## O-RAN.WG6.O2DMS-INTERFACE- ETSI-NFV-PROFILE-R004-v08.00

*Technical Specification*

**O-RAN Working Group 6**

**O2dms Interface Specification: Profile based on ETSI NFV Protocol and Data Models**

Copyright © 2024 by the O-RAN ALLIANCE e.V.

The copying or incorporation into any other work of part or all of the material available in this specification in any form without the prior written permission of O-RAN ALLIANCE e.V. is prohibited, save that you may print or download extracts of the material of this specification for your personal use, or copy the material of this specification for the purpose of sending to individual third parties for their information provided that you acknowledge O-RAN ALLIANCE as the source of the material and that you inform the third party that these conditions apply to them and that they must comply with them.

O-RAN ALLIANCE e.V., Buschkauler Weg 27, 53347 Alfter, Germany Register of Associations, Bonn VR 11238, VAT ID DE321720189

# Table of Contents

1. [Introduction 4](#_bookmark0)
   1. [Scope 4](#_bookmark1)
   2. [References 4](#_bookmark2)
   3. [Definitions and Abbreviations 6](#_bookmark39)
      1. [Definitions 6](#_bookmark40)
      2. [Abbreviations 6](#_bookmark41)
2. [Profiling concepts 7](#_bookmark42)
   1. [Introduction 7](#_bookmark43)
   2. [ETSI NFV framework and solutions 8](#_bookmark44)
   3. [ETSI NFV profiling for O2dms interfaces 9](#_bookmark45)
      1. [General 9](#_bookmark46)
      2. [Conceptual mappings 9](#_bookmark47)
      3. [API mappings 11](#_bookmark48)
   4. [Service operations 11](#_bookmark49)
      1. [Overview 11](#_bookmark50)
      2. [NF Deployment lifecycle management 12](#_bookmark51)
      3. [NF Deployment fault management 40](#_bookmark54)
      4. [NF Deployment performance management 44](#_bookmark55)
   5. [Orchestration aspects with O2dms ETSI NFV profile 54](#_bookmark56)
      1. [Resource groups and tenancy 54](#_bookmark57)
      2. [Placement information and constraints 55](#_bookmark58)
      3. [Network connectivity for NF Deployments 55](#_bookmark59)
      4. [Software images information 58](#_bookmark60)
      5. [Resource management for NF Deployments 58](#_bookmark61)
      6. [Acceleration abstraction layer 60](#_bookmark62)
      7. [DMS Interactions 65](#_bookmark63)
      8. [Monitoring of NF Deployments 67](#_bookmark64)
3. [API definitions 69](#_bookmark65)
   1. [General aspects 69](#_bookmark66)
      1. [Introduction 69](#_bookmark67)
      2. [URI structure and supported content formats 69](#_bookmark68)
      3. [Usage of HTTP header fields 70](#_bookmark69)
      4. [Result set control 71](#_bookmark70)
      5. [Error reporting 71](#_bookmark71)
      6. [Common data types 71](#_bookmark72)
      7. [Authorization of API requests and notifications 73](#_bookmark73)
      8. [Version management 73](#_bookmark74)
      9. [HTTP conditional requests 74](#_bookmark75)
   2. [O2dms\_DeploymentLifecycle Service API 74](#_bookmark76)
      1. [Description 74](#_bookmark77)
      2. [API version 74](#_bookmark78)
      3. [REST resources structure and methods 74](#_bookmark79)
      4. [REST resources 77](#_bookmark80)
      5. [Notifications 80](#_bookmark81)
      6. [Data model 81](#_bookmark82)
      7. [Error handling 94](#_bookmark83)
   3. [O2dms\_DeploymentFault Service API 96](#_bookmark84)
      1. [Description 96](#_bookmark85)
      2. [API version 96](#_bookmark86)
      3. [REST resources structure and methods 96](#_bookmark87)
      4. [REST resources 97](#_bookmark88)
      5. [Notifications 98](#_bookmark89)
      6. [Data model 98](#_bookmark90)
   4. [O2dms\_DeploymentPerformance Service API 99](#_bookmark91)
      1. [Description 99](#_bookmark92)
      2. [API version 99](#_bookmark93)
      3. [REST resources structure and methods 100](#_bookmark94)
      4. [REST resources 101](#_bookmark95)
      5. [Notifications 101](#_bookmark96)
      6. [Data model 102](#_bookmark97)

[Annex A (Informative): Examples 104](#_bookmark98)

* 1. [Overview 104](#_bookmark99)
  2. [AAL examples 104](#_bookmark100)
     1. [Example #1: Definition of acceleration capability requirements in NF Deployment descriptors 104](#_bookmark101)
     2. [Example #2: Definition of acceleration related configuration in NF Deployment descriptors and](#_bookmark102)

[exchange of information on the O2dms interfaces 105](#_bookmark102)

* + 1. [Example #3: Exchange of information on the O2dms interfaces related to acceleration 107](#_bookmark103)

[Annex (informative): Change History 108](#_bookmark104)

# Introduction

## Scope

This Technical Specification has been produced by the O-RAN Alliance.

The contents of the present document are subject to continuing work within O-RAN and may change following formal O-RAN approval. Should the O-RAN Alliance modify the contents of the present document, it will be re-released by O- RAN with an identifying change of release date and an increase in version number as follows:

Release x.y.z where:

x the first digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc. (the initial approved document will have x=01).

y the second digit is incremented when editorial only changes have been incorporated in the document.

z the third digit included only in working versions of the document indicating incremental changes during the editing process.

The present document specifies the O-RAN O-Cloud’s DMS interface protocol and data model based on the ETSI NFV solutions.

## References

The following documents contain provisions which, through reference in this text, constitute provisions of this specification (see also <https://www.o-ran.org/specifications>).

1. 3GPP TR 21.905, Vocabulary for 3GPP Specifications.
2. O-RAN.WG6.ORCH-USE-CASES (v03.00): "Cloudification and Orchestration Use Cases and Requirements for O-RAN Virtualized RAN".
3. O-RAN.WG6.CAD (v02.01): "Cloud Architecture and Deployment Scenarios for O-RAN Virtualized RAN".
4. O-RAN.WG6.O2-GA&P (v01.02): "O-RAN O2 Interface; General Aspects and Principles".
5. ETSI GS NFV-IFA 005 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Or-Vi reference point – Interface and Information Model Specification".
6. ETSI GS NFV-IFA 006 (V4.5.1): " Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Vi-Vnfm reference point – Interface and Information Model Specification"
7. ETSI GS NFV-IFA 007 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Or-Vnfm reference point - Interface and Information Model Specification".
8. ETSI GS NFV-IFA 010 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Functional requirements specification".
9. ETSI GS NFV-IFA 040 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Requirements for service interfaces and object model for OS container management and orchestration specification".
10. ETSI GS NFV-SOL 001 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; NFV descriptors based on TOSCA specification".
11. ETSI GS NFV-SOL 003 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; RESTful protocols specification for the Or-Vnfm Reference Point".
12. ETSI GS NFV-IFA 011 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; VNF Descriptor and Packaging Specification".
13. ETSI GS NFV-SOL 004 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; VNF Package and PNFD Archive specification".
14. ETSI GS NFV-SOL 006 (V4.3.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; NFV descriptors based on YANG specification".
15. ETSI GS NFV-SOL 013 (V4.4.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; Specification of common aspects for RESTful NFV MANO APIs".
16. ETSI GS NFV-SOL 014 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; YAML data model specification for descriptor-based virtualised resource management".
17. ETSI GS NFV-SOL 015 (V1.2.1): "Network Functions Virtualisation (NFV); Protocols and Data Models; Specification of Patterns and Conventions for RESTful NFV-MANO APIs".
18. ETSI GS NFV-SOL 018 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; Profiling specification of protocol and data model solution for OS container management and orchestration".
19. ETSI NFV Registries: "VIM connection information registry".

NOTE: Available at [https://nfvwiki.etsi.org/index.php?title=VIM\_connection\_information.](https://nfvwiki.etsi.org/index.php?title=VIM_connection_information)

1. IETF RFC 793: "Transmission Control Protocol". NOTE: Available at [https://tools.ietf.org/html/rfc793.](https://tools.ietf.org/html/rfc793)
2. IETF RFC 3339: "Date and Time on the Internet: Timestamps". NOTE: Available at [https://tools.ietf.org/html/rfc3339.](https://tools.ietf.org/html/rfc3339)
3. IETF RFC 3986: "Uniform Resource Identifier (URI): Generic Syntax". NOTE: Available at [https://tools.ietf.org/html/rfc3986.](https://tools.ietf.org/html/rfc3986)
4. IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2". NOTE: Available at [https://tools.ietf.org/html/rfc5246.](https://tools.ietf.org/html/rfc5246)
5. IETF RFC 7230: "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing". NOTE: Available at [https://tools.ietf.org/html/rfc7230.](https://tools.ietf.org/html/rfc7230)
6. IETF RFC 8259: "The JavaScript Object Notation (JSON) Data Interchange Format". NOTE: Available at [https://tools.ietf.org/html/rfc8259.](https://tools.ietf.org/html/rfc8259)
7. JSON Schema: "Core definitions and terminology", Version draft-07, November 19, 2017. NOTE 1: Draft-07 is available at [https://json-schema.org/specification-links.html#draft-7.](https://json-schema.org/specification-links.html#draft-7)

NOTE 2: The specification is available as Internet Draft at [https://tools.ietf.org/html/draft-handrews-json-schema-](https://tools.ietf.org/html/draft-handrews-json-schema-01) [01.](https://tools.ietf.org/html/draft-handrews-json-schema-01)

1. OpenAPI™ Specification.

NOTE 1: Available at [https://github.com/OAI/OpenAPI-Specification.](https://github.com/OAI/OpenAPI-Specification)

NOTE 2: The OpenAPI™ Word Mark and OpenAPI Logo are either registered trademarks/service marks or trademarks/service marks of the Linux Foundation, in the United Stated and other countries and are used with the Linux Foundation’s permission. O-RAN Alliance is not affiliated with, endorsed or sponsored by the Linux Foundation, or the Linux Foundation community.

1. Void
2. ETSI GS NFV-SOL 002 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Protocols and Data Models; RESTful protocols specification for the Ve-Vnfm Reference Point".
3. ETSI GR NFV 003 (V1.8.1): "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".
4. O-RAN.WG6.AAL-GAnP.0 (v05.00): "O-RAN Acceleration Abstraction Layer; General Aspects and Principles".
5. O-RAN.WG6.AAL-Common-API (v03.00): "Acceleration Abstraction Layer; Common API".
6. ETSI GR NFV-EVE 022 (V5.1.1): "Network Functions Virtualisation (NFV) Release 5; Architectural Framework; Report on VNF configuration".
7. ETSI GS NFV-IFA 027 (V4.4.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Performance Measurements Specification".
8. ETSI GS NFV-IFA 045 (V4.5.1): "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Fault and alarms modelling specification".
9. IETF RFC 2818: "HTTP over TLS".

NOTE: Available at [https://datatracker.ietf.org/doc/html/rfc2818.](https://datatracker.ietf.org/doc/html/rfc2818)

1. ETSI TS 133 310 (V18.3.0): "Universal Mobile Telecommunications System (UMTS); LTE; 5G; Network Domain Security (NDS); Authentication Framework (AF) (3GPP TS 33.310 version 18.3.0 Release 18)".

## Definitions and Abbreviations

### Definitions

For the purposes of the present document, the terms given in 3GPP TR 21.905 [[1]](#_bookmark3) and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [[1].](#_bookmark3)

### Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [[1]](#_bookmark3) and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [[1].](#_bookmark3)

API Application Programming Interface

CIR Container Image Registry

CIS Container Infrastructure Service

CISM Container Infrastructure Service Management DMS Deployment Management Services

FOCOM Federated O-Cloud Orchestration & Management HTTP Hypertext Transfer Protocol

IMS Infrastructure Management Services

JSON JavaScript Object Notation

LCM Lifecycle Management

MCIO Managed Container Infrastructure Object

MCIOP Managed Container Infrastructure Object Package NF Network Function

NFO Network Function Orchestration

NFV Network Functions Virtualisation NFV-MANO NFV Management and Orchestration NFVI NFV Infrastructure

NFVO NFV Orchestrator

REST Representational state transfer

SMO Service Management and Orchestration

TOSCA Topology and Orchestration Specification for Cloud Applications TLS Transport Layer Security

URI Uniform Resource Identifier

VIM Virtualised Infrastructure Manager

VIP Virtual IP

VL Virtual Link

VM Virtual Machine

VNF Virtualized Network Function

VNFC VNF Component

VNFD VNF Descriptor

VNFM VNF Manager

# Profiling concepts

## Introduction

The O-RAN framework follows similar principles as other frameworks and solutions related to network functions virtualisation and cloudification that have been or are being specified in other organizations, such as in the ETSI NFV. O-RAN.WG6.ORCH-USE-CASES [[2]](#_bookmark4) documents a set of use cases regarding the deployment of O-RAN NF on O- Clouds, as well as it specifies relevant functional and interface requirements between the SMO and O-Cloud. O- RAN.WG6.CAD [[3]](#_bookmark5) documents the cloudification deployment aspects of O-RAN NF based on VMs, OS container in VMs, or OS containers. These same forms of NF deployment are also considered by the ETSI NFV framework.

