNotification

### 8.6.7.1 Description

This notification informs the receiver of the creation of a new VNF instance identifier and the associated instance of a VnfInfo information element, identified by that identifier. The support of the notification is mandatory.

### 8.6.7.2 Trigger conditions

- Creation of a VNF instance identifier and the associated instance of a VnfInfo information element.

### 8.6.7.3 Attributes

The VnfIdentifierCreationNotification shall follow the indications provided in table 8.6.7.3-1.

**Table 8.6.7.3-1: Attributes of the VnfIdentifierCreationNotification**

| Attribute      | Qualifier | Cardinality | Content    | Description                                |
|----------------|-----------|-------------|------------|--|
| vnfInstancelId | M         | 1           | Identifier | The newly created VNF instance identifier. |

## 8.6.8 VnfIdentifierDeletionNotification

### 8.6.8.1 Description

This notification informs the receiver of the deletion of a VNF instance identifier and the associated instance of a VnfInfo information element identified by that identifier. The support of the notification is mandatory.

### 8.6.8.2 Trigger conditions

- Deletion of a VNF instance identifier and the associated instance of a VnfInfo information element.

### 8.6.8.3 Attributes

The VnfIdentifierDeletionNotification shall follow the indications provided in table 8.6.8.3-1.

**Table 8.6.8.3-1: Attributes of the VnfIdentifierDeletionNotification**

| Attribute     | Qualifier | Cardinality | Content    | Description  |
|---------------|-----------|-------------|------------|--|
| vnfInstanceId | M         | 1           | Identifier | The VNF instance identifier that has been deleted. |

## 8.7 Information elements and notifications related to VNF Performance Management

### 8.7.1 Introduction

This clause defines information elements and notifications related to VNF Performance Management.

### 8.7.2 ObjectSelection information element

#### 8.7.2.1 Description

This information element allows to specify VNF 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 VNF 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 the VNF related measured object types (see note).

**NOTE:** The VNF related measured object types are the measured object type(s) for which the performance measurements applicable to Or-Vnfm reference point are defined in clause 7.2 of ETSI GS NFV-IFA 027 [5].

#### 8.7.2.2 Attributes

The ObjectSelection information element shall follow the indications provided in table 8.7.2.2-1.

**Table 8.7.2.2-1: Attributes of the ObjectSelection information element**

| Attribute         | Qualifier | Cardinality | Content    | Description   |
|-------------------|-----------|-------------|------------|---|
| objectType        | M         | 0..N        | String     | Defines the measured object types.<br>The object types for this information element will be the VNF related measured object types.<br>One of the two attributes (objectType + objectFilter or objectInstancelid) 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.<br>One of the two attributes (objectType + objectFilter or objectInstancelid) shall be present.  |
| objectInstancelid | M         | 0..N        | Identifier | Identifies the object instances for which performance information is requested to be collected.<br>The object instances for this information element will be instances corresponding to the VNF related measured object types.<br>One of the two attributes (objectType+ objectFilter or objectInstancelid) shall be present. |

## 8.7.3 PmJob information element

### 8.7.3.1 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 VNF related measured object types.

### 8.7.3.2 Attributes

The PmJob information element shall follow the indications provided in table 8.7.3.2-1.

**Table 8.7.3.2-1: Attributes of the PmJob information element**

| Attribute              | Qualifier | Cardinality | Content         | Description  |
|------------------------|-----------|-------------|-----------------|--|
| pmJobId                | M         | 1           | Identifier      | Identifier of this PM job.   |
| objectSelector         | M         | 1           | ObjectSelection | Defines the object instances for which performance information is requested to be collected.<br>The object instances for this information element will be instances corresponding to the VNF related measured object types.  |
| performanceMetric      | M         | 0..N        | String          | This defines the type(s) of performance metric(s) for the specified object instances.<br>At least one of the two attributes (performance metric or group) shall be present.  |
| performanceMetricGroup | M         | 0..N        | String          | Group of performance metrics.<br>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 Name" values of the performance measurements applicable to Or-Vnfm reference point, as defined in clause 7.2 of ETSI GS NFV-IFA 027 [5].<br>At least one of the two attributes (performance metric or group) shall be present. |
| collectionPeriod       | M         | 1           | Not specified   | Specifies the periodicity at which the producer will collect performance information (see note).   |



| Attribute  | Qualifier | Cardinality | Content       | Description   |
|--|-----------|-------------|---------------|---|
| 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.<br>The boundary shall allow a single reporting as well as periodic reporting up to the boundary. |
| 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 reportingPeriod 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. |           |             |               |   |

## 8.7.4 Threshold information element

### 8.7.4.1 Description

This information element provides the details of a threshold. The object instances for this information element will be the instances corresponding to the VNF related measured object types.

### 8.7.4.2 Attributes

The Threshold information element shall follow the indications provided in table 8.7.4.2-1.

**Table 8.7.4.2-1: 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.<br>The object instances for this information element will be instances corresponding to the VNF 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 Or-Vnfm reference point, as defined in clause 7.2 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.<br>VALUES: <ul style="list-style-type: none"> <li>• SIMPLE: Single-valued static threshold</li> <li>• Etc.</li> </ul> |
| thresholdDetails  | M         | 1           | Not specified   | Details of the threshold: value to be crossed, details on the notification to be generated, etc.   |

## 8.7.5 PerformanceReport information element

### 8.7.5.1 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 VNF related measured object types.

### 8.7.5.2 Attributes

The PerformanceReport information element shall follow the indications provided in table 8.7.5.2-1.

**Table 8.7.5.2-1: Attributes of the PerformanceReport information element**

| Attribute         | Qualifier | Cardinality | Content                | Description                              |
|-------------------|-----------|-------------|------------------------|--|
| performanceReport | M         | 1..N        | PerformanceReportEntry | List of performance information entries. |

## 8.7.6 PerformanceReportEntry information element

### 8.7.6.1 Description

This information element defines a single performance report entry. This performance report entry is for a given metric of a given object instance, but can include multiple collected values. The object instances for this information element will be the instances corresponding to the VNF related measured object types.

### 8.7.6.2 Attributes

The PerformanceReportEntry information element shall follow the indications provided in table 8.7.6.2-1.

**Table 8.7.6.2-1: 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 VNF 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 VNF related measured object types.                               |
| performanceMetric | M         | 1           | String                | Name of the metric collected. This attribute's value contains the related "Measurement Name" values of the performance measurements applicable to Or-Vnfm reference point, as defined in clause 7.2 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]).   |

## 8.7.7 PerformanceValueEntry information element

### 8.7.7.1 Description

This information element defines a single performance value with its associated time stamp and measurement context (see ETSI GS NFV-IFA 027 [5]).

### 8.7.7.2 Attributes

The PerformanceValueEntry information element shall follow the indications provided in table 8.7.7.2-1.

**Table 8.7.7.2-1: 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" for the performance measurements applicable to Or-Vnfm reference point, as defined in clause 7.2 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].  |

## 8.7.8 PerformanceInformationAvailableNotification

### 8.7.8.1 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 VNF related measured object types.

### 8.7.8.2 Trigger Conditions

- New performance information is available.

### 8.7.8.3 Attributes

The PerformanceInformationAvailableNotification shall follow the indications provided in table 8.7.8.3-1.

**Table 8.7.8.3-1: Attributes of the PerformanceInformationAvailableNotification**

| Attribute        | Qualifier | Cardinality | Content    | Description  |
|------------------|-----------|-------------|------------|--|
| objectInstanceId | M         | 1..N        | Identifier | Object instance(s) for which performance information is available. The object instances for this information element will be instances corresponding to the VNF related measured object types. |

## 8.7.9 ThresholdCrossedNotification

### 8.7.9.1 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 VNF related measured object types.

### 8.7.9.2 Trigger Condition

A Threshold has been crossed. Depending on threshold type, there might be a single or multiple crossing values.

### 8.7.9.3 Attributes

The ThresholdCrossedNotification shall follow the indications provided in table 8.7.9.3-1.

Table 8.7.9.3-1: 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: <ul style="list-style-type: none"> <li>• UP</li> <li>• DOWN</li> </ul>  |
| objectInstanceId   | M         | 1           | Identifier                          | Object instance for which the threshold has been crossed. The object instances for this information element will be instances corresponding to the VNF related measured object types.   |
| performanceMetric  | M         | 1           | String                              | Performance metric associated with the threshold. This attribute's value contains the related "Measurement Name" values of the performance measurements applicable to Or-Vnfm reference point, as defined in clause 7.2 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].   |

## 8.8 Information elements and notifications related to VNF Fault Management

### 8.8.1 Introduction

This clause defines information elements and notifications related to VNF Fault Management.

### 8.8.2 AlarmNotification

#### 8.8.2.1 Description

This notification informs the receiver of alarms related to the VNFs managed by the VNFM. Alarms are created in response to:

- faults detected by the VNFM; and
- faults generated due to changes in the state of virtualised resources used by the VNF instances managed by the VNFM.

The notification is mandatory.

#### 8.8.2.2 Trigger conditions

- An alarm has been created.
- An alarm has been updated, e.g. if the severity of the alarm has changed.

#### 8.8.2.3 Attributes

The AlarmNotification shall follow the indications provided in table 8.8.2.3-1.

**Table 8.8.2.3-1: Attributes of the AlarmNotification**

| Attribute | Qualifier | Cardinality | Content | Description  |
|-----------|-----------|-------------|---------|--|
| alarm     | M         | 1           | Alarm   | Information about an alarm including AlarmId, affected VNF identifier, and FaultDetails.<br>For notifications related to changes in the state of virtualised resources (indicated using the attribute faultType), the alarm shall indicate: <ul style="list-style-type: none"> <li>The cause for the state change of the virtualised resource using the attribute probableCause, with possible values such as: maintenance of NFVI component, evacuation of NFVI component, etc.</li> <li>The identifier of the origin (VIM) responsible for the management of the virtualised resource with state change using the attribute faultDetails.</li> </ul> |

## 8.8.3 AlarmClearedNotification

### 8.8.3.1 Description

This notification informs the receiver of the clearing of an alarm related to the VNFs managed by the VNFM, e.g. the alarm's perceived severity is set to "cleared" since the corresponding fault has been solved. The notification is mandatory.

### 8.8.3.2 Trigger conditions

- An alarm has been cleared.

### 8.8.3.3 Attributes

The AlarmClearedNotification shall follow the indications provided in table 8.8.3.3-1.

**Table 8.8.3.3-1: Attributes of the AlarmClearedNotification**

| Attribute        | Qualifier | Cardinality | Content                         | Description  |
|------------------|-----------|-------------|---------------------------------|--|
| alarmId          | M         | 1           | Identifier (Reference to Alarm) | Alarm identifier.                                    |
| alarmClearedTime | M         | 1           | DateTime                        | The timestamp indicating when the alarm was cleared. |

## 8.8.4 Alarm information element

### 8.8.4.1 Description

The Alarm information element encapsulates information about an alarm.

The Managed Objects for this information element will be VNF instances.

### 8.8.4.2 Attributes

The Alarm information element shall follow the indications provided in table 8.8.4.2-1.

Table 8.8.4.2-1: 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.<br><br>The managed objects for this information element will be VNF instances.   |
| rootCauseFaultyResource | M         | 0..1        | FaultyResourceInfo                 | The virtualised resources that are causing the VNF fault. Shall be present if the alarm affects virtualised resources.  |
| alarmRaisedTime         | M         | 1           | DateTime                           | Timestamp indicating when the alarm is raised by the managed object.  |
| alarmChangedTime        | M         | 0..1        | DateTime                           | Timestamp indicating when the alarm was last changed. It shall be present if the alarm has been updated.  |
| alarmClearedTime        | M         | 0..1        | DateTime                           | Timestamp indicating when the alarm was cleared. It shall be present if the alarm has been cleared.   |
| ackState                | M         | 1           | Enum                               | State of the alarm.<br>VALUES:<br><ul style="list-style-type: none"> <li>• ACKNOWLEDGED</li> <li>• UNACKNOWLEDGED</li> </ul>  |
| perceivedSeverity       | M         | 1           | Enum                               | Perceived severity of the managed object failure.<br>VALUES:<br><ul style="list-style-type: none"> <li>• CRITICAL</li> <li>• MAJOR</li> <li>• MINOR</li> <li>• WARNING</li> <li>• INDETERMINATE</li> <li>• CLEARED</li> </ul>   |
| eventTime               | M         | 1           | DateTime                           | Timestamp indicating when the fault was observed.   |
| eventType               | M         | 1           | Enum                               | Type of the event. The values for the eventType attribute use the event type defined in Recommendation ITU-T X.733 [4].<br>VALUES:<br><ul style="list-style-type: none"> <li>• COMMUNICATIONS_ALARM</li> <li>• PROCESSING_ERROR_ALARM</li> <li>• ENVIRONMENTAL_ALARM</li> <li>• QOS_ALARM</li> <li>• EQUIPMENT_ALARM</li> </ul> |
| 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                            | Attribute indicating if this fault is the root for other correlated alarms. If TRUE, then the alarms listed in the attribute CorrelatedAlarmId are caused by this fault.  |
| correlatedAlarmId       | M         | 0..N        | Identifier<br>(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.  |

## 8.8.5 FaultyResourceInfo information element

### 8.8.5.1 Description

The FaultyResourceInfo information element encapsulates information about faulty resource that has a negative impact on a VNF.

### 8.8.5.2 Attributes

The FaultyResourceInfo information element shall follow the indications provided in table 8.8.5.2-1.

**Table 8.8.5.2-1: 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.<br>See clause 8.5.7.                                    |
| faultyResourceType | M         | 1           | Enum           | Type of the faulty resource.<br>VALUES: <ul style="list-style-type: none"> <li>• COMPUTE</li> <li>• STORAGE</li> <li>• NETWORK</li> </ul> |

## 8.8.6 AlarmListRebuiltNotification

### 8.8.6.1 Description

This notification informs the receiver that the active alarm list has been rebuilt by the VNFM. 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 VNFM.

The notification is mandatory.

### 8.8.6.2 Trigger conditions

- Active alarm list has been rebuilt by the VNFM, e.g. if the VNFM detects its storage holding the alarm list is corrupted.

### 8.8.6.3 Attributes

The AlarmListRebuiltNotification does not contain any attributes.

## 8.9 Void

## 8.10 Information elements and notifications related to VNF Indicators

### 8.10.1 Introduction

The clauses below define information elements which represent indicator values, and notifications about changes of these.

### 8.10.2 IndicatorValueChangeNotification

#### 8.10.2.1 Description

This notification informs the receiver of a value change of an indicator related to the VNF. The notification is mandatory.