With the objective to reuse existing network operator assets, e.g., as presently in use in other network deployments beyond RAN, the present document aims at specifying a profile of the ETSI NFV specifications that can also be used for the deployment of O-RAN NF on O-Clouds.

The present O-Cloud O2dms profile based on ETSI NFV demarcates a defined protocol and data model between the SMO and O-Cloud based on the standardized virtualised resource, containerized workload, and VNF management capabilities of the functional entities and functions of the NFV-MANO architectural framework summarized in clause 2.2.

In the definition of the applicable protocol and data models to profile, two main demarcations can be defined depending on the end point of interaction of the SMO towards the O-Cloud:

* "O2-D demarcation #1": SMO interacts with a DMS that offer VNFM-like functionality. This case considers a more abstract "NF Deployment" view, i.e., DMS supports deployment of defined O-RAN NFs. The "NF Deployment" view is also referred in numerous use cases of O-RAN-WG6.ORCH-USE-CASES [2]. The "NF Deployment" concept is defined in the O-RAN.WG6.O2-GA&P [[4]](#_bookmark6) and O-RAN.WG6.CAD [3]. Clause 2.2.1 of the same O-RAN.WG6.O2-GA&P [[4]](#_bookmark6) describes the DMS as the functional block responsible for the management of NF Deployments into the O-Cloud.
* "O2-D demarcation #2": SMO interacts with a DMS that offers virtualised resource/containerized workload management functionality; thus, SMO performs VNFM-like functionality. This case considers a more granular "virtualised resource/containerized workload" management view, i.e., DMS has no notion at all of O-RAN NFs.

These two demarcation cases are depicted in figures 2.1-1 and 2.1-2, respectively.

NOTE 1: VIM/CISM and NFVI/CIS placement in the architecture figures are not meant to be accurate and it is not the focus of these figures to present the architectural mappings of other elements such as SMO, NFVO, etc. Purpose of the figures is to illustrate the demarcations/boundaries of applicability for the O2dms ETSI NFV profiles.

NOTE 2: More information about the NFV-MANO architectural framework is provided in clause 2.2.

|  |  |
| --- | --- |
| Diagram  Description automatically generated | Diagram  Description automatically generated |
| **Figure 2.1-1: Representation of "O2-D demarcation #1".** | **Figure 2.1-2: Representation of "O2-D demarcation #2".** |

The present document version specifies an ETSI NFV profile of the O2dms protocol and data models based on the "O2- D demarcation #1".

NOTE 3: A profile based on "O2-D demarcation #2" is FFS.

## ETSI NFV framework and solutions

ETSI NFV defines the NFV-MANO architectural framework, which is conformed of a set of functional blocks, functions and interfaces enabling the deployment and management of VM and container-based VNFs. The management of virtualised and container-based resources is performed by technology and solution agnostic functional entities according to a specified set of interfaces, models and APIs.

The list of NFV-MANO architectural framework functional blocks and functions and their main responsibilities are:

* Virtualised Infrastructure Manager (VIM): responsible for the management of virtualised resources (virtual compute, virtual network and virtual storage) in the NFV Infrastructure (NFVI) and VM software images,
* Container Infrastructure Service Management (CISM): responsible for the management of container-based resources and container workloads in Container Infrastructure Service (CIS) clusters,
* Container Image Registry (CIR): responsible for the management of OS container software images,
* VNF Manager (VNFM): responsible for the management of VNFs, including their lifecycle, and provides corresponding management services based on a common set of interfaces and models regardless of the technology used for implementing the VNF, be it container or VM-based, and
* NFV Orchestrator (NFVO): responsible for the management of VNF Packages and other artifacts and the overall orchestration of NFVI resources across multiple sites (referred as well as NFVI-PoPs) and the lifecycle management of Network Services (NS).

ETSI GS NFV-IFA 010 [[8]](#_bookmark10) specifies the principles and scope of the NFV-MANO architectural framework and functional requirements of the functional blocks and functions that are part of it.

A set of groups specifications (GS) specifies the interfaces exposed and consumed by the NFV-MANO functional blocks and functions. The following are the GS of relevance for the present document:

* ETSI GS NFV-IFA 007 [[7]](#_bookmark9) specifies stage 2 interfaces and associated information model of interfaces between the NFVO and the VNFM. ETSI GS NFV-SOL 003 [[11]](#_bookmark13) specifies stage 3 protocols and data models of these interfaces based on RESTful. The Release 4 version of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) adds support for container-based VNF deployments. Refer to clause 2.4.1 for the specific version and features (VM and/or OS container-based deployments) supported by the present document.
* ETSI GS NFV-IFA 005 [[5]](#_bookmark7) specifies stage 2 interfaces and associated information models of interfaces between the NFVO and the VIM, which aim, among others features, at supporting VM-based NF deployments. ETSI GS NFV-SOL 014 [[16]](#_bookmark18) specifies a stage 3 data model of parameters for virtualised resource descriptors based on YAML.
* ETSI GS NFV-IFA 006 [[6]](#_bookmark8) specifies stage 2 interfaces and associated information models of interfaces between the VNFM and the VIM, which aim, among others features, at supporting VM-based NF deployments. ETSI GS NFV-SOL 014 [[16]](#_bookmark18) specifies a stage 3 data model of parameters for virtualised resource descriptors based on YAML.
* ETSI GS NFV-IFA 040 [[9]](#_bookmark11) specifies stage 2 interfaces and service requirements of interfaces produced by the CISM and CIR, which aim, among other features, at supporting OS container-based NF deployments. ETSI GS NFV-SOL 018 [[18]](#_bookmark20) aims at specifying stage 3 protocols and data models of these interfaces profiled on de- facto solutions. Refer to clause 2.4.1 for the specific version and features (VM and/or OS container-based deployments) supported by the present document.

In addition to the composition of the NFV-MANO architectural framework in terms of functional blocks, functions and interfaces, one of the key aspects is the definition of both design-time and runtime elements. Design-time elements basically comprise those objects and artifacts that describe the design and requirements for the deployment of entities into or using the framework. The design-time objects comprise packages and descriptors, among which the relevant ones for the present document are the VNF Package and VNF Descriptor (VNFD).

* ETSI GS NFV-IFA 011 [[12]](#_bookmark14) specifies stage 2 requirements related to VNF Package and VNF metadata and the information model of the VNFD. ETSI GS NFV-SOL 001 [[10]](#_bookmark12) specifies stage 3 data model of the VNFD (in addition to other descriptors such as Network Service Descriptor (NSD) and Physical Network Function Descriptor (PNFD)) based on TOSCA, while ETSI GS NFV-SOL 006 [[14]](#_bookmark16) specifies the same elements but based on YANG. ETSI GS NFV-SOL 004 [[13]](#_bookmark15) specifies the data model and structure of the VNF Packages.

## ETSI NFV profiling for O2dms interfaces

### General

The present and subsequent clauses describe specific aspects of the ETSI NFV profiling for the O2dms interfaces. In the present profiling version based on the "O2-D demarcation #1" introduced in clause 2.1, the functional capabilities of the NFVO, VNFM, VIM and CISM of the NFV-MANO architectural framework are mapped to the O-RAN framework as defined in clause 2.3.2.

The O2dms ETSI NFV profile based on "O2-D demarcation #1" is based on the ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The ETSI GS NFV-SOL 003 [[11]](#_bookmark13) specifies, among others, the RESTful API of the "VNF Lifecycle Management interface", which is used as the foundation for providing a DMS exposed API for the lifecycle management of NF Deployments of O-RAN. Specific API correspondences are defined in clause 2.3.3.

Finally, for the orchestration and management operations through the O2dms ETSI NFV profile, it is assumed that descriptors and packaging of corresponding NF Deployments, and consequently of O-RAN NF, are on-boarded and available to the SMO and DMS. In particular, it is expected that the descriptors and packaging for O-RAN NF contain relevant metadata such as identifiers of the design-time definitions of components/constituents of an NF Deployment, which are also information either conveyed or exposed through the API. Additional mapping information about descriptor and packaging artifacts is provided in clause 2.3.2, and reference to specific elements concerning various orchestration aspects are further described in clause 2.5.

### Conceptual mappings

The terminology and concepts used in referenced ETSI NFV specifications differs compared to the terminology used in O-RAN specifications.

Table 2.3.2-1 defines the mapping and correspondences of management concepts and terms applicable for the O2dms service APIs and to the profiling specified in the present document version.

##### Table 2.3.2-1: Concepts mapping

|  |  |  |
| --- | --- | --- |
| **O-RAN concept** | **ETSI NFV concept** | **Description** |
| O-RAN NF | NS or VNF | In the case of an O-RAN NF that is composed of multiple NF Deployments, the O-RAN NF can be realized by the NS construct abstractions; however, the NS abstraction is not visible over the O2dms, and it is only understood by the responsible SMO functions when composing the set of NF Deployments realizing the O-RAN NF.  In the case of an O-RAN NF that is composed of a single NF Deployment, the O-RAN NF maps one-to-one to a VNF in terms of lifecycle management. |
| NF Deployment | VNF | In terms of lifecycle management, a VNF instance maps one-to-one to an NF Deployment. A VNF instance is composed of one or more VNF Component (VNFC), and VNFCs can be deployed and managed based on NF Deployment items. |
| NF Deployment unit | VNFC | An NF Deployment unit, as described in Annex A of O- RAN.WG6.ORCH-USE-CASES [[2],](#_bookmark4) maps one-to-one to a deployable unit in a VNF. Deployable units of a VNF are VNFCs. As defined in ETSI GR NFV 003 [[30],](#_bookmark31) a VNFC is an internal component of a VNF providing a subset of the VNF’s functionality, with the main characteristic that a single instance of a VNFC maps to a single instance of an atomic deployable unit. An atomic deployable unit can be a single VM for hypervisor-based virtualization or one or a set of OS containers (e.g., a Pod in case of Kubernetes®) for OS-based  virtualization. See note 2. |
| O-RAN NF software package | VNF package | A VNF package contains all the required files and metadata descriptors required to validate and instantiate a VNF. The structure and format of  a VNF package file and its constituents are defined in ETSI GS NFV- SOL 004 [[13].](#_bookmark15) |
| NF Deployment descriptor | VNFD | A VNFD describes the infrastructure resource requirements, design constraints, dependencies, and the operational behaviour, including VNF lifecycle events, of the VNF. The information model of the VNFD is defined in ETSI GS NFV-IFA 011 [[12].](#_bookmark14) ETSI GS NFV-SOL 001 [[10]](#_bookmark12)  specifies a TOSCA-based data model for the VNFD, and ETSI GS NFV-SOL 006 [[14]](#_bookmark16) specifies a YANG-based data model for the VNFD. See note 1. |
| NOTE 1: Parts of the profiled ETSI GS NFV-SOL 003’s model references elements of a VNFD as sources of information also available at runtime over the interfaces, but the same specification does not specify a specific format and data model of a VNFD to be used.  NOTE 2: To fulfil connectivity requirements of a VNFC, an NF Deployment can also include internal VL and/or express requirements for external connectivity. At a high-level, a VL represents the network connectivity interconnecting two or more entities such as VNFC, VNF, and it is realized by the one or more (virtual)  networks. For more information, refer to clause 2.5.3 on "network connectivity for NF Deployments". | | |

Table 2.3.2-2 defines the mapping and correspondence of management functions and functional blocks that apply to endpoints of the O2 interface and to the profiling specified in the present document version.

**Table 2.3.2-2: Management functions/functional blocks mapping**

|  |  |  |
| --- | --- | --- |
| **O-RAN concept** | **ETSI NFV concept** | **Description** |
| DMS (NF  Deployment part) | VNFM | The VNFM functional block (FB) produced functionality maps to the DMS’s NF Deployment functionality.  The DMS produces O2dms service interfaces which are mapped to VNFM produced interfaces. See note 1. |
| DMS (virtualised resources management part) | VIM | The VIM FB produced functionality maps to the DMS’s virtualised resources management functionality. See note 2. |
| DMS (containerized workloads management part) | CISM | The CISM function produced services maps to the DMS’s containerized workloads management functionality. See note 3. |
| SMO (NFO part) (see note 4) | NFVO (network service management part) | The NFVO’s network service management functionality maps to the SMO’s NFO.  The NFO consumes O2dms service interfaces according to the profiled VNFM produced interfaces in the present document. |
| SMO (FOCOM part)  (see note 4) | NFVO (resource orchestration part) | The NFVO’s resources orchestration functionality maps to the SMO’s FOCOM. |
| NOTE 1: DMS is defined in O-RAN.WG6.O2-GA&P [[4]](#_bookmark6) as a functional block, but without specifying dedicated parts as indicated in the present table. The parts introduced in the present table are defined for the purpose of establishing a relevant mapping functional block or function from the NFV-MANO architectural framework.  NOTE 2: In ETSI NFV standards, VIM also handles part of infrastructure management, which can be mapped to O- Cloud IMS. IMS is also defined in O-RAN.WG6.O2-GA&P [[4]](#_bookmark6) and O-RAN.WG6.CAD [[3].](#_bookmark5)  NOTE 3: In ETSI NFV standards, CISM also handles part of container cluster management, which can be mapped to O-Cloud IMS. IMS is also defined in O-RAN.WG6.O2-GA&P [[4]](#_bookmark6) and O-RAN.WG6.CAD [[3]](#_bookmark5).  NOTE 4: The FOCOM and NFO, part of the SMO, are defined in O-RAN.WG6.O2-GA&P [[4].](#_bookmark6) | | |

### API mappings

The APIs specified in the present document map to the referenced standard APIs listed in table 2.3.3-1.

**Table 2.3.3-1: API mappings**

|  |  |
| --- | --- |
| **O2dms service API** | **Referenced standard API** |
| O2dms\_DeploymentLifecycle | "VNF Lifecycle Management interface" specified in clause 5 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) |
| O2dms\_DeploymentFault | "VNF Fault Management interface" specified in clause 7 of ETSI GS NFV-SOL 003 [11]. |
| O2dms\_DeploymentPerformance | "VNF Performance Management interface" specified in clause 6 of ETSI GS NFV-SOL 003 [11]. |

## Service operations

### Overview

The present document provides the protocol and data model profiling for the O2dms interface based on the ETSI NFV standards and covering a specific subset of service interfaces and operations. The defined service interfaces and operations are listed in the subsequent clauses.

The service interfaces and operations specified in the present document references the latest published Release 4 version of the ETSI GS NFV-SOL 003 [[11],](#_bookmark13) which is the one used in the protocol and data model profiling in clause 3. As indicated in clause 2.2, the Release 4 version of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) adds support for OS container-based NF deployments, and some relevant features are also documented in clause 2.5.

### NF Deployment lifecycle management

#### Instantiate NF Deployment

Clause 3.2.1 of O-RAN.WG6.ORCH-USE-CASES [[2]](#_bookmark4) describes the use case of "Instantiate Network Function on O- Cloud", which includes steps of requesting the lifecycle management operation of instantiating NF Deployments.

Related requirements are specified in clause 4.3 of O-RAN.WG6.ORCH-USE-CASES [[2].](#_bookmark4) In addition, clause 3.7.1 of O-RAN.WG6.O2-GA&P [[4]](#_bookmark6) lists for the O-Cloud the capability of deploying an O-RAN Cloudified NF instance.

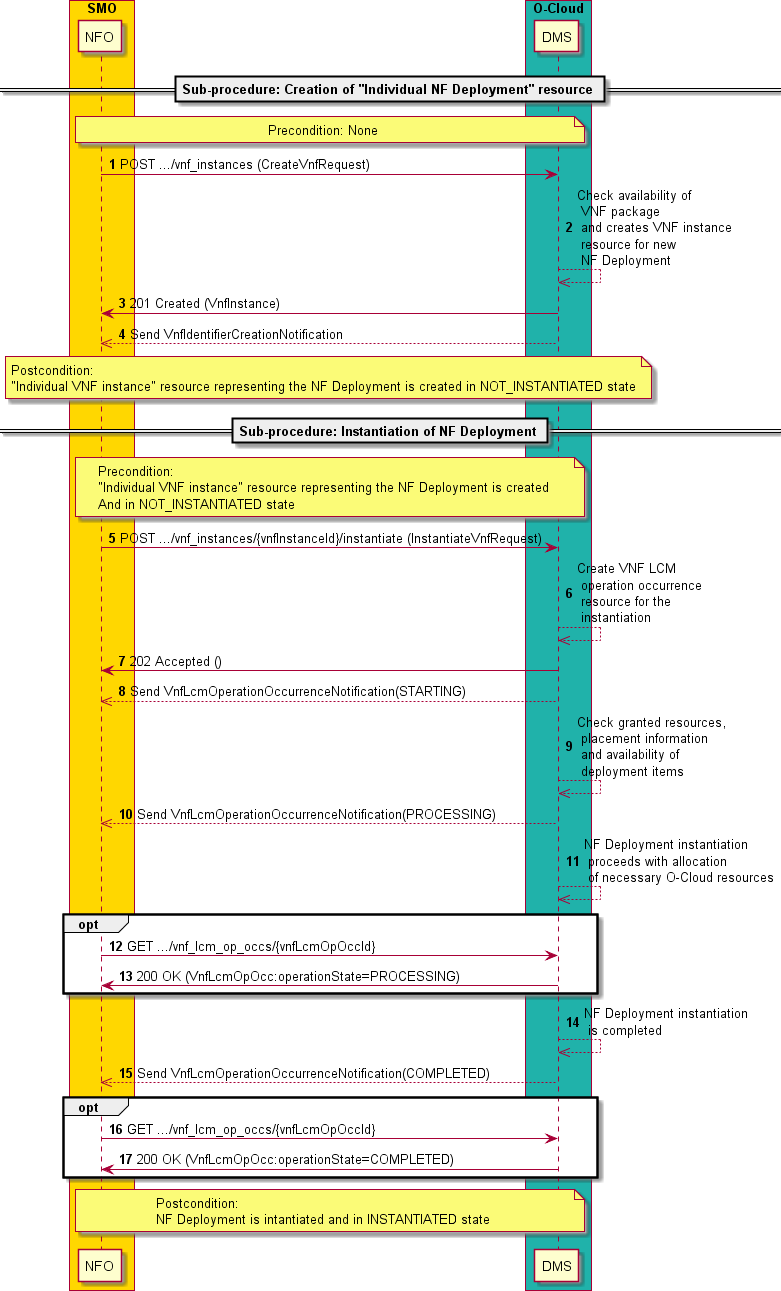
To instantiate an NF Deployment using the present O2dms API profile, the following sub-procedures are applicable:

* 1) creation of "Individual VNF instance" resource corresponding to the NF Deployment as illustrated by the "Flow of the creation of a VNF instance resource" in clause 5.3.1 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of creation of the VNF instance resource is further profiled as illustrated in figure 2.4.2.1-1.
* 2) instantiation of the NF Deployment by triggering the operation "Instantiate VNF" performed via the corresponding task resource "Instantiate VNF task" on the "Individual VNF instance" resource following the flow illustrated in "Flow of VNF lifecycle management operations triggered by task resources" in clause 5.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) the case of an NF Deployment, the sub-procedure of VNF instantiation is further profiled as illustrated in figure 2.4.2.1-1.

NOTE 1: While the profiled ETSI GS NFV-SOL 003 [[11]](#_bookmark13) VNF instantiation procedure contains steps related to VNF LCM granting exchanges between the VNFM and the NFVO of the NFV-MANO framework, the NF Deployment instantiation procedure specified in the present document version does not contain these steps.

In addition, to monitor the status of the lifecycle operation in an asynchronous mode and be able to know the status of the operation (e.g., whether in progress or in some failed status), the API consumer shall request the creation of corresponding subscriptions to NF Deployment lifecycle management events following the flow "Flow of managing subscriptions" in clause 5.3.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of a lifecycle operation by query and reading information about the VNF LCM operation occurrence, which tracks the progress of the lifecycle operation.

Figure 2.4.2.1-1 illustrates the procedure flow for the instantiation of an NF Deployment including the two sub-procedures introduced above.



##### Figure 2.4.2.1-1: Flow of NF Deployment instantiation

NOTE 2: Due to possible race conditions, the 201 response and the VnfIdentifierCreationNotification can arrive in any order at the SMO.

NOTE 3: Due to possible race conditions, the 202 response and the "STARTING" VnfLcmOperationOccurrenceNotification can arrive in any order at the SMO.

The procedure of NF Deployment instantiation consists of the following steps as illustrated in figure 2.4.2.1-1: NOTE 4: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** None.

1. The SMO sends a POST request to the "VNF instances" resource including in the message content a data structure of type "CreateVnfRequest".
2. The DMS checks the availability of the NF Deployment package (referred as VNF Package in the figure) and creates a new "Individual VNF instance" resource that represents the NF Deployment.

NOTE 5: It is expected that the NF Deployment package contents are available to the O-Cloud DMS, or that the DMS can check the availability of the package as onboarded to the SMO.

1. The DMS returns a 201 Created response containing a representation of the "Individual VNF instance" resource, which represents the NF Deployment, and provides the URI of the newly created resource in the "Location” HTTP header. See note 2.
2. The DMS sends a VNF Identifier Creation Notification to the SMO to indicate the creation of the "Individual VNF instance" resource and the associated VNF instance identifier of the NF Deployment, if the SMO is subscribed to such type of notifications. See note 2.
3. The SMO sends a POST request to the "Instantiate VNF task" resource that represents the NF Deployment instantiation lifecycle operation to be executed on the NF Deployment and includes in the message content a data structure of type "InstantiateVnfRequest".
4. The DMS creates a new "Individual VNF LCM operation occurrence" resources for the request, which represents the lifecycle operation occurrence about the NF Deployment instantiation.
5. The DMS returns a "202 Accepted" response with an empty message content and a "Location" HTTP header that points to the new "Individual VNF LCM operation occurrence" resource. See note 3.
6. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate the start of the lifecycle management operation occurrence with the "STARTING" state, if the SMO is subscribed to such type of notifications. See note 3.
7. The DMS checks the granted resources, placement information and availability of deployment items for the NF Deployment instantiation.
8. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence enters the "PROCESSING" state, if the SMO is subscribed to such type of notifications.
9. The DMS proceeds with the NF Deployment instantiation with the allocation of necessary O-Cloud resources.
10. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the ongoing lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
11. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
12. The DMS completes the NF Deployment instantiation.
13. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence is "COMPLETED", if the SMO is subscribed to such type of notifications.
14. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the completed lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
15. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".

**Postcondition:** The NF Deployment is instantiated and in "INSTANTIATE" state.

**Error handling:** If the NF Deployment lifecycle operation fails, error information is provided in the notification messages that reports about the erroneous completion of the procedure. Error information is also available in the resource that represents the "Individual VNF lifecycle management operation occurrence" related to the NF Deployment lifecycle management operation.

#### Terminate NF Deployment

Clause 3.2.5 of O-RAN.WG6.ORCH-USE-CASES [[2]](#_bookmark4) describes the use case of "Terminate Network Function on O- Cloud", which includes steps of requesting the lifecycle management operation of terminating NF Deployments.