#### 8.10.2.2 Trigger conditions

- The value of an indicator has changed.

### 8.10.2.3 Attributes

The IndicatorValueChangeNotification information element shall follow the indications provided in table 8.10.2.3-1.

**Table 8.10.2.3-1: Attributes of the IndicatorValueChangeNotification**

| Attribute            | Qualifier | Cardinality | Content              | Description  |
|----------------------|-----------|-------------|----------------------|--|
| indicatorInformation | M         | 1           | IndicatorInformation | This is to provide the indicator, the value of the indicator and the VNF instance the indicator is related to. |

## 8.10.3 IndicatorInformation information element

### 8.10.3.1 Description

This information element provides the indicator values of a VNF instance.

### 8.10.3.2 Attributes

The IndicatorInformation information element shall follow the indications provided in table 8.10.3.2-1.

**Table 8.10.3.2-1: Attributes of the IndicatorInformation information element**

| Attribute      | Qualifier | Cardinality | Content                                | Description   |
|----------------|-----------|-------------|--|---|
| vnfInstanceId  | M         | 1           | Identifier                             | Identifies the VNF instance which provides the indicator value(s).  |
| indicatorId    | M         | 1           | Identifier (Reference to VnfIndicator) | Identifies the indicator.   |
| indicatorValue | M         | 1           | Value                                  | Provides the value of the indicator. The value format is defined in the VNFD (see ETSI GS NFV-IFA 011 [3]).                         |
| indicatorName  | M         | 0..1        | String                                 | Human readable name of the indicator. Shall be present if defined in the VNFD according to clause 7.1.2 of ETSI GS NFV-IFA 011 [3]. |

## 8.10.4 SupportedIndicatorsChangeNotification

### 8.10.4.1 Description

This notification informs the receiver that the set of indicators supported by a VNF instance has changed. Such change can occur as a side effect of the "Change current VNF package" operation.

### 8.10.4.2 Trigger conditions

- The set of indicators supported by a VNF instance has changed.

### 8.10.4.3 Attributes

The SupportedIndicatorsChangeNotification information element shall follow the indications provided in table 8.10.4.3-1.

**Table 8.10.4.3-1: Attributes of the SupportedIndicatorsChangeNotification**

| Attribute          | Qualifier | Cardinality | Content                           | Description  |
|--------------------|-----------|-------------|-----------------------------------|--|
| vnfInstanceId      | M         | 1           | Identifier (Reference to VnfInfo) | Identifies the VNF instance which provides the indicators. |
| supportedIndicator | M         | 0..N        | SupportedIndicatorInformation     | Set of VNF indicators supported by the VNF instance.       |



## 8.10.5 SupportedIndicatorInformation information element

### 8.10.5.1 Description

This information element provides information about a supported VNF indicator.

### 8.10.5.2 Attributes

The SupportedIndicatorInformation information element shall follow the indications provided in table 8.10.5.2-1.

**Table 8.10.5.2-1: Attributes of the SupportedIndicatorInformation information element**

| Attribute     | Qualifier | Cardinality | Content                                | Description   |
|---------------|-----------|-------------|--|---|
| indicatorId   | M         | 1           | Identifier (Reference to VnfIndicator) | Identifies the indicator.   |
| indicatorName | M         | 0..1        | String                                 | Human readable name of the indicator. Shall be present if defined in the VNFD according to clause 7.1.2 of ETSI GS NFV-IFA 011 [3]. |

## 8.11 Notifications related to Virtualised Resources Quota

### 8.11.1 Introduction

This clause defines notifications related to virtualised resources quota.

### 8.11.2 VirtualisedResourceQuotaAvailableNotification

#### 8.11.2.1 Description

This notification indicates the availability of a quota applicable to the consumer. Support of this notification is mandatory if the Virtualised Resources Quota Available Notification interface is supported.

#### 8.11.2.2 Trigger Conditions

- A virtualised resources quota applicable to the consumer has been set.

#### 8.11.2.3 Attributes

The VirtualisedResourceQuotaAvailableNotification shall follow the indications provided in table 8.11.2.3-1.

**Table 8.11.2.3-1: Attributes of the VirtualisedResourceQuotaAvailableNotification**

| Attribute          | Qualifier | Cardinality | Content           | Description  |
|--------------------|-----------|-------------|-------------------|--|
| resourceGroupId    | M         | 1           | Identifier        | Unique identifier of the "infrastructure resource group", logical grouping of virtual resources assigned to a tenant within an Infrastructure Domain.  |
| vimConnectionInfo  | CM        | 0..1        | VimConnectionInfo | Information about the VIM connection to manage the virtualised resources quota.<br><br>This attribute shall be supported when VNF-related Resource Management in direct mode is applicable.            |
| resourceProviderId | CM        | 0..1        | Identifier        | Identifies the entity responsible for the management of the virtualised resources quota.<br><br>This attribute shall be supported when VNF-related Resource Management in indirect mode is applicable. |

## 8.12 Information elements and notifications related to multiple interfaces

### 8.12.1 Introduction

This clause defines information elements that are referenced by other information elements related to multiple interfaces.

### 8.12.2 ExtVirtualLinkData information element

#### 8.12.2.1 Description

This information element provides the information of an external VL to be used as a parameter passed to multiple interfaces.

#### 8.12.2.2 Attributes

The ExtVirtualLinkData information element shall follow the indications provided in table 8.12.2.2-1.

**Table 8.12.2.2-1: 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.   |
| vimConnectionId    | CM        | 0..1        | Identifier<br>(Reference to<br>VimConnectionInfo) | Identifier of the VIM connection to manage this resource.<br>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.<br><br>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. If this attribute is not present, the VNFM shall create the link ports on the external VL. |

### 8.12.2a ExtLinkPortData information element

#### 8.12.2a.1 Description

This information element represents an externally provided link port to be used to connect an external connection point to an external VL.

#### 8.12.2a.2 Attributes

The ExtLinkPortData information element shall follow the indications provided in table 8.12.2a.2-1.

**Table 8.12.2a.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.      |

### 8.12.3 VnfExtCpData information element

#### 8.12.3.1 Description

This information element provides input information related to one or more external CP instances created based on the same CPD.

#### 8.12.3.2 Attributes

The VnfExtCpData information element shall follow the indications provided in table 8.12.3.2-1.

**Table 8.12.3.2-1: 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. |

### 8.12.3a VnfExtCpConfig information element

#### 8.12.3a.1 Description

This information element represents an externally provided link port or network address information per instance of an external connection point. In case a link port is provided, the VNFM shall use that link port when connecting the external CP to the external VL. In case no link port is provided, the VNFM shall create a link port on the external VL, and use that link port to connect the external CP to the external VL.

#### 8.12.3a.2 Attributes

The VnfExtCpConfig information element shall follow the indications provided in table 8.12.3a.2-1.

**Table 8.12.3a.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.<br><br>Shall be present if this instance has already been created. |
| linkPortId   | M         | 0..1        | Identifier<br>(Reference to<br>ExtLinkPortData) | Identifier of a pre-configured link port to which the external CP will be associated. See note.   |

| Attribute  | Qualifier | Cardinality | Content       | Description  |
|--|-----------|-------------|---------------|--|
| 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).<br><br>For dynamic addresses, it should be possible to define per parameter set the number of network addresses to be assigned dynamically.<br><br>Other parameters could be, e.g. valid address ranges or subnets.<br>See note. |
| <p>NOTE: The following conditions apply to the attributes "linkPortId" and "cpProtocolData":</p> <ol style="list-style-type: none"> <li>1) The "linkPortId" and "cpProtocolData" attributes shall both be absent for the deletion of an existing external CP instance addressed by cpInstanceId.</li> <li>2) At least one of these attributes shall be present for a to-be-created external CP instance or an existing external CP instance.</li> <li>3) If the "linkPortId" attribute is absent, the VNFM shall create a link port.</li> <li>4) If the "cpProtocolData" attribute is absent, the "linkPortId" attribute shall be provided referencing a pre-created link port, and the VNFM can use means outside the scope of the present document to obtain the pre-configured address information for the connection point from the resource representing the link port.</li> <li>5) If both "cpProtocolData" and "linkportId" are provided, the NFVO shall ensure that the cpProtocolData can be used with the pre-created link port referenced by "linkPortId".</li> </ol> |           |             |               |  |

## 8.12.4 ExtManagedVirtualLinkData information element

### 8.12.4.1 Description

This information element provides the information of an externally-managed internal VL to be used as a parameter passed to multiple interfaces.

### 8.12.4.2 Attributes

The ExtManagedVirtualLinkData information element shall follow the indications provided in table 8.12.4.2-1.

**Table 8.12.4.2-1: 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.  |
| vimConnectionId         | CM        | 0..1        | Identifier (Reference to VimConnectionInfo)  | Identifier of the VIM connection to manage this resource.<br>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.<br><br>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.  |

| Attribute                        | Qualifier | Cardinality | Content         | Description   |
|----------------------------------|-----------|-------------|-----------------|---|
| 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 virtualLinkId shall refer to the same extManagedMultisiteVirtualLinkId. |

## 8.12.5 VimConnectionInfo information element

### 8.12.5.1 Description

This information element provides information regarding a VIM connection.

It is assumed that during the protocol design stage, VimConnectionInfo will be specified such that it allows interfacing to different VIM types.

### 8.12.5.2 Attributes

The VimConnectionInfo information element shall follow the indications provided in table 8.12.5.2-1.

**Table 8.12.5.2-1: Attributes of the VimConnectionInfo information element**

| Attribute   | Qualifier | Cardinality | Content       | Description  |
|---|-----------|-------------|---------------|--|
| vimConnectionInfoId   | M         | 1           | Identifier    | The identifier of this VimConnectionInfo information element, for the purpose of referencing it from other information elements.<br>This identifier is managed by the NFVO.  |
| vimId   | M         | 0..1        | Identifier    | The identifier of the VIM. This identifier is managed by the NFVO.<br><br>Shall be present to address additional information about the VIM if such information has been configured into the VNFM by means outside the scope of the present document, and should be absent otherwise. |
| interfaceInfo   | M         | 0..N        | Not specified | Information about the interface(s) to the VIM, if available, including interface endpoint e.g. URL API version, and protocol type.<br>Alternatively, such information may have been configured into the VNFM and bound to the vimId.   |
| accessInfo  | M         | 0..N        | Not specified | Authentication credentials for accessing the VIM. Examples can include those to support different authentication schemes, e.g. OAuth, Token, Username/password, etc. See note.   |
| extra   | M         | 0..N        | Not specified | VIM type specific additional information, if applicable.   |
| NOTE: If needed, this attribute also provides information about the resourceGroupIds that are accessible using a particular set of credentials. |           |             |               |  |

## 8.12.6 VnfLinkPortData information element

### 8.12.6.1 Description

This information element represents an externally provided link port to be used to connect a VNFC connection point to an externally-managed VL.

### 8.12.6.2 Attributes

The VnfLinkPortData information element shall follow the indications provided in table 8.12.6.2-1.

**Table 8.12.6.2-1: 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.               |

## 8.13 Information elements and notifications related to Policy Management

### 8.13.1 Introduction

The clauses below define information elements and notifications related to policy management.

### 8.13.2 Information elements related to Policy Management Operations

#### 8.13.2.1 Introduction

The clauses below define information elements related to policy management operations.

#### 8.13.2.2 PolicyInfo information element

##### 8.13.2.2.1 Description

This information element provides policy related information. It contains the policy itself and additional information related to the policy.

##### 8.13.2.2.2 Attributes

The structure of the PolicyInfo information element shall comply with the provisions for the PolicyInfo information element as defined in ETSI GS NFV-IFA 013 [i.8], clause 8.8.2.2.2.

### 8.13.3 PolicyChangeNotification

#### 8.13.3.1 Description

This notification indicates a change of a NFV-MANO policy related to operations of transferring policy, deleting policy, activating policy, deactivating policy, associating policy and disassociating policy.

Support of this notification is mandatory.

### 8.13.3.2 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.

### 8.13.3.3 Attributes

The PolicyChangeNotification shall comply with the provisions in clause 8.8.3.3 of ETSI GS NFV-IFA 013 [i.8].

## 8.13.4 PolicyConflictNotification

### 8.13.4.1 Description

This notification indicates a policy conflict is detected by the VNFM. A policy conflict can include any conflicted monitored events, conditions or actions among two or more policies enforced by the VNFM.

Support of this notification is mandatory.

### 8.13.4.2 Trigger Conditions

The notification is produced when a policy conflict is detected by the VNFM.

### 8.13.4.3 Attributes

The PolicyConflictNotification shall comply with the provisions in clause 8.8.4.3 of ETSI GS NFV-IFA 013 [i.8].

## 8.14 Information elements related to VNF Snapshot Package Management

### 8.14.1 Introduction

This clause defines information elements related to VNF Snapshot Package Management.

### 8.14.2 VnfSnapshotPkgInfo information element

#### 8.14.2.1 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.

#### 8.14.2.2 Attributes

The VnfSnapshotPkgInfo information element shall follow the indications provided in table 8.14.2.2-1.

**Table 8.14.2.2-1: Attributes of the VnfSnapshotPkgInfo information element**

| Attribute            | Qualifier | Cardinality | Content       | Description  |
|----------------------|-----------|-------------|---------------|--|
| vnfSnapshotPkgInfoId | M         | 1           | Identifier    | Identifier of information held by the VNFM about a specific VNF Snapshot Package. This identifier was allocated by the NFVO. |
| 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.   |

| Attribute  | Qualifier | Cardinality | Content                                    | Description  |
|--|-----------|-------------|--|--|
| createdAt  | M         | 0..1        | DateTime                                   | Timestamp indicating when the VNF Snapshot Package creation has been completed. See note 2.  |
| vnfSnapshotInfold  | M         | 0..1        | Identifier (Reference to VnfSnapshotInfo)  | References information 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.  |
| vnfcSnapshotInfold   | M         | 0..N        | Identifier (Reference to VnfcSnapshotInfo) | References information about specific VNFC Snapshot(s). These identifiers were allocated by the VNFM. See note 2.  |
| 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 Snapshot artifact(s) that are not VNFC Snapshot Images.  |
| state  | M         | 1           | Enum                                       | State of the VNF Snapshot Package.<br>VALUES: <ul style="list-style-type: none"> <li>• CREATED</li> <li>• BUILDING</li> <li>• UPLOADING</li> <li>• AVAILABLE</li> <li>• EXTRACTING</li> <li>• PROCESSING</li> <li>• ERROR</li> </ul> |
| 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.   |
| <p>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.</p> <p>NOTE 2: Cardinality is 0 when the VnfSnapshotPkgInfo was created but the VNF Snapshot Package was not yet built or uploaded.</p> |           |             |  |  |

### 8.14.3 SnapshotPkgArtifactInformation information element

#### 8.14.3.1 Description

This information element represents an artifact other than a VNFC Snapshot Image which is contained in the VNF Snapshot Package.

#### 8.14.3.2 Attributes

The SnapshotPkgArtifactInformation information element shall follow the indications provided in table 8.14.3.2-1.



**Table 8.14.3.2-1: 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. |

## 8.14.4 VnfcSnapshotImageInfo information element

### 8.14.4.1 Description

This information element represents VNFC Snapshot Image Information.

### 8.14.4.2 Attributes

The VnfcSnapshotImageInfo information element shall follow the indications provided in table 8.14.4.2-1.

**Table 8.14.4.2-1: 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. |

## 8.14.5 VnfStateSnapshotInfo information element

### 8.14.5.1 Description

This information element represents information about VNF-specific state snapshot data and where to retrieve it.

### 8.14.5.2 Attributes

The VnfStateSnapshotInfo information element shall follow the indications provided in table 8.14.5.2-1.

**Table 8.14.5.2-1: Attributes of the VnfStateSnapshotInfo information element**

| <b>Attribute</b>  | <b>Qualifier</b> | <b>Cardinality</b> | <b>Content</b> | <b>Description</b>  |
|-------------------|------------------|--------------------|----------------|---|
| accessInformation | M                | 1                  | Not specified  | Information (such as a path) that identifies/addresses this VNF state snapshot.         |
| metadata          | M                | 1                  | Not specified  | The metadata of the VNF state snapshot, such as content type, size, creation date, etc. |

---

## Annex A (informative): Examples of VNF connectivity patterns

### A.1 Introduction

This annex illustrates examples of possible connectivity patterns for a VNF. The purpose is to illustrate the relationship among the different information elements specified in clause 8.5 that are used to describe the connectivity of and within a VNF instance.

The present annex A also illustrates the use of the "Change External VNF Connectivity" operation to re-connect external CPs of a VNF instance to a different external VL.

**NOTE:** The information related to connectivity as shown in the annex A is to be understood in the context of the present document, i.e. availability of certain information on the Or-Vnfm reference point follows the conditions that are detailed in the respective attribute descriptions and notes in the present document.

---

### A.2 Example of a VNF with two different types of external connections points

The present example shows a regular connectivity pattern of a VNF where the two external CPs of the VNF use different connectivity patterns. Figure A.2-1 illustrates the example, from which it is highlighted the following:

- An external CP of the VNF instance (see VnfExtCp #1) that maps to an internal CP, i.e. a CP of a specific VNFC.
- An external CP of the VNF instance (see VnfExtCp #2) that refers to a link port of an internal VL of the VNF (see VnfLinkPort #2.2).
- An internal VL of the VNF instance (see VnfVirtualLink #1) that is only used for connectivity of VNFCs within the VNF.
- An internal VL of the VNF instance (see VnfVirtualLink #2) that is used as provider of a link port for connectivity of external CPs of the VNF.
- Link ports of internal VL(s) of the VNF instance (see VnfLinkPort #1.1 to #1.3 and VnfLinkPort #2.1) that are optionally exposed on Or-Vnfm reference point.
- Internal CPs, i.e. CPs of specific VNFCs (see grey VNFC CPs) that are optionally exposed on the Or-Vnfm reference point.

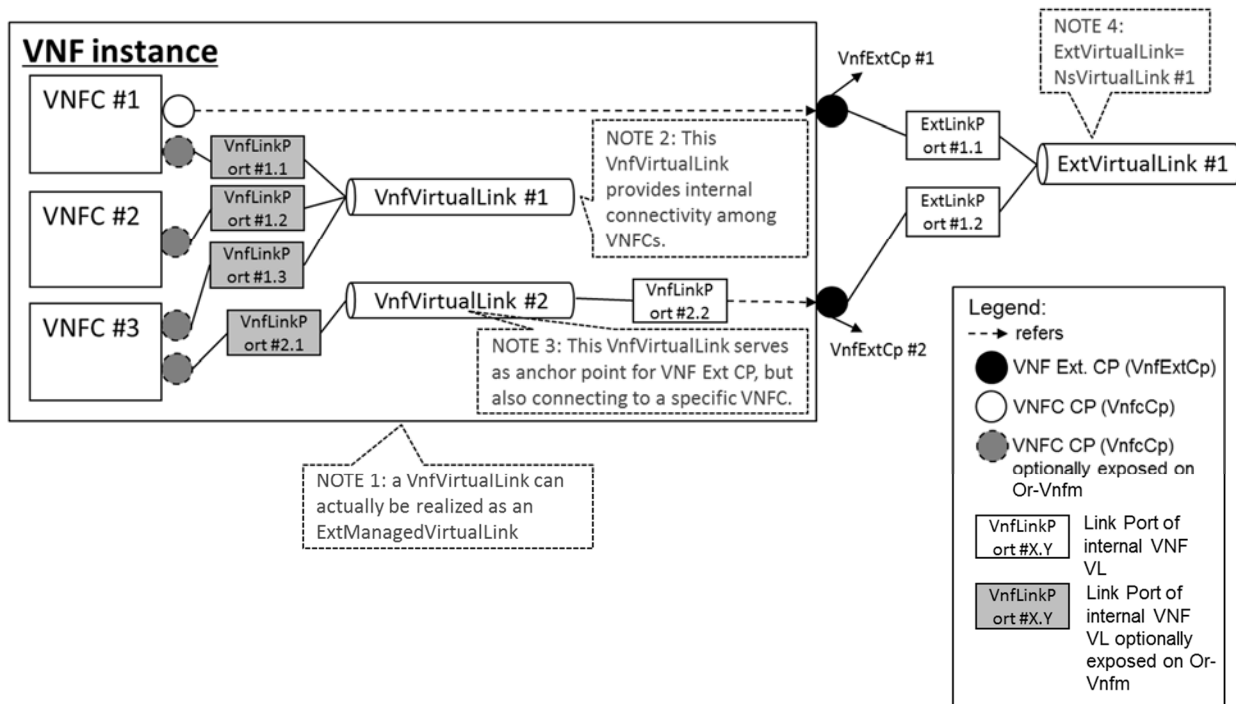


Figure A.2-1: Example of a VNF with two different types of external connection points

### A.3 Example of changing VNF connectivity

This example illustrates the operation "Change external VNF connectivity" (clause 7.2.18). The scenario depicted disconnects all external CP instances that were created based on a particular CPD from a "source" external VL and connects them to a "target" external VL.

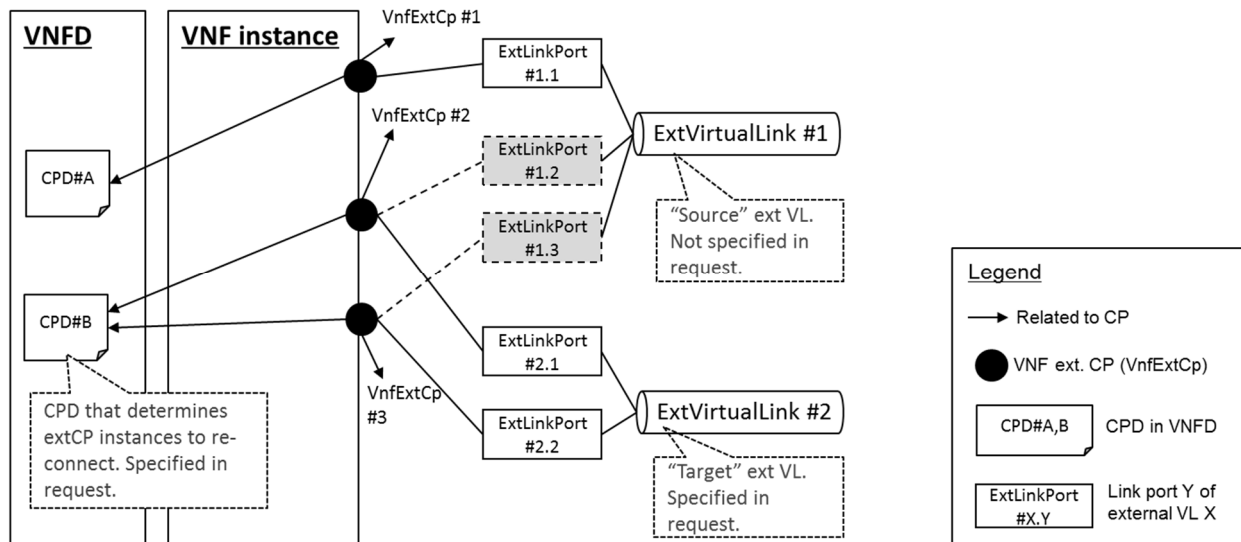


Figure A.3-1: Illustration of disconnecting external CPs from one external VL and connecting them to another external VL

---

## Annex B (informative): VNF software modification

### B.1 Introduction

The present annex describes the procedures for the modification of the VNF software, both not assisted by the NFV-MANO and assisted by the NFV-MANO via change of current VNF Package of a VNF instance according to the interfaces and operations specified in the present document.

The procedures introduced in clauses B.2 and B.3 focus primarily on the interactions concerning the VNFM and on the reference points of the VNFM with other NFV-MANO or external functional blocks. Therefore, details of interactions and interface operations over other reference points are either not detailed or summarized.

---

### B.2 VNF software modification not assisted by NFV-MANO

#### B.2.1 Description

In this type of VNF software modification, the VNF application is updated/updated by external management systems (e.g. OSS/EM) without any involvement of the NFV-MANO functional blocks. The process requires no change of the virtualised resources and resource composition of the current VNF instance. As part of the process, a new VNF Package containing the new VNF application software is made available to the NFV-MANO to be used by LCM operations after the VNF software modification.

The purpose of the NFV-MANO procedure is limited to:

- a) Ensure that the information in the NFV-MANO entities is synchronized with respect to the software modification performed externally.
- b) Ensure that the necessary VNF Package, artifacts and VIM assets are available within the NFV-MANO in order to handle any current or subsequent lifecycle management procedures on the affected VNF instance.

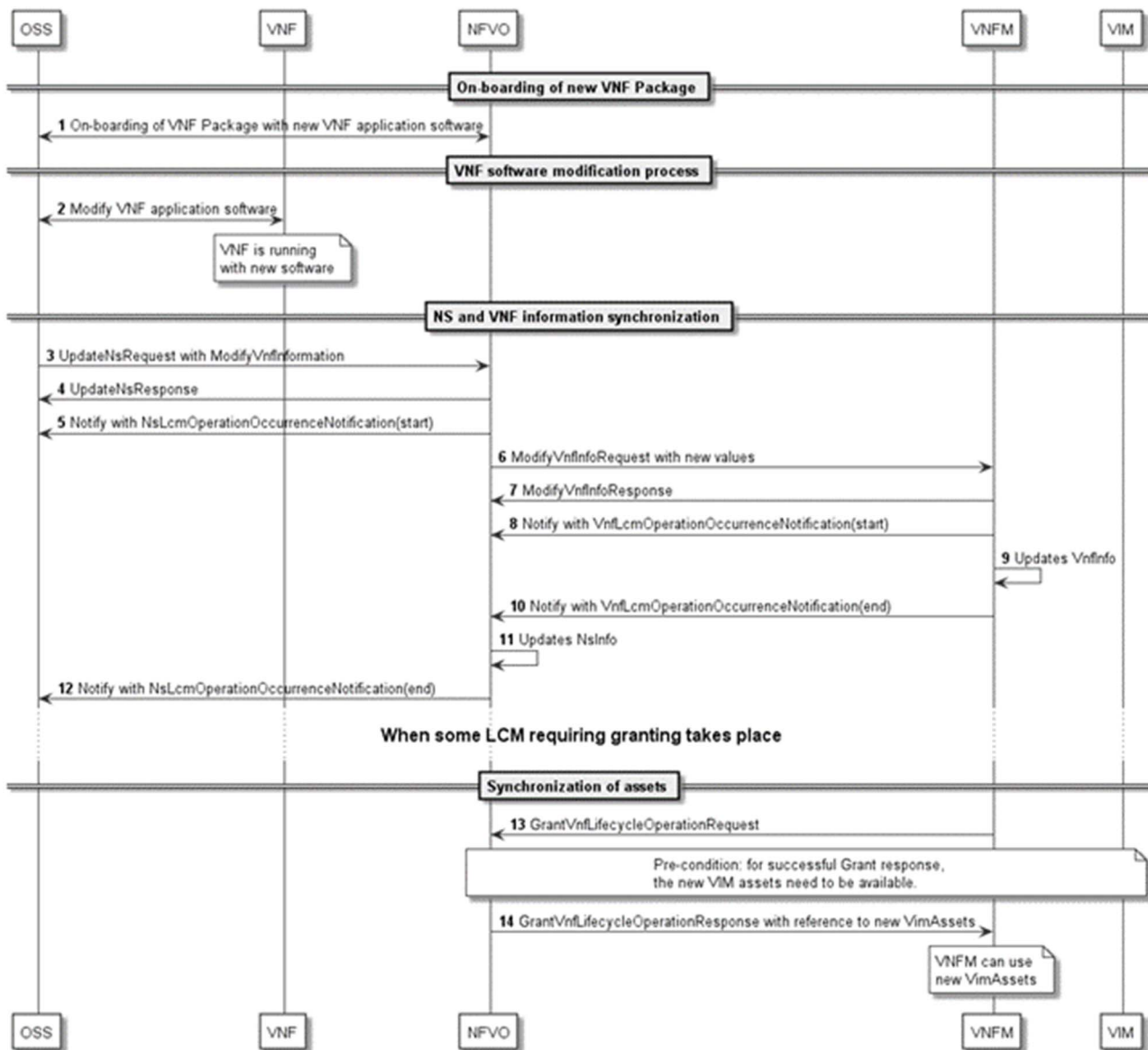
The fulfilment of the above points a) and b) allow the VNFM to use the new software images from the new VNF Package for the creation of new VNFC instances, such as during scale-out, or for the re-creation of existing VNFC instances, such as during VNF healing procedures.

**NOTE:** This software modification procedure was already supported in Release 2 versions of the present document as a reference and for specification completion purposes. Additional information is also present in clause B.2.3.2 of ETSI GS NFV-REL 006 [i.10].

#### B.2.2 Procedure

Figure B.2.2-1 shows the steps of the VNF software modification when it is not assisted by NFV-MANO.

As a pre-condition for the modification process, the VNF Package with the new VNF application software needs to be on-boarded to the NFVO according to step 1 of the procedure.