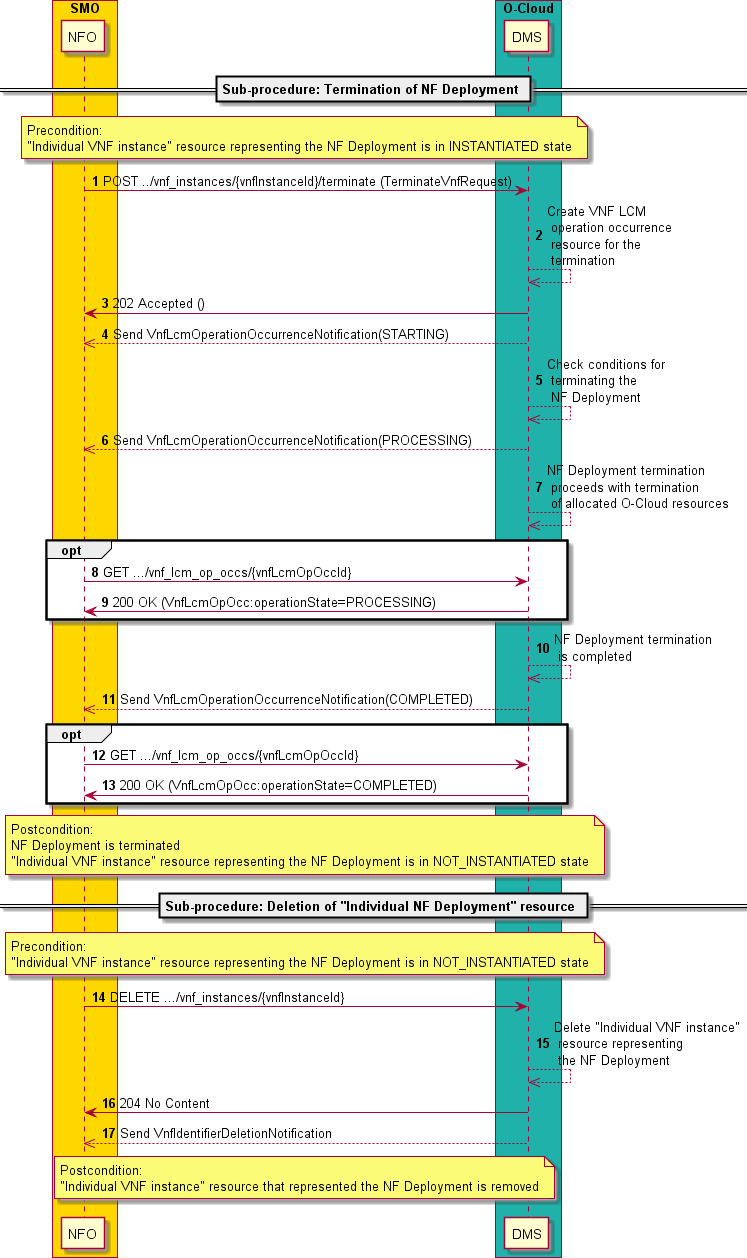
Related requirements are specified in clause 4.3 of O-RAN.WG6.ORCH-USE-CASES [[2].](#_bookmark4) In addition, clause 3.7.1 of O-RAN.WG6.O2-GA&P [[4]](#_bookmark6) lists for the O-Cloud the capability of terminating an O-RAN Cloudified NF instance.

To terminate an NF Deployment using the present O2dms API profile, the following sub-procedures are applicable:

* 1) termination of the NF Deployment by triggering the operation "Terminate VNF" performed via the corresponding task resource "Terminate VNF task" on the "Individual VNF instance" resource following the flow illustrated in "Flow of VNF lifecycle management operations triggered by task resources" in clause 5.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of VNF termination is further profiled as illustrated in figure 2.4.2.2-1.
* 2) deletion of the "Individual VNF instance" resource representing the NF Deployment as illustrated by the "Flow of the deletion of a VNF instance resource" in clause 5.3.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of deletion of the VNF instance resource is further profiled as illustrated in figure 2.4.2.2-1.

In addition, to monitor the status of the lifecycle operation in an asynchronous mode and be able to know the status of the operation (e.g., whether in progress or in some failed status), the API consumer shall request the creation of corresponding subscriptions to NF Deployment lifecycle management events following the flow "Flow of managing subscriptions" in clause 5.3.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of a lifecycle operation by query and reading information about the VNF LCM operation occurrence, which tracks the progress of the lifecycle operation.

Figure 2.4.2.2-1 illustrates the procedure flow for the termination of an NF Deployment instance including the two sub- procedures introduced above.



##### Figure 2.4.2.2-1: Flow of NF Deployment termination

NOTE 1: Due to possible race conditions, the 204 response and the VnfIdentifierDeletionNotification can arrive in any order at the SMO.

NOTE 2: Due to possible race conditions, the 202 response and the "STARTING" VnfLcmOperationOccurrenceNotification can arrive in any order at the SMO.

The procedure of NF Deployment termination consists of the following steps as illustrated in figure 2.4.2.2-1: NOTE 3: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** The "Individual VNF instance" resource representing the NF Deployment is in INSTANTIATED state.

1. The SMO sends a POST request to the "Terminate VNF task" resource that represents the NF Deployment termination lifecycle operation to be executed on the NF Deployment and includes in the message content a data structure of type "TerminateVnfRequest".
2. The DMS creates a new "Individual VNF LCM operation occurrence" resources for the request, which represents the lifecycle operation occurrence about the NF Deployment termination.
3. The DMS returns a "202 Accepted" response with an empty message content and a "Location" HTTP header that points to the new "Individual VNF LCM operation occurrence" resource. See note 2.
4. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate the start of the lifecycle management operation occurrence with the "STARTING" state, if the SMO is subscribed to such type of notifications. See note 2.
5. The DMS checks the conditions for terminating the NF Deployment.
6. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence enters the "PROCESSING" state, if the SMO is subscribed to such type of notifications.
7. The DMS proceeds with the NF Deployment termination with the termination of allocated O-Cloud resources of the NF Deployment.
8. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the ongoing lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
9. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
10. The DMS completes the NF Deployment terminates.
11. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence is "COMPLETED", if the SMO is subscribed to such type of notifications.
12. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the completed lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
13. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
14. The SMO sends a DELETE request to the "Individual VNF instance" resource that represented the NF Deployment.
15. The DMS deletes the "Individual VNF instance" resource that represented the NF Deployment and the associate NF Deployment identifier.
16. The DMS returns to the SMO a "204 No Content" response with an empty message content. See note 1.
17. The DMS sends to the SMO a VNF Identifier Deletion Notification to indicate the deletion of the "Individual VNF instance" resource that represented the NF Deployment and the associated NF Deployment identifier. See note 1.

**Postcondition:** The NF Deployment is terminated and the "Individual VNF instance" resource that represented the NF Deployment is removed.

**Error handling:** If the NF Deployment lifecycle operation fails, error information is provided in the notification messages that reports about the erroneous completion of the procedure. Error information is also available in the resource that represents the "Individual VNF lifecycle management operation occurrence" related to the NF Deployment lifecycle management operation.

#### Query NF Deployment information

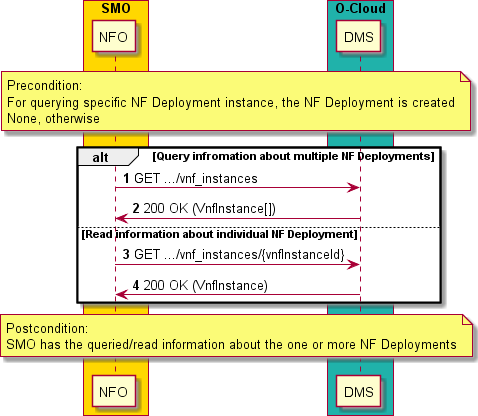
NOTE: The O-RAN.WG6.ORCH-USE-CASES [[2]](#_bookmark4) neither describes use cases about querying information of "NF Deployment" or O-RAN NF, nor specifies related functional and interface requirements. Clause

3.7.1 of the O-RAN.WG6.O2-GA&P [[4]](#_bookmark6) lists the querying information about an O-RAN Cloudified NF instance as one of the expected functional capabilities of the O-Cloud. The profiled ETSI GS NFV-SOL 003 provides the capabilities to query information about "NF Deployments" by enabling the necessary HTTP methods on corresponding REST resources. This provides a more complete set of functionalities to the SMO to query information about NF Deployments that have been instantiated. The present document version includes such a functionality and service operation, even though it is not a target for the O-RAN WG6’s "2022 March Train".

To query information about one or more NF Deployment instances using the present O2dms API profile, the following sub-procedures are applicable:

* To query information about multiple NF Deployment instances, execute a GET method on the "VNF instances" resource following the flow illustrated in "Flow of the Query VNF operation" in clause 5.3.5 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of query information of multiple VNF instances is further profiled as illustrated in figure 2.4.2.3-1.
* To query information about a specific NF Deployment instance, execute a GET method on the "Individual VNF instance" resource following the flow illustrated in "Flow of the Query VNF operation" in clause 5.3.5 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of query information of a particular VNF instance is further profiled as illustrated in figure 2.4.2.3-1.

Figure 2.4.2.3-1 illustrates the procedure flow for querying information about a specific NF Deployment instance or multiple ones.



##### Figure 2.4.2.3-1: Flow of NF Deployment termination

The procedure of query NF Deployment information consists of the following steps as illustrated in figure 2.4.2.3-1: NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** For querying a specific NF Deployment, the NF Deployment is created. None, otherwise.

1. If the SMO intends to query information about multiple NF Deployments, it sends a GET request to the "VNF instances" resource that contains the set of NF Deployments.
2. The DMS returns a "200 OK" response to the SMO and includes zero or more data structures of type "VnfInstance" in the message content, depending on the query result.
3. If the SMO intends to read information about a particular NF Deployment, it sends a GET a request to the "Individual VNF instance" resource with the appropriate NF Deployment identifier.
4. The DMS returns a "200 OK" response to the SMO and includes one data structure of type "VnfInstance" in the message content.

**Postcondition:** The SMO has the queried/read information about the one or more NF Deployments.

**Error handling:** In case of failure, appropriate error information is provided in the response.

#### Heal NF Deployment

* + - * 1. Overview

Clause 3.6 of O-RAN.WG6.ORCH-USE-CASES [2] describes use cases about recovery at different level. Clause 3.6.2 of O-RAN.WG6.ORCH-USE-CASES [2] details the "NF Deployment level reset use case". According to the use case, when some fatal error is identified which is expected (but not guaranteed) it can be resolved by resetting the NF Deployment, the SMO can send a specific reset request to the DMS over the O2dms interface. In addition, clause 3.7.1 of O-RAN.WG6.O2-GA&P [[4]](#_bookmark6) lists for the O-Cloud the capability of healing an O-RAN Cloudified NF instance. From

the ETSI NFV profile perspective, the (part of) reset capabilities can be performed via a "Healing VNF instance" operation.

As defined in ETSI GR NFV 003 [[30],](#_bookmark31) VNF healing is a lifecycle procedure that includes all virtualisation-related corrective actions to repair a faulty VNF, and/or its VNFC instances and internal VNF Virtual Links (refer to concepts mapping in clause 2.3.2). The procedure is intended to leverage the resource virtualisation management capabilities such as, starting/stopping/restarting VMs, re-allocating VMs or containerized workloads, and other possible actions, as means to recover from faulty conditions of the VNF. Furthermore, VNF healing can be performed on-demand when a consumer explicitly requests to the producer of the lifecycle management operation or automatically triggered by the producer, also referred as "auto-healing", based on various inputs such as indicators, alarms and KPIs processed from the VNF and/or infrastructure.

NOTE: As described above, VNF healing considers virtualisation-related corrective actions, and it does not cover other possible VNF correction actions at the application level.

The healing capabilities of a VNF are defined by the VNF provider through the VNF descriptors. The specific list of parameters to convey the trigger of what healing action to perform are also defined through VNF descriptors (see generic information modelling in clause 7.1.5.6 of ETSI GS NFV-IFA 011 [[12]](#_bookmark14)). Healing actions can be different between VNFs and there are no specific means specified by the referenced ETSI NFV specifications to determine what actions are applicable to a VNF instance and not to another VNF instance as this are regarded to be VNF-specific and implementation dependent. From an interface perspective, the VNF healing operation enables the consumer to provide the list of expected parameters and their values as defined in the actual descriptors delivered by the VNF provider in the VNF package. Dedicated LCM scripts (see clause 7.1.13 in ETSI GS NFV-IFA 011 [[12]](#_bookmark14)) either defined via the VNF descriptors or as additional artifacts implement the logic of the concrete procedure actions. The details of the request inputs to trigger the NF Deployment healing are specified in clause 3.2.6.2.9.

* + - * 1. On-demand healing

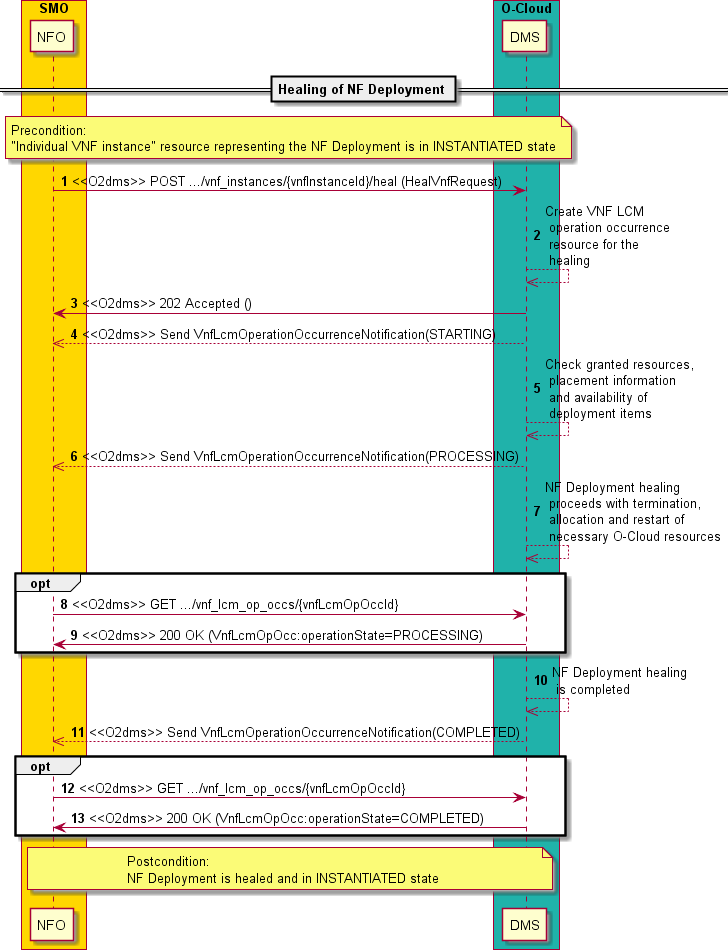
To perform the on-demand heal of an NF Deployment using the present O2dms API profile, the following sub- procedures are applicable:

- 1) healing of the NF Deployment by triggering the operation "Heal VNF" performed via the corresponding task resource "Heal VNF task" on the "Individual VNF instance" resource following the flow illustrated in "Flow of VNF lifecycle management operations triggered by task resources" in clause 5.3.3 of ETSI GS NFV- SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of VNF healing is further profiled as illustrated in figure 2.4.2.4.2-1.

NOTE 1: While the profiled ETSI GS NFV-SOL 003 [[11]](#_bookmark13) VNF healing procedure contains steps related to VNF LCM granting exchanges between the VNFM and the NFVO of the NFV-MANO framework, the NF Deployment healing procedure specified in the present document version does not contain these steps.

In addition, to monitor the status of the lifecycle operation in an asynchronous mode and be able to know the status of the operation (e.g., whether in progress or in some failed status), the API consumer shall request the creation of corresponding subscriptions to NF Deployment lifecycle management events following the flow "Flow of managing subscriptions" in clause 5.3.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of a lifecycle operation by query and reading information about the VNF LCM operation occurrence, which tracks the progress of the lifecycle operation.

Figure 2.4.2.4.2-1 illustrates the procedure flow for the healing of an NF Deployment including the sub-procedure introduced above.



##### Figure 2.4.2.4.2-1: Flow of on-demand NF Deployment healing

NOTE 2: Due to possible race conditions, the 202 response and the "STARTING" VnfLcmOperationOccurrenceNotification can arrive in any order at the SMO.

The procedure of on-demand NF Deployment healing consists of the following steps as illustrated in figure 2.4.2.4.2-1:

NOTE 3: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** "Individual VNF instance" resource representing the NF Deployment is in INSTANTIATED state.

1. The SMO sends a POST request to the "Heal VNF task" resource that represents the NF Deployment healing lifecycle operation to be executed on the NF Deployment and includes in the message content a data structure of type "HealVnfRequest".
2. The DMS creates a new "Individual VNF LCM operation occurrence" resources for the request, which represents the lifecycle operation occurrence about the NF Deployment healing.
3. The DMS returns a "202 Accepted" response with an empty message content and a "Location" HTTP header that points to the new "Individual VNF LCM operation occurrence" resource. See note 2.
4. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate the start of the lifecycle management operation occurrence with the "STARTING" state, if the SMO is subscribed to such type of notifications. See note 2.
5. The DMS checks the granted resources, placement information and availability of deployment items for the NF Deployment healing. The DMS also checks the specific actions (e.g., terminate and create new deployment item, restart a deployment item, etc.) to be performed for the healing based on the information contained in the descriptors of the NF Deployment.
6. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence enters the "PROCESSING" state, if the SMO is subscribed to such type of notifications.
7. The DMS proceeds with the NF Deployment healing with the allocation, termination and restart of necessary O-Cloud resources.
8. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the ongoing lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
9. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
10. The DMS completes the NF Deployment healing.
11. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence is "COMPLETED", if the SMO is subscribed to such type of notifications.
12. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the completed lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
13. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".

**Postcondition:** The NF Deployment is healed and in "INSTANTIATED" state.

**Error handling:** If the NF Deployment lifecycle operation fails, error information is provided in the notification messages that reports about the erroneous completion of the procedure. Error information is also available in the resource that represents the "Individual VNF lifecycle management operation occurrence" related to the NF Deployment lifecycle management operation.

* + - * 1. Auto-healing

To perform the auto-healing of an NF Deployment using the present O2dms API profile, the following sub-procedures are applicable:

* 1) auto-healing of the NF Deployment triggered by the DMS based on the detection of the healing condition following the flow illustrated in "Flow of automatic invocation of VNF scaling and VNF healing" in clause

5.3.4 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of VNF healing is further profiled as illustrated in figure 2.4.2.4.3-1.

NOTE 1: While the profiled ETSI GS NFV-SOL 003 [[11]](#_bookmark13) VNF auto-healing procedure contains steps related to VNF LCM granting exchanges between the VNFM and the NFVO of the NFV-MANO framework, the NF Deployment healing procedure specified in the present document version does not contain these steps.

For the DMS to trigger the auto-healing, two conditions shall be met:

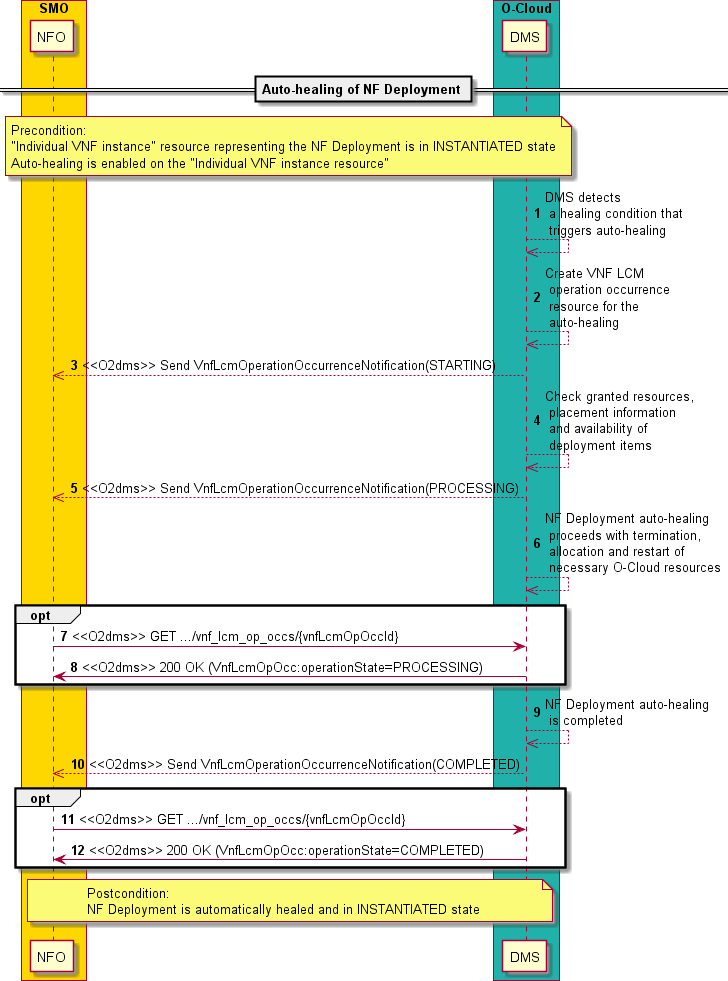
* The auto-healing is enabled on the NF Deployment. This, in the ETSI GS NFV-SOL 003 based profile, is controlled via setting the corresponding attribute value of "isAutohealEnabled" in the "vnfConfigurableProperties" of the "VnfInstance" data type that represents the NF Deployment instance (see clause 3.2.6.2.2) and that the DMS manages. Changing the value of the attribute is made upon request by the API consumer (e.g., the SMO) by "patching" the corresponding "Individual VNF instance" resource of the NF Deployment. Patching and NF Deployment does not require re-instantiating the NF Deployment.

NOTE 2: PATCHing of the "Individual VNF instance" resource, which represents the NF Deployment, to support modifying information of an NF Deployment is supported as indicated in clause 3.2.3.

* The healing condition is met, e.g., by monitoring alarms related to faults of virtualised resources/containerized workloads part of the NF Deployment, NF Deployment-specific indicators like "health degradation" and/or performance data (via PM jobs or Thresholds) of the NF Deployment such as degradation of input/output packet transfer on connection points of the NF Deployment. The criterion for fulfilling the condition is NF Deployment-specific and implementation dependent and not specified in the present document.

In addition, to monitor the status of the lifecycle operation in an asynchronous mode and be able to know the status of the operation (e.g., whether in progress or in some failed status), the API consumer shall request the creation of corresponding subscriptions to NF Deployment lifecycle management events following the flow “Flow of managing subscriptions” in clause 5.3.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of a lifecycle operation by query and reading information about the VNF LCM operation occurrence, which tracks the progress of the lifecycle operation. The API consumer can know that the operation has been triggered automatically as the attribute "isAutomaticInvocation" of the "VnfLcmOperationOccurrenceNotification" (see clause 3.2.5.2) and the “VnfLcmOpOcc” (see clause 3.2.6.2.6) is set to "true".

Figure 2.4.2.4.3-1 illustrates the procedure flow for the auto-healing of an NF Deployment including the sub-procedure introduced above.



##### Figure 2.4.2.4.3-1: Flow of NF Deployment auto-healing

The procedure of NF Deployment auto-healing consists of the following steps as illustrated in figure 2.4.2.4.3-1:

NOTE 3: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** "Individual VNF instance" resource representing the NF Deployment is in INSTANTIATED state and auto-healing is enabled on the "Individual VNF instance resource".

1. The DMS detects a healing condition that triggers the auto-healing of the NF Deployment and selects the appropriate parameters for the operation to proceed based on the information contained in the descriptors of the NF Deployment or operational policies.
2. The DMS creates a new "Individual VNF LCM operation occurrence" resources for the operation, which represents the lifecycle operation occurrence about the NF Deployment auto-healing.
3. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate the start of the lifecycle management operation occurrence with the "STARTING" state, if the SMO is subscribed to such type of notifications.
4. The DMS checks the granted resources, placement information (see clause 2.5.2 and 3.2.6.5 which document related information) and availability of deployment items for the NF Deployment auto-healing. The DMS also checks the specific actions (e.g., terminate and create new deployment item, restart a deployment item, etc.) to be performed for the auto-healing based on the information contained in the descriptors of the NF Deployment.
5. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence enters the "PROCESSING" state, if the SMO is subscribed to such type of notifications.
6. The DMS proceeds with the NF Deployment auto-healing with the allocation, termination and restart of necessary O-Cloud resources.
7. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the ongoing lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
8. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
9. The DMS completes the NF Deployment auto-healing.
10. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence is "COMPLETED", if the SMO is subscribed to such type of notifications.
11. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the completed lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
12. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".

**Postcondition:** The NF Deployment is automatically healed and in "INSTANTIATED" state.

**Error handling:** If the NF Deployment lifecycle operation fails, error information is provided in the notification messages that reports about the erroneous completion of the procedure. Error information is also available in the resource that represents the "Individual VNF lifecycle management operation occurrence" related to the NF Deployment lifecycle management operation.

#### Scale NF Deployment

* + - * 1. Overview

Clauses 3.2.2 and 3.2.3 of O-RAN.WG6.ORCH-USE-CASES [[2]](#_bookmark4) describe use cases about scaling out and scaling in of NF, respectively. According to the use cases, NFs can be scaled out/in by adding/removing resources from the NF Deployments that conform the NF. In the use cases, scaling is performed based on a management request from the SMO, who sends a specific request to the DMS over the O2dms interface to scale the NF Deployment.

As defined in ETSI GR NFV 003 [[30],](#_bookmark31) VNF scaling is the ability to dynamically extend/reduce resources of a VNF, and it can include scaling up/down and scaling out/in. In particular, the scaling out/in refers to the ability to scale by adding/removing resource instances to a VNF. For instance, in the case of a VM-based VNF, scaling out will add instances of VM to the VNF, while scaling in will remove VM instances to the VNF; and in the case of a container- based VNF, scaling out will add instances of one or a set of OS containers (aka Pod), while scaling in will remove instances of one or a set of OS containers.