**Figure B.2.2-1: Procedure of the VNF software modification not assisted by NFV-MANO**

The procedure comprises the following phases and their steps:

**A) On-boarding of the new VNF Package:**

- 1) The VNF Package containing the new VNF application software is on-boarded to the NFVO as requested by the OSS. The NFVO checks the integrity and authenticity of the VNF Package and verifies that all mandatory information in the VNF Package is present and complies with the standard.

**B) VNF software modification process:**

- 2) The VNF application software modification is performed towards the VNF by the external management systems outside the NFV-MANO. As a result, at the end of this step the VNF is running the new software.

**C) NS and VNF information synchronization:**

- 3) Once the VNF application software modification is completed, the OSS requests the NFVO with an UpdateNsRequest to modify the information of the VNF so that it points to the new VNF Package, which has been on-boarded and which contains the new version of the VNF application software which also runs in the VNF instance as a result of step 2.
- 4) The NFVO acknowledges the request for modifying the VNF information by sending an UpdateNsResponse.

- 5) A notification from the NFVO is issued to subscribed consumers to notify them about the start of the NS LCM Update NS operation (see note 1).

NOTE 1: It is assumed in this procedure that the OSS has previously subscribed to the NFVO for this type of notifications.

- 6) The NFVO requests with a `ModifyVnfInfoRequest` the VNFM to modify the information of the VNF instance so that it points to the new VNF Package which has been on-boarded and which contains the new version of the VNF application software.
- 7) The VNFM acknowledges the request for modifying the VNF information by sending a `ModifyVnfInfoResponse`.
- 8) A notification from the VNFM is issued to subscribed consumers to notify them about the start of the Modify VNF information operation (see note 2).

NOTE 2: It is assumed in this procedure that the NFVO has previously subscribed to the VNFM for this type of notifications.

- 9) The VNFM updates the VNF information (`VnfInfo`) accordingly.
- 10) A notification from the VNFM is issued to subscribed consumers to notify them about the end of the Modify VNF information operation.
- 11) The NFVO updates the NS information (`NsInfo`) with the modified information of the VNF instance.
- 12) A notification from the NFVO is issued to subscribed consumers to notify them about the end of the NS LCM Update NS operation.

#### D) Synchronization of VIM assets:

The synchronization of VIM assets in between the NFVO and the VNFM takes place in the first subsequent granting exchange, whenever such granting becomes necessary.

- 13) The VNFM sends a `GrantVnfLifecycleOperationRequest` providing the new `vnfdId` according to the updated information in the `VnfInfo`.
- 14) The NFVO sends to the VNFM a `GrantVnfLifecycleOperationResponse` providing references to the new `VimAssets`. As a pre-condition for a successful grant response, the new VIM assets need to be available (see note 3).

NOTE 3: The creation of the new VIM assets can take place during any of the preceding steps before the VNF LCM granting response, e.g. after VNF Package on-boarding, in parallel to the VNF software modification process, in parallel during the NS and VNF information synchronization. If the creation of VIM assets is initiated only at the time of the granting request, it can significantly delay the granting operation, thus, it can adversely affect the LCM operation requiring the granting (e.g. healing, scaling).

## B.3 VNF software modification assisted by NFV-MANO via change of current VNF Package

### B.3.1 Overview

In this type of VNF software modification, the current VNF Package of a VNF instance is changed. In this process, the NFV-MANO functional blocks support the modification by providing and handling the necessary VNF lifecycle and virtualised resources management operations. By using the NFV-MANO functional blocks, a common handling of the VNF software modification processes can be achieved leveraging also the capabilities of NFV-MANO to create, modify and terminate virtualised resources.

As part of changing the current VNF Package of the VNF instance, modifications to the current set of virtualised resources and/or composition of the VNF instance can take place. As part of the modification process, and in support of it, temporary resources can also be created and used by the VNF to perform the change of the current VNF Package.

### B.3.2 Procedure

Figure B.3.2-1 shows the steps of changing the current VNF Package of a VNF instance with resource modifications performed via the NFV-MANO functional blocks.

As a pre-condition for the modification process, the VNF Package with the new VNF application software needs to be on-boarded to the NFVO according to step 1 in the procedure.

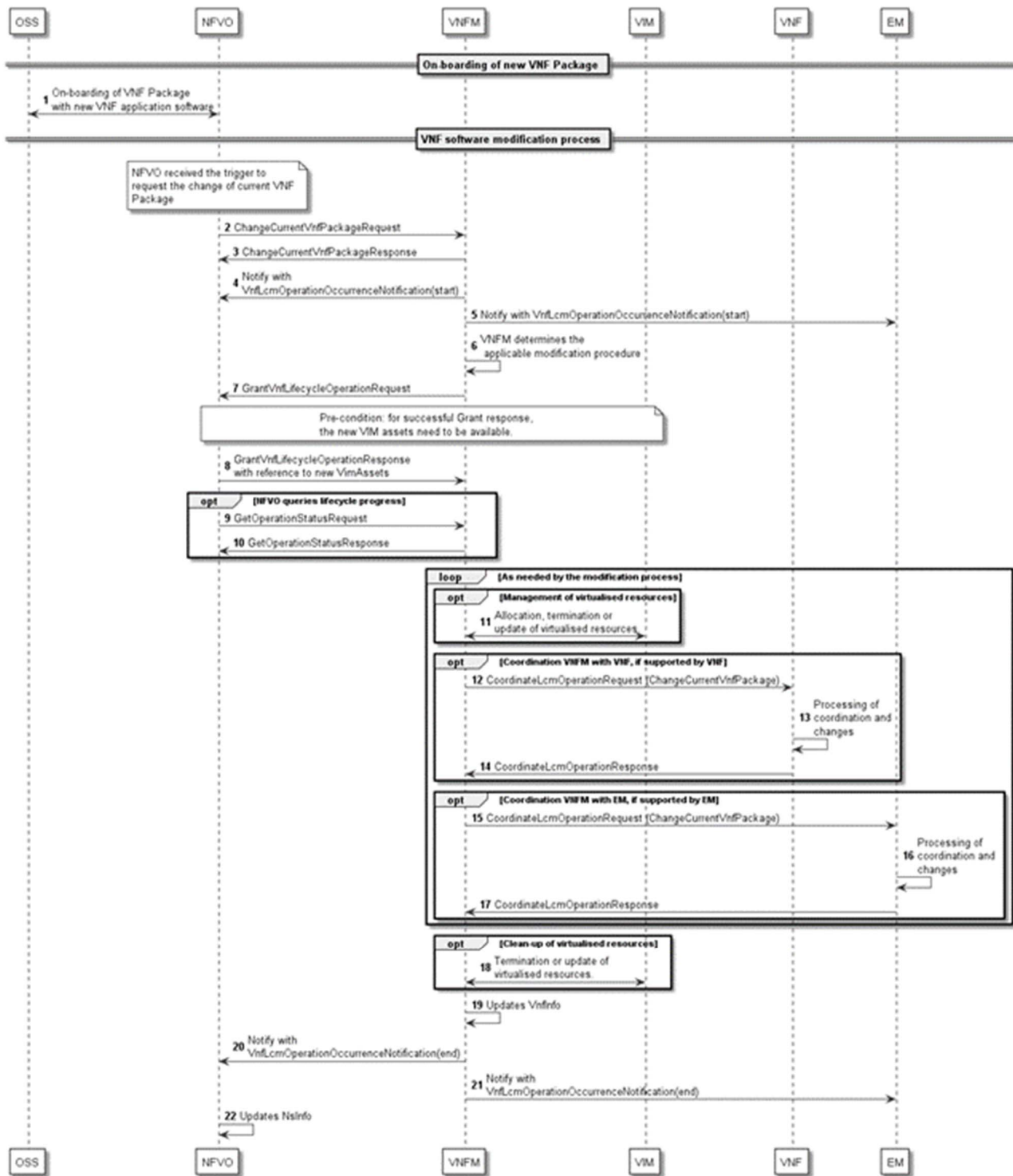


Figure B.3.2-1: Procedure of the VNF software modification performed by NFV-MANO with resource modifications



The procedure comprises the following phases and their steps:

**A) On-boarding of new VNF Package:**

- 1) The VNF Package containing the new VNF application software is on-boarded to the NFVO as requested by the OSS. The NFVO checks the integrity and authenticity of the VNF Package and verifies that all mandatory information in the VNF Package is present and complies with the standard.

**B) VNF software modification process:**

The NFVO has received the request from the OSS/BSS to change the current VNF Package of the VNF. The NFVO has also checked the pre-conditions for the start of the modification process (e.g. availability of the VNF Package with the new software images).

- 2) The NFVO sends a `ChangeCurrentVnfPackageRequest` with the identifier of the destination VNF Package (`dstVnfdId`) to be used for the change.
- 3) If the request has been accepted, the VNFM sends to the NFVO a `ChangeCurrentVnfPackageResponse` with the `lifecycleOperationOccurrenceId`., and the change of current VNF Package will continue with the following steps. In case of error, the modification process is stopped, and appropriate error information is returned by the VNFM to the NFVO.
- 4 & 5) The VNFM sends the "start" notification of the VNF lifecycle operation occurrence indicating the type of operation as "ChangeCurrentVnfPackage" to the subscribed entities of VNF LCM notification. If any error happens during the lifecycle operation, a "result" notification with appropriate error information is sent to the subscribers. See note 1.

NOTE 1: It is assumed in this procedure that the NFVO and other external management systems such as the EM have subscribed to the VNFM for this type of notifications.

- 6) The VNFM verifies and processes the `ChangeCurrentVnfPackageRequest`. This includes determining the applicable VNF Package change information (`VnfPackageChangeInfo`) by searching for the appropriate `VersionSelector` based on the input information from the operation request and the runtime information of the VNF instance (i.e. `VnfInfo`).
- 7) The VNFM determines the virtualised resources of the VNF instance that will be updated due to the change of current VNF Package and/or any virtualised resources needed for the VNF instance. The VNFM sends a `GrantVnfLifecycleOperationRequest` indicating the change with the following input parameters:
  - a) the `dstVnfdId` corresponding to the destination VNF Package used for the modification;
  - b) the `vnfdId` corresponding to the VNF Package in use before the request;
  - c) the type of `lifecycleOperation` as "ChangeCurrentVnfPackage"; and
  - d) the list of `updateResource` with the resources that are updated (e.g. those that will use new software images), the list of `tempResource` for the temporary resources, the list of `addResource` with the new resources to be added, and the list of `removeResource` with the existing resources to be terminated.
- 8) The NFVO sends to the VNFM a `GrantVnfLifecycleOperationResponse`. The response message contains information about the new `VimAssets` (e.g. new software images) available for the software modification. If temporary resources have been also requested, the NFVO confirms the granted temporary resources. As a pre-condition for a successful grant response, the new VIM assets need to be available (see note 2).

NOTE 2: The creation of the new VIM assets can take place during any of the preceding steps before the VNF LCM granting response, e.g. after VNF Package on-boarding.

- 9) (Optionally) After the `ChangeCurrentVnfPackage` has been activated, the NFVO can send to the VNFM, anytime during the modification process, a `GetOperationStatusRequest` to request status information of the ongoing VNF LCM operation.
- 10) If a `GetOperationStatusRequest` has been received by the VNFM, the VNFM processes the request and provides a `GetOperationStatusResponse` with information about the status of the operation.

The modification process continues in a loop of resource management and/or coordination interaction steps. In this flow, the applicable steps supported by interfaces defined in the ETSI GS NFV-IFA 006 [1] and ETSI GS NFV-IFA 008 [i.5] are listed.

NOTE 3: If virtualised resource management in indirect mode is used, virtualised resource management interactions would be supported by interfaces defined in the present document and ETSI GS NFV-IFA 005 [i.4], but these are not detailed in the present information flow.

The number of iterations and the content of each step are specific to the VNF, and moreover depend on the actual modification determined by the source VNFD, source VNF deployment flavour and target VNFD.

NOTE 4: If applicable, some steps of the loop may be executed in parallel.

- 11) If the current iteration step requires virtualised resource management interactions, and direct mode is applicable, the VNFM requests the allocation, termination and/or update of virtualised resources to the corresponding VIM.

If the current iteration step requires coordination with the VNF and the LCM Coordination interface is supported by the VNF, the following steps 12 to 14 are executed:

- 12) The VNFM sends to the VNF a `CoordinateLcmOperationRequest` corresponding to the `ChangeCurrentVnfPackage` operation.
- 13) The VNF processes the coordination request and performs any changes in the VNF instance as needed.
- 14) The VNF sends to the VNFM a `CoordinateLcmOperationResponse`.

If the current iteration step requires coordination with the EM (as an example of an external management system) and the LCM Coordination interface is supported by the EM, the following steps 15 to 17 are executed:

- 15) The VNFM sends to the EM a `CoordinateLcmOperationRequest` corresponding to the `ChangeCurrentVnfPackage` operation.
- 16) The EM processes the coordination request and performs any changes in the VNF instance as needed.
- 17) The EM sends to the VNFM a `CoordinateLcmOperationResponse`.

Upon completion of the modification process iterations, the virtualised resource management process continues as follows:

- 18) If the process requires additional virtualised resource management interactions (e.g. virtualised resource clean-up), and direct mode is applicable, the VNFM requests the termination and/or update of virtualised resources from the corresponding VIM.

Upon completion of the above steps, the procedure continues as follows:

- 19) The VNFM updates the VNF information (`VnfInfo`) as appropriate.
- 20 & 21) A notification from the VNFM is issued to subscribed consumers to notify them about the end of the VNF LCM operation occurrence.
- 22) The NFVO updates the NS information (`NsInfo`) according to the modified information of the VNF instance.

---

## B.4 VNF software modification relationship to NS and NSD management

### B.4.1 Introduction

The NS and the VNF are interrelated at different phases of the network design, deployment and operation, among others:

- The NSD has references to specific VNF Packages and their deployable VNF flavours via VNF profiles.

- The NS instance information maintained by the NFVO has references to the VNF instances that are part of the NS instance.

Due to the relationship between the NS and the VNF, a VNF software modification that involves handling a different VNF Package version needs additional steps for the preparation of the NS and the synchronization/update of its information.