The scaling capabilities of a VNF are defined by the VNF provider through the VNF descriptors. The specific list of parameters to convey the trigger of what scaling aspect and levels can be performed are also defined through VNF descriptors (see generic information modelling in clause 7.1.5.4 and 7.1.5.5 of ETSI GS NFV-IFA 011 [[12]](#_bookmark14)). The scaling can also be handled in two forms:

* Scale VNF: scaling a VNF by defining which aspects to scale and by which number of scaling steps, and
* Scale VNF to Level: scaling a VNF to a specified scaling or VNF instantiation level.

More information about the concepts of scaling a VNF instance is available in clause B.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The following clauses document the service operation of NF Deployment scaling considering the mappings of NF Deployment to VNF according to the present document.

Scaling of an NF Deployment can be triggered in two ways:

* Scaling based on a management request: in this case, the DMS receives and processes a management request sent by the API consumer (SMO). The corresponding procedure is further specified in clause 2.4.2.5.2.
* Auto-scaling: in this case, the DMS automatically invokes the scaling of the NF Deployment based on some criteria, such as changes in values of monitoring parameters (e.g., clause 7.1.11 of ETSI GS NFV-IFA 011 [[12]](#_bookmark14) specifies capabilities to define monitoring parameters and Indicators of a VNF) that can be monitored by the DMS on the NF Deployment. The corresponding procedure is further specified in clause 2.4.2.5.3.
  + - * 1. Scaling based on management request

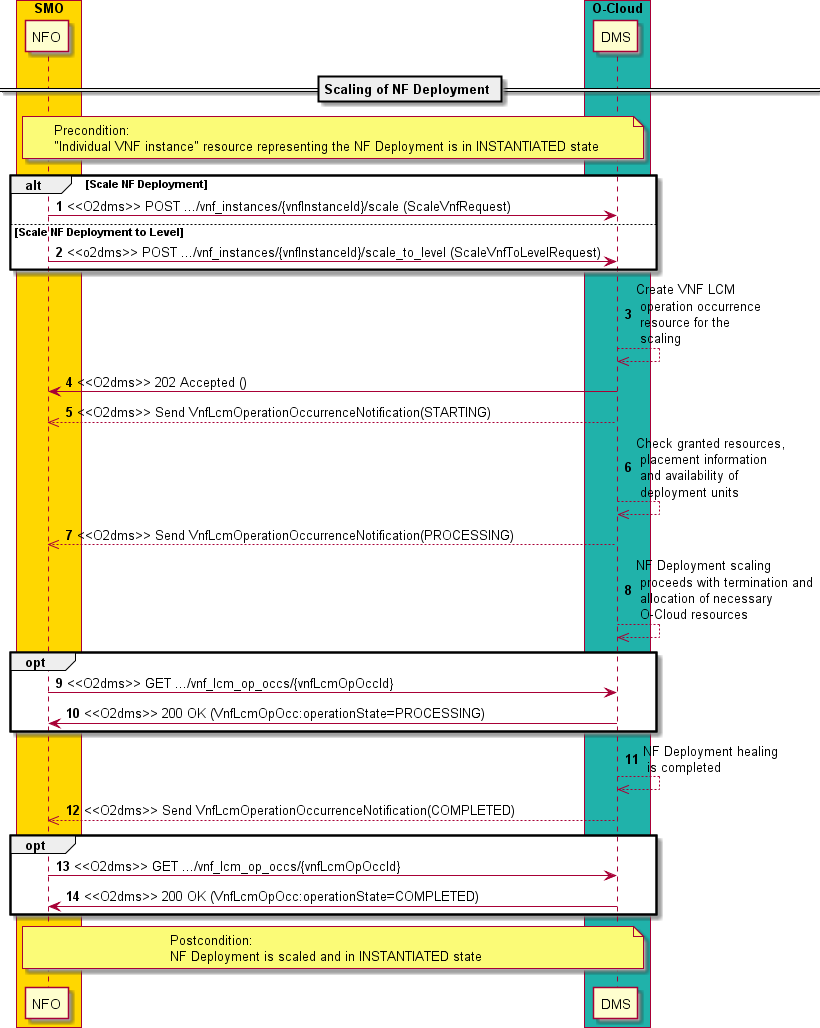
To perform scaling of an NF Deployment based on an explicit management request using the present O2dms API profile, the following sub-procedures are applicable:

- 1) scaling of the NF Deployment by triggering the operation "Scale VNF" performed via the corresponding task resource "Scale VNF task" or the operation "Scale VNF to Level" performed via the corresponding task resource "Scale VNF to Level task" on the "Individual VNF instance" resource following the flow illustrated in "Flow of VNF lifecycle management operations triggered by task resources" in clause 5.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of VNF scaling is further profiled as illustrated in figure 2.4.2.5.2-1.

NOTE 1: While the profiled ETSI GS NFV-SOL 003 [[11]](#_bookmark13) VNF scaling procedure contains steps related to VNF LCM granting exchanges between the VNFM and the NFVO of the NFV-MANO framework, the NF Deployment healing procedure specified in the present document version does not contain these steps.

In addition, to monitor the status of the lifecycle operation in an asynchronous mode and be able to know the status of the operation (e.g., whether in progress or in some failed status), the API consumer shall request the creation of corresponding subscriptions to NF Deployment lifecycle management events following the flow "Flow of managing subscriptions" in clause 5.3.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of a lifecycle operation by query and reading information about the VNF LCM operation occurrence, which tracks the progress of the lifecycle operation.

Figure 2.4.2.5.2-1 illustrates the procedure flow for the scaling of an NF Deployment including the sub-procedure introduced above.



##### Figure 2.4.2.5.2-1: Flow of NF Deployment scaling based on management request

NOTE 2: Due to possible race conditions, the 202 response and the "STARTING" VnfLcmOperationOccurrenceNotification can arrive in any order at the SMO.

The procedure of NF Deployment scaling based on management request consists of the following steps as illustrated in figure 2.4.2.5.2-1:

NOTE 3: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** "Individual VNF instance" resource representing the NF Deployment is in INSTANTIATED state.

1. The SMO sends a POST request to the "Scale VNF task" resource that represents the NF Deployment scaling lifecycle operation to be executed on the NF Deployment and includes in the message content a data structure of type "ScaleVnfRequest".
2. Alternatively, to step 1, the SMO sends a POST request to the "Scale VNF to Level task" resource that represents the NF Deployment scaling to level lifecycle operation to be executed on the NF Deployment and includes in the message content a data structure of type "ScaleVnfToLevelRequest".
3. The DMS creates a new "Individual VNF LCM operation occurrence" resources for the request, which represents the lifecycle operation occurrence about the NF Deployment scaling.
4. The DMS returns a "202 Accepted" response with an empty message content and a "Location" HTTP header that points to the new "Individual VNF LCM operation occurrence" resource. See note 2.
5. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate the start of the lifecycle management operation occurrence with the "STARTING" state if the SMO is subscribed to such type of notifications. See note 2.
6. The DMS checks the granted resources, placement information and availability of deployment items for the NF Deployment scaling. The DMS also checks the specific actions (e.g., terminate and create new deployment unit, restart a deployment unit, etc.) to be performed for the scaling based on the information contained in the descriptors of the NF Deployment.
7. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence enters the "PROCESSING" state if the SMO is subscribed to such type of notifications.
8. The DMS proceeds with the NF Deployment scaling with the allocation, termination, and restart of necessary O-Cloud resources.
9. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the ongoing lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
10. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
11. The DMS completes the NF Deployment scaling.
12. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence is "COMPLETED” if the SMO is subscribed to such type of notifications.
13. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the completed lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
14. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".

**Postcondition:** The NF Deployment is scaled and in "INSTANTIATED" state.

**Error handling:** If the NF Deployment lifecycle operation fails, error information is provided in the notification messages that reports about the erroneous completion of the procedure. Error information is also available in the resource that represents the "Individual VNF lifecycle management operation occurrence" related to the NF Deployment lifecycle management operation.

* + - * 1. Auto-scaling

To perform the auto-scaling of an NF Deployment using the present O2dms API profile, the following sub-procedures are applicable:

* 1) auto-scaling of the NF Deployment triggered by the DMS based on the detection of the scaling condition following the flow illustrated in "Flow of automatic invocation of VNF scaling and VNF healing" in clause

5.3.4 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of VNF scaling is further profiled as illustrated in figure 2.4.2.5.3-1.

NOTE 1: While the profiled ETSI GS NFV-SOL 003 [[11]](#_bookmark13) VNF auto-scaling procedure contains steps related to VNF LCM granting exchanges between the VNFM and the NFVO of the NFV-MANO framework, the NF Deployment scaling procedure specified in the present document version does not contain these steps.

For the DMS to trigger the auto-scaling, two conditions shall be met:

* The auto-scaling is enabled on the NF Deployment. This, in the ETSI GS NFV-SOL 003 based profile, is controlled via setting the corresponding attribute value of "isAutoscaleEnabled" in the "vnfConfigurableProperties" of the "VnfInstance" data type that represents the NF Deployment instance (see clause 3.2.6.2.2) and that the DMS manages. Changing the value of the attribute is made upon request by the API consumer (e.g., the SMO) by "patching" the corresponding "Individual VNF instance" resource of the NF Deployment. Patching an NF Deployment does not require re-instantiating the NF Deployment.

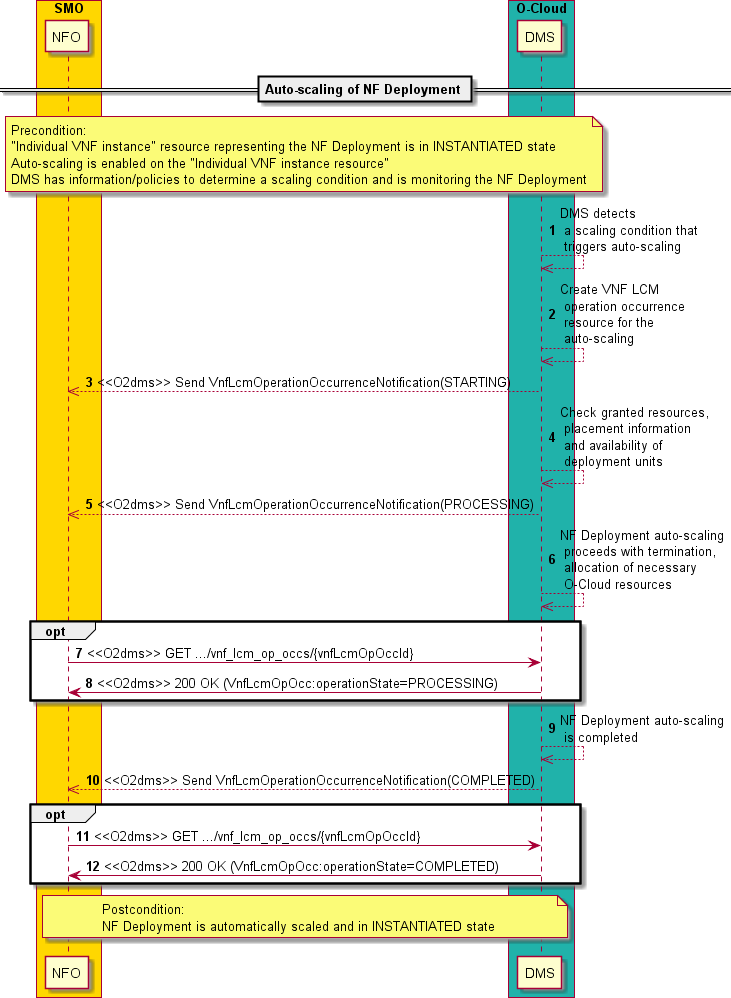
NOTE 2: PATCHing of the "Individual VNF instance" resource, which represents the NF Deployment, to support modifying information of an NF Deployment is supported as indicated in clause 3.2.3.

* The scaling condition is met, e.g., by monitoring performance metrics of virtualised resources/containerized workloads part of the NF Deployment, NF Deployment-specific indicators like "capacity usage" and/or performance data (via PM jobs or Thresholds) of the NF Deployment such as degradation of input/output packet transfer on connection points of the NF Deployment. The criterion for fulfilling the condition is NF Deployment-specific and implementation dependent and not specified in the present document. As a pre- condition, the DMS is expected to have the necessary information and/or policies to determine the fulfilment of the auto-scaling criterion, e.g., as contained in the NF Deployment descriptor when such descriptor is made available to the DMS. See also note 5.