## Annex C (informative): Change History

| Date             | Version | Information about changes  |
|------------------|---------|--|
| 18 December 2014 | V0.0.1  | Skeleton and ToC   |
| 07 January 2015  | V0.0.2  | Updates based on NFVIFA(14)000028r4  |
| 26 January 2015  | V0.1.0  | Early draft after IFA Shanghai Interim Meeting, including contributions: <ul style="list-style-type: none"> <li>- NFVIFA(15)000034r3_IFA007_section_4_Overview</li> <li>- NFVIFA(15)000036r3_IFA0nn_Interface_WIs_section_1_Scope_small_addition</li> <li>- NFVIFA(15)000091r3_IFA009_section_3_Definitions</li> </ul>   |
| 23 February 2015 | V0.1.1  | Editorial: Title corrected for alignment   |
| 25 June 2015     | V0.1.2  | Contributions included: <ul style="list-style-type: none"> <li>- NFVIFA(15)000066r2_IFA007_Clause_5_VNF_Package_interface_notification_req</li> <li>- NFVIFA(15)000067r3_IFA007-008_Clause_5_VNF_LC_change_interface_reqs</li> <li>- NFVIFA(15)000189r5_VNF_Package_management_interface_Requirements</li> <li>- NFVIFA(15)000252r1_IFA007_interface_requirements_VNF_LCM_Granteeing</li> <li>- NFVIFA(15)0000254r4_IFA007_interface_requirements_VNF_LCM</li> <li>- NFVIFA(15)000256r2_IFA007_interface_requirements_VNF_Lifecycle_Chg_Notif</li> <li>- NFVIFA(15)000357r1_Change_to_conventions_for_conditional_attributes</li> <li>- NFVIFA(15)000523r2_IFA010-007-008_Extend_VNF_lifecycle_change_notification</li> <li>- NFVIFA(15)000567r1_Adding_note_from_458r3_to_all_interface_GSs</li> <li>- NFVIFA(15)000675r1_IFA007_Section_5_Reference_Point_Requirements</li> <li>- NFVIFA(15)000798r1_IFA007_Add_VNF_LCM_interface_requirements</li> </ul> Applicability of multi-document changes checked but no changes necessary: <ul style="list-style-type: none"> <li>- NFVIFA(15)000035_Blueprint_Phase_1_GSs_as_Informative_References</li> </ul> Editorial changes: <ul style="list-style-type: none"> <li>- Aligned document structure with template and IFA005 and IFA006</li> <li>- Information elements clause is now clause 8</li> <li>- Inserted separate "Interface requirements" and "Reference point requirements" subclauses in clauses 6 and 7</li> <li>- Implemented disclaimer from NFVTSC(15)000041r3</li> </ul> |
| 10 July 2015     | V0.1.3  | Contributions included: <ul style="list-style-type: none"> <li>- NFVIFA(15)000845r1_IFA007_Move_VNF_LCM_Notification_requirement_from_798r1</li> </ul> Editorial changes: <ul style="list-style-type: none"> <li>- Aligned labels of requirements with IFA conventions as per NFVIFA(15)000853r2</li> </ul>  |
| 13 August 2015   | V0.2.0  | Contributions included: <ul style="list-style-type: none"> <li>- NFVIFA(15)000526r3_IFA007_LCM_operation_granteeing_requirements</li> <li>- NFVIFA(15)000939r1_IFA007_section_5_Or-Vnfm_reference_point_requirements</li> <li>- NFVIFA(15)000082r8_IFA007_VNF_Lifecycle_Manager_and_Lifecycle_Operation_Granteeing</li> <li>- NFVIFA(15)000722r4_IFA007_detailed_interface_design_LCM_-_InstantiateVNF</li> <li>- NFVIFA(15)000723r5_IFA007_detailed_interface_design_LCM_Notification</li> <li>- NFVIFA(15)000838r5_IFA007_VNF_PM_interface</li> <li>- NFVIFA(15)000933r3_IFA007_detailed_interface_design_LCM_-_ScaleVNF</li> <li>- NFVIFA(15)000935r1_IFA007_detailed_interface_design_LCM_-_QueryVNF</li> </ul>  |
| 27 August 2015   | V0.2.1  | Incomplete implementation of NFVIFA(15)000838r5 in v0.2.0 was fixed (clause 8 content from 838r5 was missing in v0.2.0)<br>Editorial alignments in clause 8 (structured into subclauses as done in 838r5, text from 838 adapted to latest conventions (Parameter → Attribute, Type → Content)).  |

| Date             | Version | Information about changes   |
|------------------|---------|---|
| 02 October 2015  | V0.3.0  | Contributions included: <ul style="list-style-type: none"> <li>- NFVIFA(15)0001141r1_IFA007-008_VNF_LCM_Healing_operation_interface_requirement</li> <li>- NFVIFA(15)0001197_IFA007_VNF_Package_Management_small_fix</li> <li>- NFVIFA(15)000837r4_IFA007_VNF_FM_interface</li> <li>- NFVIFA(15)0001142r2_IFA007_VNF_LCM_Healing_operation_interface_specification</li> <li>- NFVIFA(15)000953r4_IFA010-007-008_VNF_FM_extra_notifications</li> <li>- NFVIFA(15)0001199r1_IFA007_IFA013_VNF_Package_Management_Notification_Additional</li> <li>- NFVIFA(15)0001084r2_IFA007_IE_names_alignment</li> <li>- NFVIFA(15)0001221_IFA007_add_description_to_VNF_LCM_interface</li> <li>- NFVIFA(15)0001022r2_IFA007_FM_PM_interface_naming_alignment</li> </ul> Editorial fixes, e.g. to align with latest interface template  |
| 09 November 2015 | V0.4.0  | Contributions included: <ul style="list-style-type: none"> <li>- NFVIFA(15)0001154r2_IFA007_VNF_Package_interface_modify_and_query_operations</li> <li>- NFVIFA(15)0001139r3_IFA007_7-8_IFA008_7-9_VNF_FM_extension_for_VR_state_changes</li> <li>- NFVIFA(15)0001302_IFA007_Adding_VNF_performance_management_requirements</li> <li>- NFVIFA(15)0001152r2_IFA007_7_X_IFA008_7_X_VNFM-produced_VNF_Config_interface</li> <li>- NFVIFA(15)000065r4_IFA007_5_3_3_IFA008_5_2_1_1_Operate_VNF_interface_requiremen</li> <li>- NFVIFA(15)0001150r2_IFA007_7_2_IFA008_7_2_Operate_VNF_interface_specification</li> <li>- NFVIFA(15)0001151r3_IFA007_4-5_IFA008_4-5_VNFM-produced_VNF_Configuration_reqs</li> <li>- NFVIFA(15)000638r8_IFA007_5_2_5_3_resource_management_requirements</li> <li>- NFVIFA(15)0001225r1_IFA013_IFA015_Merging_PM_Information_Elements</li> <li>- NFVIFA(15)0001266r2_IFA007_Exclude_Error_Cases_from_Output_IE_Cardinality</li> <li>- NFVIFA(15)0001347r3_IFA010_Section_7_2_Functional_requirements_for_VNF_LCM</li> </ul> Editorial fixes: <ul style="list-style-type: none"> <li>- Change of affiliation of Marc Flauw</li> <li>- Subclauses of Clauses 5 and 8 renumbered to keep sequence of IEs in sync with sequence of interfaces</li> </ul>   |
| 21 December 2015 | V0.5.0  | Contributions included: <ul style="list-style-type: none"> <li>- NFVIFA(15)0001453r3_IFA007_rapporteur_s_cleanup_of_v040</li> <li>- NFVIFA(15)000934r7_IFA007_detailed_interface_design_LCM_TerminateVNF</li> <li>- NFVIFA(15)0001290r9_IFA007_Virtualised_Compute_Interfaces</li> <li>- NFVIFA(15)0001291r9_IFA007_Virtualised_Network_Interfaces</li> <li>- NFVIFA(15)0001292r9_IFA007_Virtualised_Storage_Interfaces</li> <li>- NFVIFA(15)0001455r2_IFA005_IFA006_IFA007_IFA008_IFA013_FM_PM_fixes</li> <li>- NFVIFA(15)0001458r1_IFA007_fixes_References_Introduction</li> <li>- NFVIFA(15)0001485r4_IFA007_IFA008_VNF_Scaling_Parameters</li> <li>- NFVIFA(15)0001495r2_IFA007_5-3-3_IFA008_5-2-1-1_VNF_LCM_extension</li> <li>- NFVIFA(15)0001500r1_IFA007_6_2_2_IFA013_7_7_5_Addressing_note_VNF_Package_mgmt</li> <li>- NFVIFA(15)0001515r3_IFA007_Non-normative_should_and_may_separated_from_1453r1</li> <li>- NFVIFA(15)0001529_IFA007-008_7_X_Correction_to_subscribe_filter_for_VNF_FM_i_f</li> <li>- NFVIFA(15)0001596_IFA005_IFA006_IFA007_IFA008_IFA012_IFA013_Remove_section_9_S</li> <li>- NFVIFA(15)0001608r2_IFA005_IFA006_IFA007_IFA008_IFA013_Normative_Reference_to_IF</li> <li>- NFVIFA(15)0001613_IFA007_Subscribe_Notify_description_fixes</li> </ul> Editorial fixes: <ul style="list-style-type: none"> <li>- Implemented the agreement regarding table numbering</li> <li>- Implemented the agreement regarding the text referencing the tables for input and output parameters</li> </ul> |

| Date          | Version | Information about changes  |
|---------------|---------|--|
| February 2016 | V0.6.0  | Contributions included: <ul style="list-style-type: none"> <li>- NFVIFA(15)0001454r5_IFA007_5_3_3_4_fixing_Virtualised_Resources_Change_Notificat</li> <li>- NFVIFA(16)000072r2_IFA007_5_2_and_5_3_3_Additional_requirements_for_indirect_RM</li> <li>- NFVIFA(15)0001519r5_IFA007_numberOfSteps_support_signaling</li> <li>- NFVIFA(16)000007r1_IFA007_referencing_IFA011</li> <li>- NFVIFA(16)000042r2_IFA011_IFA007_VNF_LCM_related_information_in_VNFD</li> <li>- NFVIFA(16)000106r1_IFA007_5_3_and_8_6_IFA008_5_2_and_9_4_1_Identification_for_V</li> <li>- NFVIFA(16)000117r1_IFA007_8_5_4_Adding_basic_VnfInfo_attributes</li> <li>- NFVIFA(16)000119r1_IFA007-IFA008_7_2_Addressing_editor_note_on_VNF_operate</li> <li>- NFVIFA(16)000121r1_IFA007_5_3_3_Rapporteurs_fixes_indirect_RM</li> <li>- NFVIFA(16)000123r1_IFA007_7_2_6_2_Additional_params_in_healing</li> <li>- NFVIFA(16)000151r1_IFA007_Indicator_Interface, with editorial fixes (copy&amp;paste error (replaced in change 2 "Ve-Vnfm-em by Or-Vnfm), "parameters" instead of "information elements" in operations, added "... and notifications" in 8.10 headline)</li> <li>- NFVIFA(15)000511r9_IFA007_6_3_detailed_interface_design_LCM_Operation_Granteeing (with editorial fixes to align with the interface template)</li> </ul> Editorial fixes: <ul style="list-style-type: none"> <li>- Pre-processing done before TB approval E-mail: mailto:edithelp@etsi.org. Rapporteur's note: Had to undo the changes to front matter as this document is still intended for being made available through the open area.</li> <li>- Applied conventions according to NFVIFA(15)0001562r5_Interface_template_update, including removal of the editor's notes that stated the need to add UML diagrams to the IE clauses</li> <li>- Various small typo fixes</li> </ul>   |
| March 2016    | V0.6.1  | Re-created the ZIP archive due to a problem in the ZIP file of V 0.6.0. No changes to content.   |
| 21 March 2016 | V0.7.0  | Version to enter WG review<br>Contributions included: <ul style="list-style-type: none"> <li>- NFVIFA(16)000175r3_IFA007_Alarm_Cleared_Notification_and_Alarm_IE_Update</li> <li>- NFVIFA(16)000183_IFA007_Referencing_IFA013_informatively</li> <li>- NFVIFA(16)000102r2_IFA007_Section7_2_Modification_on_Query_operation</li> <li>- NFVIFA(16)000142r6_IFA007_8_5_IFA008_9_3_current_scale_level_in_VNFInfo</li> <li>- NFVIFA(16)000267r4_IFA007_IFA008_IFA011_scale_VNF_to_instantiation_level</li> <li>- NFVIFA(16)000171r3_IFA007_C_D_8_G_H_8_I_J_Virtualised_Resources_Performance_Ma</li> <li>- NFVIFA(16)000170r3_IFA007_A_B_8_A_B_8_C_D_8_E_F_Virtualised_Resources_Fault_Man</li> <li>- NFVIFA(16)000176r3_IFA007_Fixing_normative_and_informative_references_to_IFA_GS</li> <li>- NFVIFA(16)000197_IFA007_IFA008_instantiation_level_in_InstantiateVNF</li> <li>- NFVIFA(16)000217_IFA007_IFA008_Adding_description_to_VNF_Instance</li> <li>- NFVIFA(16)000219r2_IFA007_IFA008_resolving_editor_s_note_on_VnfInfo</li> <li>- NFVIFA(16)000220_IFA007_scaling_step_note_alignment_with_proposal_from_779</li> <li>- NFVIFA(16)000228r2_IFA007_editor_s_notes_on_externally_managed_internal_VLs</li> <li>- NFVIFA(16)000231r1_IFA007_Adding_deployment_flavour_to_grant_request</li> <li>- NFVIFA(16)000232r3_IFA007_6_3_2_Adding_level_to_grant_request</li> <li>- NFVIFA(16)000234r2_IFA007_6_4_2-4_X_8_X_Y_Virtualised_Resources_Change_Notifica</li> <li>- NFVIFA(16)000235_IFA007_Adding_deployment_flavour_to_VnfInfo</li> <li>- NFVIFA(16)000239_IFA007_7_5_3_Notify_operation</li> <li>- NFVIFA(16)000248r2_IFA007_7_2_8_Change_VNF_Flavour</li> <li>- NFVIFA(16)000258r1_IFA007_5_2_5_3_3_quota_management_requirements_in_indirect</li> <li>- NFVIFA(16)000259r2_IFA007_6_x_8_x_quota_management_interfaces_in_indirect_mode</li> <li>- NFVIFA(16)000262r1_IFA007_5_2_5_3_virtualised_resources_quota_available_notifi</li> <li>- NFVIFA(16)000265r4_IFA007_8_3_2_8_5_5_Adding_ResourceInfo</li> <li>- NFVIFA(16)000268r2_IFA007_6_2_2_and_8_2_x_IFA013_7_7_5_and_8_7_x_accessing_VNF</li> </ul> |

| Date          | Version | Information about changes   |
|---------------|---------|---|
|               |         | <ul style="list-style-type: none"> <li>- NFVIFA(16)000269r1_IFA007_5_3_3_Fixing_Virtualised_Resources_Management_interfa (also applied the pattern to the newly added indirect RM interface requirements)</li> <li>- NFVIFA(16)000276r1_IFA007_7_6_2_7_2_X_Clarification_on_ModifyVnfConfiguration_a</li> <li>- NFVIFA(16)000277r1_IFA007_6_3_2_4_Clarification_of_rejection_in_granting_operat</li> <li>- NFVIFA(16)000279_IFA007_6_3_2_LC_operation_occurrence_identifier_in_granting</li> <li>- NFVIFA(16)000280r1_IFA007_6_4_and_8_4_Interface_spec_of_reservation_mgmt_in_ind</li> <li>- NFVIFA(16)000285_IFA007_8_3_8_adding_resourceProviderId_in_ConstraintResource</li> <li>- NFVIFA(16)000287r1_IFA007_Scaling_description</li> <li>- NFVIFA(16)000288r2_IFA007_6_4_2-3_Y_8_4_A-C_8_K_L_Virtualised_Resources_Inform</li> <li>- NFVIFA(16)000299r1_IFA007_8_3_4_Addressing_editor_note_in_VimInfo</li> <li>- NFVIFA(16)000300_IFA007_8_3_5_Addressing_editor_note_in_ZoneInfo</li> <li>- NFVIFA(16)000301r1_IFA007_8_5_5_IFA_011_7_1_X_Attributes_for_VnfInfo_and_VNFD<br/>(this document has inserted flavourId as well as doc 235. IFA agreed on 24 March by email to remove the duplicate variant of flavourId that was introduced by 301)</li> <li>- NFVIFA(16)000314r2_IFA007_5_6_and_8_Adding_VR_reservation_change_notification</li> <li>- NFVIFA(16)000327r1_IFA007_Resolve_Editor_s_Notes_NFV002_reference</li> <li>- NFVIFA(16)000328r1_IFA007_Resolve_Editor_s_Notes_Functional_requirements_refer</li> <li>- NFVIFA(16)000329r1_IFA007_Resolve_Editor_s_Notes_Granting_in_ScaleVnf_Descript</li> <li>- NFVIFA(16)000330_IFA007_Resolve_Editor_s_Notes_Indirect_RM_IE_clause_introdu</li> <li>- NFVIFA(16)000331_IFA007_Resolve_quote_easy_quote_Editor_s_Notes</li> <li>- NFVIFA(16)000342_IFA007_8_7_VNF_PM_mirror</li> </ul> <p>Editorial fixes:</p> <ul style="list-style-type: none"> <li>- there were still occurrences of "input/output information element" in the GS where "input/output parameter" needs to be used. Fix as editorial</li> <li>- Table 7.2.4.2-1: Subscribe operation input parameters --&gt; TerminateVnf operation input parameters</li> <li>- various typos</li> <li>- change "section" to "clause"- in FM/PM interfaces, there were still a quite few table references for input and output parameters that used the old formulation ("are listed") instead of the latest convention "shall follow the indications". Fixed.</li> <li>- made ToC of depth 3 instead of 4</li> <li>- converted those additional Editor's Notes that were inserted during GS preparation by the rapporteur into "Rapporteur's notes". A Rapporteur's note has not been agreed by the group but represents the opinion of/tracks an action for/points out an issue detected by the rapporteur during GS preparation</li> </ul> |
| 20 April 2016 | V0.8.0  | <p>Contributions included (review EA part 1):</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)000373_IFA007_6_2_4_Add_missed_text_for_new_VNF_package_on-boarded</li> <li>- NFVIFA(16)000421r2_IFA007_7_2_10_IFA008_7_2_10_review_Modify_Vnf_fixes</li> <li>- NFVIFA(16)000423_IFA007_8_2_7_2_IFA013_8_6_5_2_review_UserMetadata_mandatory</li> <li>- NFVIFA(16)000424r1_IFA007_6_and_7_and_8_Remove_stage3_term</li> <li>- NFVIFA(16)000425r2_IFA007_many_IFA008_many_review_Small_Technical_Alignment</li> <li>- NFVIFA(16)000431_IFA007_7_5_3_Editorial_change_for_AlarmClearedNotification</li> <li>- NFVIFA(16)000443r1_IFA007_6_3_2_2_review_Temp_Resource_in_Notifications_delete</li> <li>- NFVIFA(16)000471r1_IFA007_4_1_Alignment_listing_of_interfaces</li> <li>- NFVIFA(16)000473_IFA007_8_2_5_VNF_Package_mgmt_correction_on_VnfPackageChange</li> <li>- NFVIFA(16)000481r1_IFA007_8_5_6_and_8_5_7_VNF_LCM_updates_to_VnfInfo_and_VnfRes</li> <li>- NFVIFA(16)000482_IFA007_7_4_2_and_7_4_5_VNF_PM_changes</li> </ul>   |

| Date          | Version | Information about changes  |
|---------------|---------|--|
|               |         | <ul style="list-style-type: none"> <li>- NfVIFA(16)000488r1_IFA007_5_2_edits_interface_naming_in_requirements_and_titles</li> <li>- NfVIFA(16)000501r1_IFA007_6_2_2_Query_VNF_package_operation</li> <li>- NfVIFA(16)000503r1_IFA007_6_3_2_1_Policy_in_VNF_Lifecycle_Operation_Granteeing_in</li> <li>- NfVIFA(16)000506r2_IFA007_7_2_9_1_operate_VNF_operation</li> <li>- NfVIFA(16)000515r1_IFA007_7_2_3_Clarifications_on_Scale_VNF_operation</li> <li>- NfVIFA(16)000517r1_IFA007_8_3_2_and_8_3_3_Updates_to_IEs_related_to_Granteeing</li> <li>- NfVIFA(16)000519_IFA007_8_7_2_and_8_7_3_Updates_IEs_related_to_ObjectSelectio</li> <li>- NfVIFA(16)000521_IFA007_7_7_VNF_Indicator_interface_description_alignment</li> <li>- NfVIFA(16)000529_IFA007_5_3_4_Correcting_req_on_query_VNF_operation</li> <li>- NfVIFA(16)000558r1_IFA007_7_2_3_2_7_2_4_2_7_2_6_2_8_5_6_2_IFA008_7_2_7_2_7_2_8</li> </ul>   |
| 20 April 2016 | V0.8.0  | <p>Contributions included (review EA part 2):</p> <ul style="list-style-type: none"> <li>- NfVIFA(16)000422_IFA007_7_2_11_IFA008_sect_review_GetOperationStatus_mandat</li> <li>- NfVIFA(16)000476r1_IFA007_7_2_3_and_6_3_2_moving_text_about_granteeing</li> <li>- NfVIFA(16)000357r2_IFA007_Scaling_description_delta_after_Espoo</li> <li>- NfVIFA(16)000444r2_IFA007_many_IFA008_9_4_2_review_Removing_Editor_s_Notes</li> <li>- NfVIFA(16)000418r2_IFA007_section_7_2_3_2_IFA008_section_7_2_7_2_-_Fixing_aspec</li> <li>- NfVIFA(16)000408r1_IFA007_Renaming_VL_and_CP_IEs</li> </ul> <p>Contributions included (review EA part 3):</p> <ul style="list-style-type: none"> <li>- NfVIFA(16)000441r1_IFA007_IFA008_Remove_the_definition_of_KeyValuePair</li> <li>- NfVIFA(16)000478r3_IFA007_8_6_2_VNF_LC_Change_Notification_addressing_EN_and_co</li> <li>- NfVIFA(16)000495_IFA007_6_4_5_VR_PM_indirect_add_missing_resourceProviderId</li> <li>- NfVIFA(16)000496r1_IFA007_5_3_5_lifecycle_change_notification_interface_require</li> <li>- NfVIFA(16)000502r1_IFA007_5_3_2_6_3_2_2_VNF_instance_id_for_granteeing_interface</li> <li>- NfVIFA(16)000523r1_IFA007_5_3_4_clarification_on_VNF_instance_information_modif</li> <li>- NfVIFA(16)000546r2_IFA007_5_3_5_7_3_3_8_6_1_8_6_X_Add_new_type_notification</li> <li>- NfVIFA(16)000549r1_IFA007_8_2_7_8_7_7_8_8_3_IFA008_9_3_4_9_7_7_Use_of_time</li> </ul> <p>Contributions included (ATL meeting):</p> <ul style="list-style-type: none"> <li>- NfVIFA(16)000398_IFA007_8_4_2_8_4_4_8_4_6_Alignment_to_inheritance_pattern</li> <li>- NfVIFA(16)000419_IFA007_6_3_2_1_review_Resource_types_in_Granteeing</li> <li>- NfVIFA(16)000420r2_IFA007_7_2_3_IFA008_7_2_7_IFA011_7_1_5_3_review_Scale_up_dow</li> <li>- NfVIFA(16)000445r1_IFA007_8_5_6_IFA008_9_4_2_review_VnfInfo_fixes</li> <li>- NfVIFA(16)000450r3_IFA007_8_5_7_IFA008_9_4_4_VnfResourceInfo_IE</li> <li>- NfVIFA(16)000465r3_IFA007_8_3_review_VDU_reference_duplicated</li> <li>- Note on change in clause 8.3.2.2/resourceTemplate: The insertion of "or modification" was in the wrong place in the sentence, hinting a "modification of new resources" which is nonsense. This was corrected as an editorial action to read "modification of existing resources"</li> <li>- NfVIFA(16)000466r5_IFA007_6_3_2_3_review_computeFlavour_swImage_assets_multi_VI</li> <li>- NfVIFA(16)000474r1_IFA007_8_3_7_and_6_3_2_2_Granteeing_IE_PlacementConstraint_vs</li> <li>- NfVIFA(16)000504r4_IFA007_6_3_2_changes_on_VNF_Lifecycle_Operation_Granteeing_int</li> <li>- NfVIFA(16)000513_IFA007_6_3_2_Clarifications_for_Grant_VNF_LC_operation</li> </ul> |



| Date         | Version | Information about changes   |
|--------------|---------|---|
|              |         | <ul style="list-style-type: none"> <li>- NFVIFA(16)000514r3_IFA007_5_2_5_3_3_10_6_4_8_quota_available_notification_inter</li> <li>- NFVIFA(16)000520_IFA007_5_3_7_VNF_FM_missing_requirements</li> <li>- NFVIFA(16)000522r1_IFA007_7_2_9_Operate_VNF_graceful_and_forceful_stop</li> <li>- NFVIFA(16)000527r1_IFA007_7_2_2_and_7_2_7_VNF_LCM_QueryVNF_filter_and_correctio</li> </ul> <p>Note on change in table 7.2.7.3-1: The note in the description column is not in line with the EDR. The note has been moved to the last row of the table as an editorial action</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)000533r1_IFA007_8_5_3_Addressing_EN_on_ConnectionPoint</li> <li>- NFVIFA(16)000551r4_IFA007_7_2_3_IFA008_7_2_7_VNF_Scaling_description</li> <li>- NFVIFA(16)000592_IFA007_8_5_6_IFA008_9_4_1_review_VimInfo_in_VnfInfo</li> <li>- NFVIFA(16)000597_IFA007_8_5_6_review_Remove_Error_from_OperateVnf</li> <li>- NFVIFA(16)000600r6_IFA007_IFA013_Add_support_for_Create_and_Delete_VNF</li> <li>- NFVIFA(16)000652_IFA007_7_2_11_IFA008_7_2_9_GetOperationStatus_op_specific_st</li> <li>- NFVIFA(16)000676r1_IFA007_7_2_5_ext_VLs_in_ChangeVnfFlavour</li> </ul> <p>Contributions included (S1a#36 call):</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)000667r1_IFA007_IFA008_small_fixes</li> <li>- NFVIFA(16)000721_IFA007_7_2_Adding_LCM_operation_occurrence_identifier</li> </ul> <p>Contributions included (S1b#50 call and EA ending 19 May):</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)000453r2_IFA007_8_8_4_IFA008_9_3_4_Referencing_resources_in_alarm_IE</li> <li>- NFVIFA(16)000461r7_IFA007_8_6_2-5_IFA008_9_5_1-4_VnfLifecycleChangeNotification</li> <li>- NFVIFA(16)000720r2_IFA007_8_5_8_IFA008_9_4_5_Clarification_for_resource_identif</li> <li>- NFVIFA(16)000484r9_IFA008_7_2_2_9_4_x_IFA007_7_2_2_7_6_2_8_5_x_8_9_x_Adding_Vi</li> </ul> <p>Editorial fixes:</p> <ul style="list-style-type: none"> <li>- Reference i.3b renamed</li> <li>- "See note" harmonized</li> <li>- Virtualised → Virtualised</li> <li>- "Interface" → interface consistently</li> <li>- Convention enforcement: "parameter" → "attribute" in information element descriptions</li> <li>- "Functional requirement" → "Requirement" (table headings in Interface requirements sections)</li> <li>- Various minor fixes</li> </ul> |
| 17 June 2016 | V0.9.0  | <p>Contributions included (second review EA#1, 9 June):</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)000719r1_IFA007_8_5_8_Adding_back_the_reservationId</li> <li>- NFVIFA(16)000769r2_IFA008_7_4_2_9_2_9_8_6_2_IFA007_7_6_2_and_IFA011_7_1_6</li> <li>- NFVIFA(16)000784_IFA008_5_3_1_3_IFA007_5_3_9_renaming_VNF_Indicator_interfac</li> <li>- NFVIFA(16)000786r1_IFA007_5_3_4_IFA008_5_2_1_1_Add_missing_requirements_on_crea</li> <li>- NFVIFA(16)000788_IFA007_IFA008_IFA013_4_3_Removal_of_N_A_condition</li> <li>- NFVIFA(16)000793_IFA007_8_3_2_2nd_review_Removing_Rapp_note</li> <li>- NFVIFA(16)000794r1_IFA007_7_2_1_IFA008_7_2_1_2nd_review_lcOpOcclD_clarification</li> <li>- NFVIFA(16)000809_IFA007_Typo_Correction</li> <li>- NFVIFA(16)000820r1_IFA007_Resolution_of_editor_s_notes</li> <li>- NFVIFA(16)000835r3_IFA007_8_3_3_8_12_4_Adding_Resource_Group_Id_to_Grant_respon</li> <li>- NFVIFA(16)000836r2_IFA007_5_3_5_and_IFA008_5_2_1_2_Missing_req_subscription_for</li> </ul>  |