In addition, to monitor the status of the lifecycle operation in an asynchronous mode and be able to know the status of the operation (e.g., whether in progress or in some failed status), the API consumer shall request the creation of corresponding subscriptions to NF Deployment lifecycle management events following the flow "Flow of managing subscriptions" in clause 5.3.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of a lifecycle operation by query and reading information about the VNF LCM operation occurrence, which tracks the progress of the lifecycle operation. The API consumer can know that the operation has been triggered automatically as the attribute "isAutomaticInvocation" of the "VnfLcmOperationOccurrenceNotification" (see clause 3.2.5.2) and the "VnfLcmOpOcc" (see clause 3.2.6.2.6) is set to "true".

NOTE 3: The referenced current version of the O-RAN.WG6.ORCH-USE-CASES [[2]](#_bookmark4) does not describe use cases about O-RAN NF auto-scaling. The present procedure flow might have to be updated in future versions of the present document in case additional or changed requirements are derived in case O-RAN NF auto- scaling use cases are described in future versions of the referenced document.

Figure 2.4.2.5.3-1 illustrates the procedure flow for the auto-scaling of an NF Deployment including the sub-procedure introduced above.



##### Figure 2.4.2.5.3-1: Flow of NF Deployment auto-scaling

The procedure of NF Deployment auto-healing consists of the following steps as illustrated in figure 2.4.2.5.3-1:

NOTE 4: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** "Individual VNF instance" resource representing the NF Deployment is in INSTANTIATED state, auto- healing is enabled on the "Individual VNF instance resource", and DMS has information/policies to determine a scaling condition and is monitoring the NF Deployment.

NOTE 5: The provision of the information/policy criterion for the auto-scaling is not in scope of the present procedure flow. It is expected the SMO provides such criterion to the DMS. Clause 3.2.2.2 of O-RAN- WG6.ORCH-USE-CASES [[2]](#_bookmark4) indicates that SMO has policies (or configuration) related to NF capacity modification.

1. The DMS detects a scaling condition that triggers the auto-scaling of the NF Deployment and selects the appropriate parameters for the operation to proceed based on the information contained in the descriptors of the NF Deployment or operational policies.
2. The DMS creates a new "Individual VNF LCM operation occurrence" resources for the operation, which represents the lifecycle operation occurrence about the NF Deployment auto-scaling.
3. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate the start of the lifecycle management operation occurrence with the "STARTING" state if the SMO is subscribed to such type of notifications.
4. The DMS checks the granted resources, placement information (see clause 2.5.2 and 3.2.6.5 which document related information) and availability of deployment items for the NF Deployment auto-scaling. The DMS also checks the specific actions (e.g., terminate and create new deployment unit, restart a deployment unit, etc.) to be performed for the auto-scaling based on the information contained in the descriptors of the NF Deployment.
5. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence enters the "PROCESSING" state if the SMO is subscribed to such type of notifications.
6. The DMS proceeds with the NF Deployment auto-scaling with the allocation, termination, and restart of necessary O-Cloud resources.
7. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the ongoing lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
8. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
9. The DMS completes the NF Deployment auto-scaling.
10. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence is "COMPLETED” if the SMO is subscribed to such type of notifications.
11. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the completed lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
12. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".

**Postcondition:** The NF Deployment is automatically scaled and in "INSTANTIATED" state.

**Error handling:** If the NF Deployment lifecycle operation fails, error information is provided in the notification messages that reports about the erroneous completion of the procedure. Error information is also available in the resource that represents the "Individual VNF lifecycle management operation occurrence" related to the NF Deployment lifecycle management operation.

#### Change external connectivity of an NF Deployment

With the change external connectivity operation, the following types of connectivity changes of an NF Deployment are supported:

* Disconnect external CPs from the NF Deployment that are connected to a particular external VL and connect them to a different external VL.
* Disconnect external CP instances of an NF Deployment that represent subports in a trunk and delete them. The external CP that represents the "parent" port on the trunk cannot be neither deleted, nor disconnected, and it can only reconnect to a different external VL.
* Change the connectivity parameters of existing external CP, e.g., changing L3 addresses.
* Create new external CP that represent subports in a trunk and connect them to a particular external VL. The operation does not support creating the parent port of the trunk.

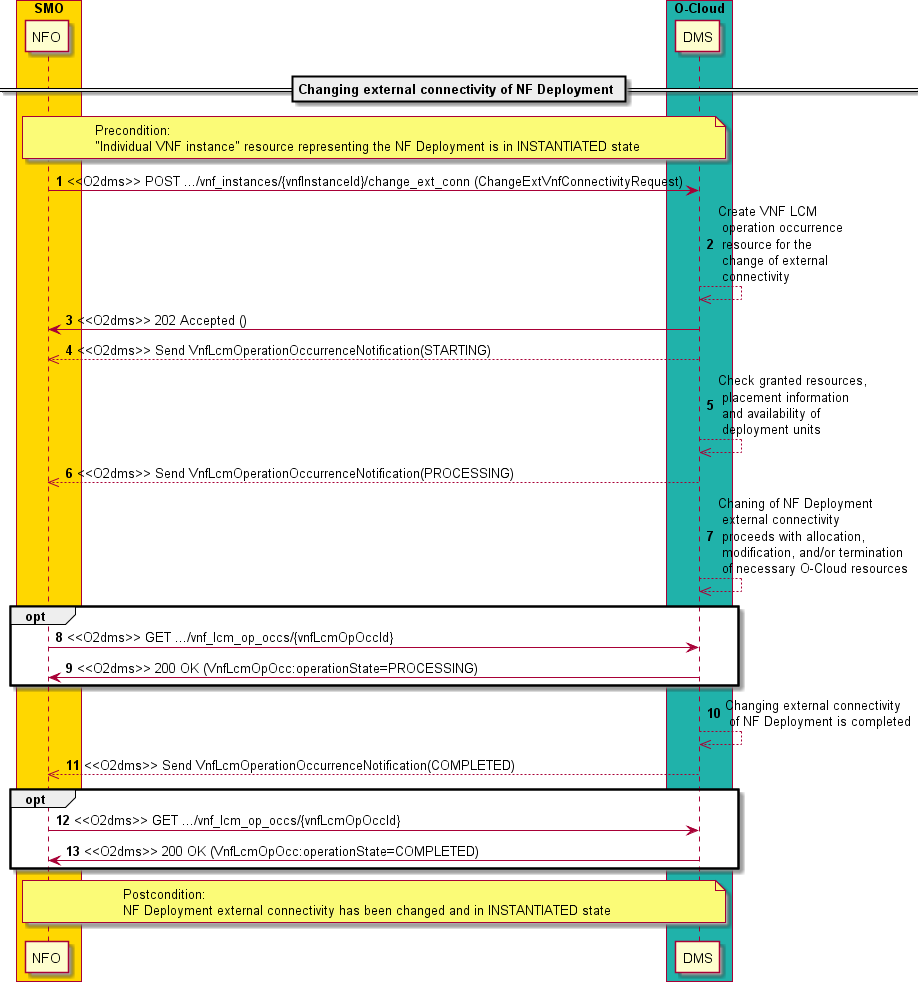
To perform the change external connectivity of an NF Deployment using the present O2dms API profile, the following sub-procedures are applicable:

* 1) changing external connectivity of the NF Deployment by triggering the operation "Change external VNF connectivity" performed via the corresponding task resource "Change external VNF connectivity task" on the "Individual VNF instance" resource following the flow illustrated in "Flow of VNF lifecycle management operations triggered by task resources" in clause 5.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of VNF scaling is further profiled as illustrated in figure 2.4.2.6-1.

NOTE 1: While the profiled ETSI GS NFV-SOL 003 [[11]](#_bookmark13) changing external VNF connectivity procedure contains steps related to VNF LCM granting exchanges between the VNFM and the NFVO of the NFV-MANO framework, the NF Deployment healing procedure specified in the present document version does not contain these steps.

In addition, to monitor the status of the lifecycle operation in an asynchronous mode and be able to know the status of the operation (e.g., whether in progress or in some failed status), the API consumer shall request the creation of corresponding subscriptions to NF Deployment lifecycle management events following the flow "Flow of managing subscriptions" in clause 5.3.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of a lifecycle operation by query and reading information about the VNF LCM operation occurrence, which tracks the progress of the lifecycle operation.

Figure 2.4.2.6-1 illustrates the procedure flow for the changing external connectivity of an NF Deployment including the sub-procedure introduced above.



##### Figure 2.4.2.6-1: Flow of changing external connectivity of NF Deployment

NOTE 2: Due to possible race conditions, the 202 response and the "STARTING" VnfLcmOperationOccurrenceNotification can arrive in any order at the SMO.

The procedure of changing external connectivity of NF Deployment consists of the following steps as illustrated in figure 2.4.2.6-1:

NOTE 3: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** "Individual VNF instance" resource representing the NF Deployment is in INSTANTIATED state.

1. The SMO sends a POST request to the "Change external VNF connectivity task" resource that represents the changing external NF Deployment connectivity lifecycle operation to be executed on the NF Deployment and includes in the message content a data structure of type "ChangeExtVnfConnectivityRequest".
2. The DMS creates a new "Individual VNF LCM operation occurrence" resources for the request, which represents the lifecycle operation occurrence about the change of external connectivity of the NF Deployment.
3. The DMS returns a "202 Accepted" response with an empty message content and a "Location" HTTP header that points to the new "Individual VNF LCM operation occurrence" resource. See note 2.
4. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate the start of the lifecycle management operation occurrence with the "STARTING" state if the SMO is subscribed to such type of notifications. See note 2.
5. The DMS checks the granted resources, placement information and availability of deployment items for the change of external NF Deployment connectivity. The DMS also checks the specific actions (e.g., allocate, modify, or terminate existing O-Cloud resources) to be performed for the changing of external connectivity based on the information contained in the descriptors of the NF Deployment and the current runtime information held about the NF Deployment.
6. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence enters the "PROCESSING" state if the SMO is subscribed to such type of notifications.
7. The DMS proceeds with the change of external NF Deployment connectivity with the allocation, modification and/or termination of necessary O-Cloud resources.
8. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the ongoing lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
9. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
10. The DMS completes the change of external NF Deployment connectivity.
11. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence is "COMPLETED” if the SMO is subscribed to such type of notifications.
12. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the completed lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
13. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".

**Postcondition:** The NF Deployment external connectivity has been changed and is in "INSTANTIATED" state.

**Error handling:** If the NF Deployment lifecycle operation fails, error information is provided in the notification messages that reports about the erroneous completion of the procedure. Error information is also available in the resource that represents the "Individual VNF lifecycle management operation occurrence" related to the NF Deployment lifecycle management operation.

#### Change current NF Deployment package

Clause 3.2.4 of O-RAN.WG6.ORCH-USE-CASES [[2]](#_bookmark4) describes the use case of "Software upgrade of NF", which includes steps of deploying NF with new software and terminating the previously deployed NF in a build-and-replace software upgrade approach. Other possible approaches are mentioned as well, such as "break-fix" or "in-place upgrade". Related requirements are specified in clauses 4.1 and 4.3 of O-RAN.WG6.ORCH-USE-CASES [[2].](#_bookmark4) In the general capabilities of O-Cloud Deployment Lifecycle Management listed in clause 3.7.2 of O-RAN.WG6.O2-GA&P [[4],](#_bookmark6) a reference is also made to the capability of O-Cloud supporting upgrading of any or all components of an O-RAN Cloudified NF instance.

The functionality of software modification of a VNF are covered in the ETSI NFV framework by a specified VNF lifecycle management operation named "Change current VNF package". This operation provides "in-place upgrade" capabilities, i.e., a VNF’s software can be modified without need to perform a fresh new deployment. In the case of ETSI NFV framework, a "build-and-replace" approach is covered by reusing the VNF lifecycle management operations

of "VNF instantiation" and "VNF termination". Both operations are profiled in the present document in the context of managing the lifecycle of NF Deployments, refer to clauses [0](#_bookmark52) and [0,](#_bookmark53) respectively.

NOTE 1: In the case of "build-and-replace", it is assumed that the SMO is in control of when the "old" NF Deployment instance can be terminated, e.g., by monitoring data collected via the O1.

NOTE 2: "Current NF Deployment package" means the NF Deployment (software) package from which a given NF Deployment instance is based on at a specific point in time. For instance, at a given point in time NF Deployment instance #A1 is deployed and runs based on the content from NF Deployment package #A.v1. By changing the "current NF Deployment package", the lifecycle process changes the NF Deployment instance to run based on the content from another NF Deployment package. Following with the example, for instance, the NF Deployment instance #A1 is updated to run based on the content of NF Deployment package #A.v2.