| Date         | Version | Information about changes   |
|--------------|---------|---|
|              |         | <p>NOTE: In the change tracked version, this was implemented using the same name tag as for 835r3, i.e. r0-835r2.</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)000837_IFA007_5_3_8_7_6_8_9_and_IFA008_5_2_1_5_7_4_9_2_on_adding</li> <li>- NFVIFA(16)000838r2_IFA007_8_5_x_IFA008_9_4_x_Add_Info_the_VL_and_CP_IEs</li> <li>- NFVIFA(16)000839_IFA007_5_3_9_IFA008_5_2_1_4_VNF_Indicator_interface_require</li> <li>- NFVIFA(16)000841r1_IFA007_6_2_2_6_2_5_Query___Fetch_VNF_Package_operation</li> <li>- NFVIFA(16)000852_IFA007_IFA008_Editorials_and_alignments</li> <li>- NFVIFA(16)000790r5_IFA007_many_IFA008_many_2nd_review_Create_VNF_terminology_an <ul style="list-style-type: none"> <li>• Rapporteur's changes when implementing this contribution: Table 8.6.8.3-1 VnfIdentifierDeletionNotification: Used past tense in Description column, instead of future as suggested by the 790r5, as notifications are about past events, not future ones. See also the entry for NFVIFA(16)0001016</li> </ul> </li> <li>- NFVIFA(16)000857r2_IFA007_7_3_2_IFA008_7_5_2_2nd_review_Subscribe_to_Create_Del</li> <li>- NFVIFA(16)000860_IFA007_8_5_7_IFA008_9_8_4_2nd_review_ResourceHandle_fix</li> <li>- NFVIFA(16)000862r2_IFA007_7_2_6_8_2_12_IFA008_7_2_13_9_4_12_2nd_review_Aligning</li> <li>- NFVIFA(16)000864_IFA007_8_5_6_8_5_3_8_12_IFA008_9_4_3_9_4_11_9_4_9_2_4_2nd_ex</li> </ul>  |
| 17 June 2016 | V0.9.0  | <ul style="list-style-type: none"> <li>- NFVIFA(16)000887r2_IFA007_Implementing_identifier_conventions_from_614r3_in_IFA (implemented under user name r2-864 same as previous contribution) <ul style="list-style-type: none"> <li>• Rapporteur's changes when implementing this contribution (mostly because the IE name is different where it is declared): <ul style="list-style-type: none"> <li>- Table 8.5.2.2-1 one occurrence of "Vld" replaced by "VnfVld" ("Vld" IE does not exist)</li> <li>- Table 8.5.7.2-1 extVirtualLink -&gt; extVirtualLinkId not applied since another document has modified this attribute, such that it is not of type "Identifier" any longer - hence Identifier conventions do not apply</li> <li>- Table 8.6.3.2-1 VnfcResourceInformation -&gt; VnfcResourceInfo</li> <li>- Table 8.6.4.2-1 VirtualLinkResourceInformation -&gt; VResourceInfo</li> <li>- Table 8.6.5.2-1 VirtualStorageResourceInformation -&gt; VirtualStorageResourceInfo</li> </ul> </li> </ul> </li> </ul> <p>Contributions included (after NFVIFA#33, Sophia Antipolis):</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)0001016_IFA007_many_IFA008_many_Create_VNF_terminology_and_states_re (Rapporteur's comment: 790r5 has been superseded by 1016 which is in fact r6 of 790. The delta between 1016 and 790r5 is implemented in this revision, as 790r5 was implemented previously, effectively being equivalent to having implemented 1016 directly, instead of 790r5. Also, some instances of "VNF information element" in clauses 7.2.7.1 and 7.2.2.4 were missed to be replaced by 1016; these instances were replaced too)</li> <li>- NFVIFA(16)000791r5_IFA007_many_IFA008_many_2nd_review_Renaming_VI_and_Vld_in_IE</li> <li>- NFVIFA(16)000795r3_IFA007_8_5_6_7_2_3_IFA008_9_4_3_7_2_3_2nd_review_VNF_localiz</li> <li>- NFVIFA(16)000889r1_IFA007_inner_grouping_of_indirect_RM_IEs (Rapporteur's comment: When used in the context of "InformationChangeNotification", replaced a few occurrences of "VirtualisedResourceWithRpChangeNotification" by "InformationWithRpChangeNotification", namely in 6.4.2.3, 6.4.3.3 and 6.4.4.3, assuming this was a copy&amp;paste error in the original contribution, and "InformationWithRpChangeNotification" is the correct substitute of "InformationChangeNotification" in indirect RM)</li> <li>- NFVIFA(16)000919r1_IFA007_IFA008_IFA011_IFA012_IFA013_IFA014_stage_3_data_types (Rapporteur's comment: In clauses 7.3.2.2 and 7.6.3.2, added "Filter" in the an empty content column instead of "not specified" as the parameter name is "filter", following the convention)</li> <li>- NFVIFA(16)000856_614bis_Conventions for Identifiers_UPDATED</li> <li>- NFVIFA(16)000869r1_IFA007_IFA008_IFA011_IFA012_IFA013_IFA014_Proposal for an update of the inheritance pattern convention</li> <li>- NFVIFA(16)000920r1_IFA007_6_3_2nd_review_operation_names_in_granting</li> <li>- NFVIFA(16)000983r2_IFA007_6_3_Ext_VLs_in_Granting</li> </ul> |

| Date           | Version | Information about changes   |
|----------------|---------|---|
|                |         | <ul style="list-style-type: none"> <li>- NFVIFA(16)000989_IFA007_8_11_2_3_VimInfo_in_VirtualisedResource QuotaAvailable</li> <li>- NFVIFA(16)0001001r3_IFA007_8_5_2_IFA008_9_4_10_Change_to_Virtual_Lin</li> <li>- NFVIFA(16)0001011r2_IFA007_8_5_5_IFA008_9_4_2_Note_on_modification_of_VnfInfo</li> </ul> <p>Contributions included (after EA ending 14 Jul 2016):</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)0001041_IFA007_query_filter</li> <li>- NFVIFA(16)000806r11_IFA007_and_IFA013_identification_of_the_VNF_Package (Rapporteur's comment: The previous changes (r0-841r1 that were applied to the Fetch VNF Package operation were moved to the "Fetch onboarded VNF Package artifacts" operation to which they apply after the change introduced by 806)</li> <li>- NFVIFA(16)0001032r2_IFA007_6_3_VNF_Lifecycle_Operation_Granteeing_interface_8_3_6_</li> <li>- NFVIFA(16)001020r3_IFA013_Abort_VnfPackage_Deletion</li> <li>- NFVIFA(16)000922r6_Conventions_for_the_use_of_abbreviations</li> </ul> <p>Contributions included (after S1a#43 on Jul 20):</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)0001066r1_IFA007_IFA008_move_extension_and_vnfConfigurableProperty_to_VnfInfo</li> <li>- NFVIFA(16)0001063r1_IFA007_IFA008_IFA013_vnfInstanceName_in_ModifyVnfConfig</li> </ul> <p>Editorial fixes:</p> <ul style="list-style-type: none"> <li>- Minor typos (flavour → flavour, identifier → identifier, etc.)</li> <li>- Renamed extVirtualLinkLink to extVirtualLink</li> <li>- Changed the filename convention to use six digit version string</li> <li>- Replaced "GrantLifecycleOperation" by "GrantVnfLifecycleOperation" in captions in line with the name of the operation and in related message names</li> <li>- Corrected wrong references to IFA006 from clauses 6.4 and 8.4 (indirect resource management)</li> <li>- In the body of clause 8.4.7.4.2, replaced "AlarmNotification" by "AlarmClearedNotification" as the whole clause uses "AlarmClearedNotification" elsewhere, so this is assumed a copy&amp;paste error</li> <li>- Using plural in the description of attributes and parameters of 0..N/1..N cardinality</li> <li>- Aligned operation names usage (single words, all uppercase) in the table captions and clause headline</li> <li>- Rapporteur action #1 from 489r1: ensure consistent use of "VNF Package" -&gt; s/VNF package/VNF Package/</li> </ul> |
| 29 July 2016   | V0.9.1  | <p>Contributions included (S1b call with approval power on 25 July 2016):</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)0001077_IFA007_IFA008_IFA013_IFA015_ExtCP_and_LinkPort_fixes</li> <li>- NFVIFA(16)0001078r2_IFA013_8_3_3_IFA007_8_5_IFA008_9_4_Alignment_of_VnfInfo</li> <li>- NFVIFA(16)0001088r1_IFA007_IFA008_IFA013_virtualStorage_Alignment_with_IFA011</li> </ul> <p>Contributions included (S1a call with approval power on 27 July 2016):</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)001094r1-NFV-IFA007v000901-cb</li> </ul> <p>Editorials:</p> <ul style="list-style-type: none"> <li>- Cross-checked references to IFA011 and removed related rapporteur's notes</li> <li>- 4.2 using proper interface name: s/VNF Configuration/VNF Configuration Management/</li> <li>- Table 6.3.2.2-1: Renamed vnfDesclId --&gt; vnfId in line with the changes done in 806r11</li> </ul>   |
| 01 August 2016 | V 0.9.2 | <p>Editorials:</p> <ul style="list-style-type: none"> <li>- Restructured the sequence of sub-clauses of clause 8.5 to align with IFA008 (i.e. start with VnfInfo and InstantiatedVnfInfo)</li> <li>- Fixed some typos and editorial inconsistencies</li> <li>- Fixed references in clause 7.4.1 Description to be: <ul style="list-style-type: none"> <li>• PerformanceInformationAvailableNotification (see clause 8.7.8)</li> <li>• PerformanceReport information element (see clause 8.7.5)</li> </ul> </li> </ul>   |
| 03 August 2016 | V 0.9.3 | <p>Contributions included:</p> <ul style="list-style-type: none"> <li>- NFVIFA(16)001126r3_IFA007_IFA008_inconsistency_fixes</li> </ul>   |

| Date              | Version | Information about changes  |
|-------------------|---------|--|
|                   |         | Editorials as documented in NFVIFA(16)0001129:<br>- "Change VNF Deployment Flavor" replaced by the generally-used term "Change VNF Flavour"<br>- Applied convention for notifications<br>- There are some references left to VirtualLinkDesc but in fact the IE is named VnfVirtualLinkDesc in IFA011. Fixed   |
| 04 August 2016    | V 0.9.4 | Contributions included:<br>- NFVIFA(16)000770_Replace_primitive_type_TimeStamp_by_DateTime<br>- Extended the implementation of change 5 in NFVIFA(16)001126r3_IFA007_IFA008_inconsistency_fixes to all places where the text is applicable (Description of input/output parameters of type ExtVirtualLink and ExtManagedVirtualLink starts with "Information about", rather than "Reference to")   |
| 19 August 2016    | V0.9.4b | Alignment of the Operation Result clauses: result of an operation use past tense and return parameter use passive present tense and avoid future tense (will be). Output parameter mentioned in attribute descriptions are also changed to use "is returned" or "are returned"<br>Other editorial bugs fixed<br>Replaced many occurrences of NVFO with NFVO  |
| 05 September 2016 | V0.9.5  | Including NFVIFA(16)0001215  |
| October 2016      | V2.1.1  | Publication  |
| 23 January 2017   | V2.1.2  | Specification maintenance begins<br><br>CRs included:<br>- NFVIFA(17)000010r2_IFA007ed221_Merging_LCCN_with_LCM_interface_and_adding_subsc<br>Editorial changes:<br>- Changed page header to "Draft GS", added NFV's DRAFT GS disclaimer   |
| 03 April 2017     | V2.1.3  | CRs included:<br>- NFVIFA(17)000056r1_IFA007ed221_IFA013ed221_VNF_Package_Management_modifications<br>- NFVIFA(17)000062r4_IFA007ed221_IFA008ed221_VimInfo_fixes_without_VimId_changes<br>- NFVIFA(17)000094r3_IFA007ed221_Update_the_content_and_description_of_the_alarm_<br>- NFVIFA(17)000103r3_IFA007ed221_ModifyVnfConfig_Split_and_Merge<br>- NFVIFA(17)000116r2_IFA007ed221_Various_small_bugfixes<br>- NFVIFA(17)000155_IFA007ed221_LifecycleChangeNotification_terminology_<br>- NFVIFA(17)000176r2_IFA007ed221_IFA008ed221_VimId_changes_separated_from_62r3<br>- NFVIFA(17)000193_IFA007ed221_ThresholdCrossedNotification_trigger_condition_f<br>- NFVIFA(17)000236r1_IFA007ed221_8_5_3_clarify_description_of_MonitoringParameter<br>- NFVIFA(17)000257r2_IFA007ed221_IFA008ed221_IFA013ed221_Fix_to_dynamic_addresses   |
| 25 May 2017       | V2.1.4  | CRs included:<br>- NFVIFA(17)000274r2_IFA007ed221_IFA008ed221_VNF_FM_Acknowledge_Alarm_operation<br>- NFVIFA(17)000275r1_IFA007ed221_IFA008ed221_VNF_FM_Alarm_List_Rebuilt_operation<br>- NFVIFA(17)000355_IFA007_Fix_inconsistencies_in_the_FaultyResourceInfo_IE<br>- NFVIFA(17)000438_IFA007ed221_removing_two_attributes_from_SoftwareImageInform<br>- NFVIFA(17)000454r1_IFA007ed221_IFA008ed221_Add_notes_to_the_Delete_PM_Jobs_op<br>- NFVIFA(17)000455r1_IFA007ed221_Add_notes_to_the_Delete_Thresholds_operation_fo<br>- NFVIFA(17)000458r3_IFA007ed221_Clarify_the_results_of_operations_implicitly_upd<br>- NFVIFA(17)000460_IFA007ed221_IsAutomaticInvocation_flag_for_autoscale_and_aut<br>- NFVIFA(17)000462r1_IFA007ed221_ChangeExtVLs_fixes<br>- NFVIFA(17)000469_IFA007ed221_resource_metadata<br>- NFVIFA(17)000473_IFA007ed221_identifier_changes_related_to_IFA_document_256r1 |
| 13 June 2017      | V2.1.5  | CRs included:  |

| Date             | Version | Information about changes  |
|------------------|---------|--|
|                  |         | <ul style="list-style-type: none"> <li>- NfVIFA(17)000390r2_IFA007ed221_CR_add_error_handling_operations</li> <li>- NfVIFA(17)000427r2_IFA007ed221_VL_and_CP_consistency</li> <li>- NfVIFA(17)000450r4_IFA007ed221_ChangeExtVLs_support_status</li> <li>- NfVIFA(17)000468r2_IFA007ed221_Notifications_triggered_by_ModifyVnf</li> <li>- NfVIFA(17)000470r1_IFA007ed221_Problem_with_storage_resources_in_AffectedVnfc (rapporteur changed "VnfInstance" to "VnfInfo", as this is a leftover from the original contribution having been targeted towards SOL003)</li> <li>- NfVIFA(17)000471r1_IFA007ed221_additionalParameters_missing_from_TerminateVnfRe</li> <li>- NfVIFA(17)000520r3_IFA007ed221_VimConstraint_for_resourceGroup</li> <li>- NfVIFA(17)000531_IFA007ed221_Improvement_of_attribute_usage_discription (implemented on top of the changes from NfVIFA(17)000427r2, as intended by this CR)</li> <li>- NfVIFA(17)000535r1_IFA007ed221_Add_VimConnectionInfo_input_parameter_to_Change</li> </ul>  |
| 21 June 2017     | V2.1.6  | <p>Final draft for approval after NfVIFA#57</p> <p>CRs included:</p> <ul style="list-style-type: none"> <li>- NfVIFA(17)000525_IFA007ed221_VimConnectionInfo_inter_stages_consistency</li> <li>- NfVIFA(17)000547r3_IFA007_ed221_CR_Align_the_usage_of_VNF_instantiation_state</li> <li>- NfVIFA(17)000580r1_IFA007ed231_ChangedInfo_fix_of_cardinality</li> <li>- NfVIFA(17)000597_IFA007ed231_Small_fix_leftover_from_renaming_to_vimConnectio</li> </ul> <p>Editorial fixes (table formatting, empty table rows removed)</p>  |
| 07 November 2017 | V 2.3.2 | <p>Contributions incorporated that were approved at NfVIFA#73:</p> <ul style="list-style-type: none"> <li>- NfVIFA(17)000919_IFA007ed241_miscellaneous_small_fixes</li> <li>- NfVIFA(17)000923r1_IFA007ed241_resource_metadata_in_AffectedC_N_S_resources (change by rapporteur when implementing this CR: replaced "VnfVirtualStorageResourceInfo" by "VirtualStorageResourceInfo" in the new text to reflect the correct name of that pre-existing IE)</li> <li>- NfVIFA(17)000970r1_IFA007ed241_VnfcCps_in_AffectedVnfc</li> </ul> <p>Editorials:</p> <ul style="list-style-type: none"> <li>- Added draft disclaimer box</li> <li>- Fixed small typos</li> </ul>   |
| 21 December 2017 | V 2.3.3 | <p>Contributions incorporated that were approved at NfVIFA#79:</p> <ul style="list-style-type: none"> <li>- NfVIFA(17)0001037r4_IFA007ed241_CR_from_IFA_1029 (CR conflict handling by the rapporteur: the change in the description of the "address" attribute in CR 1037r4 was applied in the new "CpProtocolInfo" IE that was introduced by 1114r1, as the "address" attribute has been moved by CR 1114r1 to this new IE)</li> <li>- NfVIFA(17)0001089r1_IFA007ed241_Align_query_VNF_package_operation_with_IFA013</li> <li>- NfVIFA(17)0001090r1_IFA007ed241_Align_fetch_VNF_package_operation_with_IFA013</li> <li>- NfVIFA(17)0001091_IFA007ed241_Align_VNF_package_notification_IE_with_IFA013</li> <li>- NfVIFA(17)0001101_IFA007ed241_Clarification_of_ExtManagedVirtualLink_and_ExtVi</li> <li>- NfVIFA(17)0001108_IFA007ed241_remove_redundant_description_of_vnfConfigurableP</li> <li>- NfVIFA(17)0001111r1_IFA007ed241_LCM_operation_response_and_notification_in_opera (The Rapporteur has fixed the following mis-alignment: The first pattern in change 1 includes the text 'trigger the sending of the "start" LCM notification' whereas the second pattern in change 3 includes the text 'trigger the "start" notification'. As the first pattern is more precise, the rapporteur has aligned the second pattern with the first one when implementing the CR).</li> <li>- NfVIFA(17)0001114r1_IFA007ed241___Corrections_related_to_multiple_layer_protocol</li> <li>- NfVIFA(17)0001120r1_IFA007ed241_fixing_VNF_connectivity_figure</li> </ul> |
| February 2018    | V 2.4.1 | Publication  |
| 01 March 2018    | V 2.4.2 | <p>Contributions incorporated that were approved at NfVIFA#89:</p> <ul style="list-style-type: none"> <li>- NfVIFA(18)000143_IFA007ed251_align_filters_in_FM_interface_Subscribe_operation</li> </ul>  |
| May 2018         | V 3.0.0 | Release 3 baseline version created from draft v2.4.2, as agreed in NfVIFA#98   |

| Date           | Version | Information about changes  |
|----------------|---------|--|
| June 2018      | V 3.0.1 | Contributions incorporated that were approved at NFVIFA#101 and NFVIFA#102: <ul style="list-style-type: none"> <li>- NFVIFA(18)000421r1_IFA007_MegaCR_FEAT04_Compute_Host_Reservatio<br/>n</li> <li>- NFVIFA(18)000507r1_IFA007ed311_-_Mirror_-_Linking_<br/>VNFC_CP_and_VnfExt_CP</li> <li>- NFVIFA(18)000508r1_IFA007ed311_-_Mirror_-_<br/>_Clarifying_association_from_VnfLinkPort_to_VnfcCp_and_VnfExtCp</li> </ul>  |
| June 2018      | V 3.0.2 | Contributions incorporated that were approved at NFVIFA#104 and NFVIFA#105: <ul style="list-style-type: none"> <li>- NFVIFA(18)000430r1_FEAT07_IFA007_MegaCR_Support_of<br/>policy_management_interface</li> <li>- NFVIFA(18)000561r3_IFA007_MegaCR_FEAT15_VNF_Snapshot</li> <li>- NFVIFA(18)000564_IFA007ed311_Remove_current_values_of<br/>monitoringParameter_attribute_from_VNF_LCM_interface</li> <li>- NFVIFA(18)000612r1_IFA007ed311_-_Rel3Mirror_-_<br/>_Fixing_sentence_related_to_PM_delivery_mechanism</li> <li>- NFVIFA(18)000619_IFA007ed311_-_Rel3Mirror_-_<br/>_Fixing_note_in_VnfLinkPortInfo</li> <li>- NFVIFA(18)000628_IFA007ed311_-_Rel3Mirror_-_<br/>_different_names_for_virtual_link_descriptor_ids</li> <li>- NFVIFA(18)000637_IFA007ed311_Rel3Mirror_of_435r2_metadata_for_CP_I<br/>Es</li> <li>- NFVIFA(18)000651_IFA007ed311_Rel3Mirror_of<br/>477_Fixing_cardinality_of_ConstraintResourceRef</li> </ul> |
| September 2018 | V 3.1.2 | Contribution incorporated that was approved at NFVIFA#118: <ul style="list-style-type: none"> <li>- NFVIFA(18)000806_IFA007_Support_for_partial_VNF_Snapshot_Packages</li> <li>- NFVIFA(18)000834_IFA007ed321_Mirror_for_SOL_contribution_on_making_t<br/>he_API_su_rface_consistent_for_bootData</li> </ul> Editorial fixes   |
| November 2018  | V 3.1.3 | Contributions incorporated that were approved at NFVIFA#124 and NFVIFA#125: <ul style="list-style-type: none"> <li>- NFVIFA(18)000859r2_IFA007ed321_Metadata_Extension_ConfigurableProps<br/>_clarification_for_VnfInfo</li> <li>- NFVIFA(18)000887r2_IFA007ed321_update_of_Ies_related_to_PM_interface<br/>_for_IFA027_alignment</li> <li>- NFVIFA(18)000922_IFA007_Clause_7_2_Corrections_additional_params_for<br/>_create_and_revert_snapshot_operations</li> <li>- NFVIFA(18)000923r1_IFA007_Clause_8_5_Correction_cardinality_VnfcSnap<br/>shotInfo_and_description_VnfSnapshotInfo</li> </ul>   |
| January 2019   | V 3.1.4 | Contributions incorporated that were approved at NFVIFA#128 and NFVIFA#129: <ul style="list-style-type: none"> <li>- NFVIFA(18)000983r4_IFA007ed321_Add_best_effort_in_PlacementContraint</li> <li>- NFVIFA(18)0001002_IFA007ed321_CR_add_policy_associate_disassociate_<br/>operations</li> <li>- NFVIFA(18)0001032_IFA007_VNF_snapshot_createdAt_and_userDefinedDat<br/>a</li> <li>- NFVIFA(18)0001067r1_IFA007ed321_Fix_for_condition_in_VnfLcmOperation<br/>OccurrenceNotification</li> <li>- NFVIFA(18)0001068r1_IFA007ed321_declaration_of_metadata_and_extensio<br/>ns_</li> <li>- NFVIFA(18)0001070r1_IFA007ed321_Aligning_conditions_for_vduld_and_res<br/>ourceTemplateId_in_Granteeing_interface</li> </ul>   |
| February 2019  | V 3.1.5 | Contributions incorporated that were approved at NFVIFA#137: <ul style="list-style-type: none"> <li>- NFVIFA(18)0001112r4_IFA007_MegaCR_FEAT010_Or-<br/>Vnfm_ref_point_interface_specification_for_Multi-Site_Service</li> <li>- NFVIFA(19)000059r4_FEAT02_IFA007_MegaCR</li> </ul>  |
| February 2019  | V 3.1.6 | Contribution incorporated that was approved at NFVIFA#139: <ul style="list-style-type: none"> <li>- NFVIFA(19)000142r2_FEAT02_IFA007_Review_add_missing_parameters_to<br/>_ChangeCurrent</li> </ul>  |
| April 2019     | V3.2.1  | Version update for publication   |
| May 2019       | V3.2.2  | Base line version for Release 3 Drop 3 created from published version 3.2.1  |
| June 2019      | V3.2.3  | Contributions incorporated that were approved at NFVIFA#155: <ul style="list-style-type: none"> <li>- NFVIFA(19)000506_IFA007_Alignment_with_Stage_3_work_on_VNF_snapsh<br/>ot_feature</li> <li>- NFVIFA(19)000532r1_IFA007_8.5.17_Correction_of_NOTE</li> </ul>   |
| July 2019      | V3.2.4  | Update with CRs:<br>NFVIFA(19)000653: IFA007ed331 Rel3Mirror 8.10.3 IndicatorInformation IE<br>NFVIFA(19)000649: IFA007ed331 Rel3Mirror 8.6.2<br>VnfLcmOperationOccurrenceNotification IE  |

| Date           | Version | Information about changes   |
|----------------|---------|---|
|                |         | NFVIFA(19)000643: IFA007ed331 Rel3Mirror VnfPkgInfo,<br>VnfPackageOnboardingNotification, VnfPackageSoftwareImageInformation IEs<br>NFVIFA(19)000640: IFA007ed331 PerformanceValueEntry IE<br>NFVIFA(19)000634: IFA007ed331 Rel3Mirror VnfcResourceInfo,<br>VnfVirtualLinkResourceInfo, VirtualStorageResourceInfo IE<br>NFVIFA(19)000621: IFA007ed331 Rel3Mirror AffectedVnfc, AffectedVirtualLink,<br>AffectedVirtualStorage<br>NFVIFA(19)000619: IFA007ed331 Rel3Mirror 8.8.4 Alarm IE - rephrase "legal values"<br>NFVIFA(19)000614r1:<br>IFA007ed331_Rel3_mirror_initial_configurable_properties_values  |
| September 2019 | V3.3.1  | Version update for publication  |
| October 2019   | V3.3.2  | First draft for ed341   |
| October 2019   | V3.3.3  | Update with CRs:<br>NFVIFA(19)000760: IFA007ed341 measurementContext in<br>ThresholdCrossedNotification<br>NFVIFA(19)000818: IFA007ed341 modifying VNF package references<br>NFVIFA(19)000841: IFA007ed341 relaxing PM subscriptions<br>NFVIFA(19)000825r1: IFA007ed341 exposing maxScaleLevels<br>Rapporteur action: Removing Annex on Authors & Contributors  |
| December 2019  | V3.3.4  | Update with CRs:<br>NFVIFA(19)000923r5: IFA007ed341 Dynamic creation and deletion of trunk supports<br>NFVIFA(19)000950: IFA007ed341 5.3.11 add requirement for Update VNF snapshot<br>package<br>NFVIFA(19)000875r5: IFA007ed341 Enhancements in ChangeExtVnfConnectivity  |
| February 2020  | V3.3.5  | Update with CRs:<br>NFVIFA(19)000993: IFA007 Improve wording left for protocol design stage<br>NFVIFA(19)0001004: IFA007ed341 adding vnfConfProps to<br>ChangeCurrentVnfPackage<br>NFVIFA(20)000012: IFA007ed341 adding missing extensions and<br>vnfConfigurableProperties to ChangeVnfFlavour<br>NFVIFA(20)000049r1: IFA007ed341 add missing support statements<br>NFVIFA(20)000069r1: IFA007ed341 sync to IFA015 work according to 942r2 part1<br>NFVIFA(20)000090: IFA007ed341 FEAT15 maintenance Enhancing granting related to<br>VNF snapshot<br>Rapporteur action: correcting references of "supportedOperations" to<br>"supportedOperation" |
| March 2020     | V3.3.6  | Update with CRs:<br>NFVIFA(20)000140r1: IFA007ed341 sync to IFA015 work according to 942r18<br>NFVIFA(20)000169r2: IFA007ed341 fix Enum values<br>NFVIFA(20)000172r1: IFA007ed341 FEAT15 alignment with stage 3<br>NFVIFA(20)000192: IFA007ed341 FEAT15 alignment with stage 3 - part 2   |
| May 2020       | V3.3.7  | Update with CRs:<br>NFVIFA(20)000204: IFA007ed341 FEAT15 alignment with stage 3 - part 3<br>NFVIFA(20)000232: IFA007ed341 FEAT15 Moving VNF snapshot package API<br>NFVIFA(20)000223r2: IFA007ed341 FEAT02 Indicator changes triggered by<br>changeCurrentVnfPkg<br>NFVIFA(20)000287: IFA007ed341 fixing description of GetIndicatorValue<br>Rapporteur action: correcting description of attributes that has Reference<br>NFVIFA(20)000318r3: IFA007ed341 FEAT15 VNF state snapshot data<br>NFVIFA(20)000326: IFA007ed341 Mirror of NFVIFA(20)000257 alignment issue   |

---

## History

| <b>Document history</b> |                |             |
|-------------------------|----------------|-------------|
| V3.1.1                  | August 2018    | Publication |
| V3.2.1                  | April 2019     | Publication |
| V3.3.1                  | September 2019 | Publication |
| V3.4.1                  | June 2020      | Publication |
|                         |                |             |