With the change of current package of an NF Deployment, the following types of changes/modifications on an NF Deployment are supported:

* Changes of the NF Deployment virtualised resources and/or containerized workloads, such as their requirements in terms of infrastructure resources or the composition of sets of resources, but without changing the software version of the NF Deployment.
* Changes both, of the software version of the NF Deployment and the virtualised resources and/or containerized workloads. In this case, the software is changed by means of performing tasks and management on the sets of virtualised resources and/or containerized workloads. For instance, a virtualised resource running a current software version can be terminated and a new virtualised resource be instantiated running a different software version.
* Changes related to the NF Deployment descriptor. This supports the correction of bugs in the descriptors, changes in naming schemes used in the descriptors, etc.

During the procedure of changing the current NF Deployment package, the DMS can add temporary virtualised resources and/or containerized workloads. And as indicated above, during such procedure adding and removal of virtualised resources and/or containerized workloads can also be performed as needed by the process.

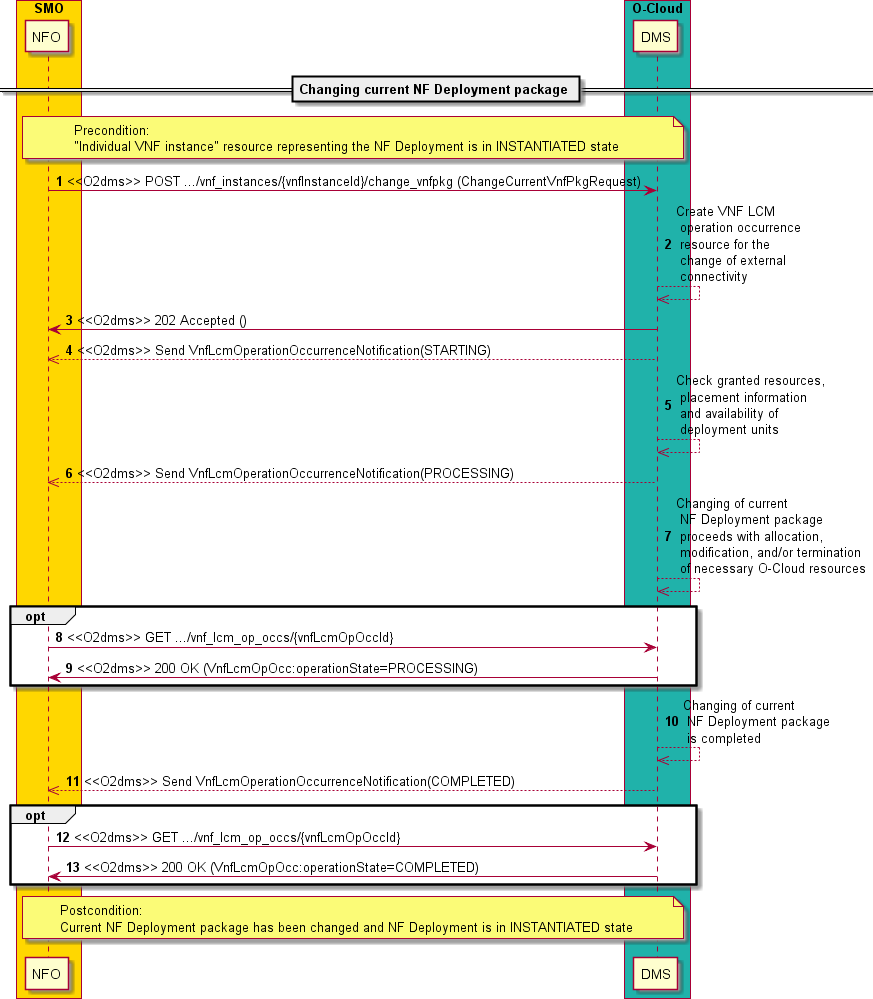
To perform the change of current NF Deployment package using the present O2dms API profile, the following sub- procedures are applicable:

* 1) changing the current NF Deployment package by triggering the operation "Change current VNF package" performed via the corresponding task resource "Change current VNF package task" on the "Individual VNF instance" resource following the flow illustrated in "Flow of VNF lifecycle management operations triggered by task resources" in clause 5.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub- 7procedure of change current VNF package is further profiled as illustrated in figure 2.4.2.7-1.

NOTE 3: While the profiled ETSI GS NFV-SOL 003 [[11]](#_bookmark13) changing current VNF package procedure contains steps related to VNF LCM granting exchanges between the VNFM and the NFVO of the NFV-MANO framework, the change current NF Deployment package procedure specified in the present document version does not contain these steps.

In addition, to monitor the status of the lifecycle operation in an asynchronous mode and be able to know the status of the operation (e.g., whether in progress or in some failed status), the API consumer shall request the creation of corresponding subscriptions to NF Deployment lifecycle management events following the flow "Flow of managing subscriptions" in clause 5.3.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of a lifecycle operation by query and reading information about the VNF LCM operation occurrence, which tracks the progress of the lifecycle operation.

Figure 2.4.2.7-1 illustrates the procedure flow for the changing current NF Deployment package including the sub- procedure introduced above.



##### Figure 2.4.2.7-1: Flow of changing current NF Deployment package

NOTE 4: Due to possible race conditions, the 202 response and the "STARTING" VnfLcmOperationOccurrenceNotification can arrive in any order at the SMO.

The procedure of changing current NF Deployment package consists of the following steps as illustrated in figure 2.4.2.7- 1:

NOTE 5: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** "Individual VNF instance" resource representing the NF Deployment is in INSTANTIATED state.

1. The SMO sends a POST request to the "Change current VNF package task" resource that represents the changing current NF Deployment package lifecycle operation to be executed on the NF Deployment and includes in the message content a data structure of type "ChangeCurrentVnfPkgRequest".
2. The DMS creates a new "Individual VNF LCM operation occurrence" resources for the request, which represents the lifecycle operation occurrence about the changing the current NF Deployment package.
3. The DMS returns a "202 Accepted" response with an empty message content and a "Location" HTTP header that points to the new "Individual VNF LCM operation occurrence" resource. See note 4.
4. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate the start of the lifecycle management operation occurrence with the "STARTING" state if the SMO is subscribed to such type of notifications. See note 4.
5. The DMS checks the granted resources, placement information and availability of deployment items for the change of current NF Deployment package. The DMS also checks the specific actions (e.g., allocate, modify, or terminate existing O-Cloud resources) to be performed for the change of current NF Deployment package based on the information contained in the descriptors of the NF Deployment and the current runtime information held about the NF Deployment.
6. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence enters the "PROCESSING" state if the SMO is subscribed to such type of notifications.
7. The DMS proceeds with the change of current NF Deployment package with the allocation, modification and/or termination of necessary O-Cloud resources.
8. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the ongoing lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
9. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
10. The DMS completes the change of current NF Deployment package.
11. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence is "COMPLETED” if the SMO is subscribed to such type of notifications.
12. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the completed lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
13. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".

**Postcondition:** The current NF Deployment package has been changed and the NF Deployment is in "INSTANTIATED" state.

**Error handling:** If the NF Deployment lifecycle operation fails, error information is provided in the notification messages that reports about the erroneous completion of the procedure. Error information is also available in the resource that represents the "Individual VNF lifecycle management operation occurrence" related to the NF Deployment lifecycle management operation.

#### Change NF Deployment flavour

An NF Deployment may have one or more NF Deployment flavours. An NF Deployment flavour is a deployment configuration that describes a specific deployment of an NF Deployment supporting specific KPIs, such as capacity and performance, or capabilities, such as specific NF Services.

NOTE 1: No reference to a specific use case from the O-RAN.WG6.ORCH-USE-CASES [[2].](#_bookmark4)

The functionality of changing the flavour of an NF Deployment is specified in the VNF lifecycle management as a dedicated operation named "Change VNF flavour".

During the procedure of changing the NF Deployment flavour, the DMS can add temporary virtualised resources and/or containerized workloads. And as indicated above, during such procedure adding and removal of virtualised resources and/or containerized workloads can also be performed as needed by the process and the requested new deployment configuration.

To perform the change NF Deployment flavour using the present O2dms API profile, the following sub-procedures are applicable:

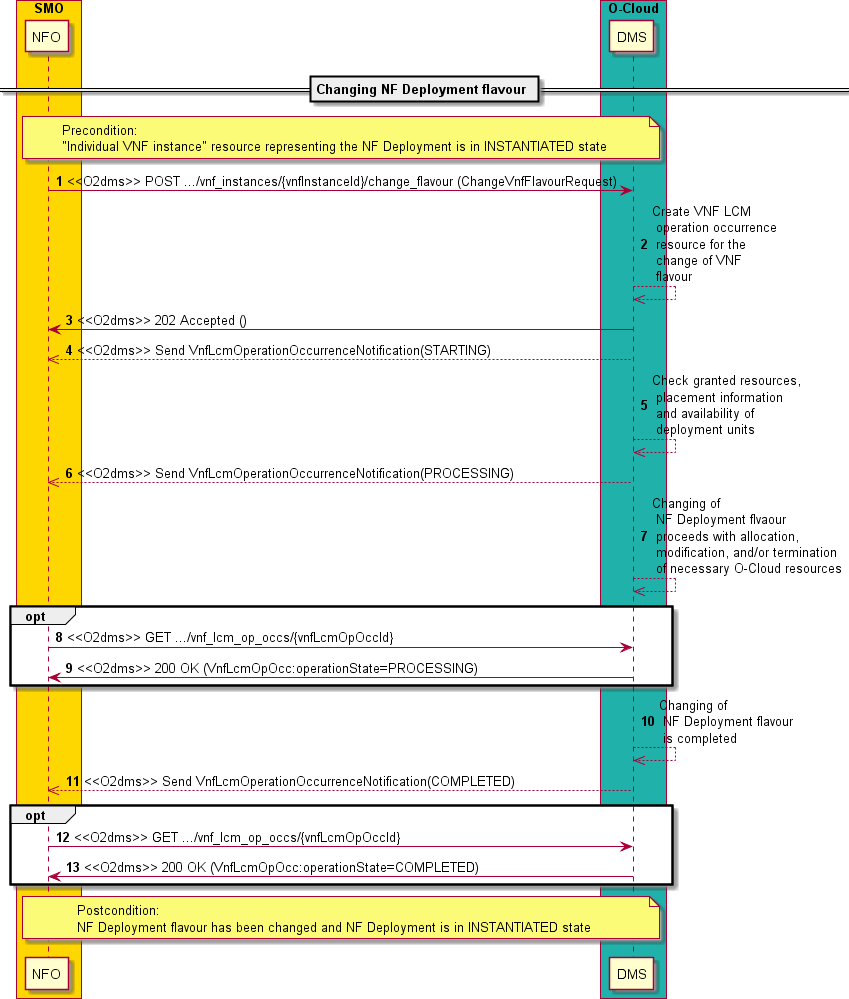
- 1) changing the NF Deployment flavour by triggering the operation "Change VNF flavour" performed via the corresponding task resource "Change VNF Flavour task" on the "Individual VNF instance" resource following the flow illustrated in "Flow of VNF lifecycle management operations triggered by task resources" in clause

5.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure of change VNF flavour is further profiled as illustrated in figure 2.4.2.8-1.

NOTE 2: While the profiled ETSI GS NFV-SOL 003 [[11]](#_bookmark13) changing VNF flavour procedure contains steps related to VNF LCM granting exchanges between the VNFM and the NFVO of the NFV-MANO framework, the change NF Deployment flavour procedure specified in the present document version does not contain these steps.

In addition, to monitor the status of the lifecycle operation in an asynchronous mode and be able to know the status of the operation (e.g., whether in progress or in some failed status), the API consumer shall request the creation of corresponding subscriptions to NF Deployment lifecycle management events following the flow "Flow of managing subscriptions" in clause 5.3.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of a lifecycle operation by query and reading information about the VNF LCM operation occurrence, which tracks the progress of the lifecycle operation.

Figure 2.4.2.8-1 illustrates the procedure flow for the changing current NF Deployment package including the sub- procedure introduced above.



##### Figure 2.4.2.8-1: Flow of changing current NF Deployment package

NOTE 3: Due to possible race conditions, the 202 response and the "STARTING" VnfLcmOperationOccurrenceNotification can arrive in any order at the SMO.

The procedure of changing NF Deployment flavour consists of the following steps as illustrated in figure 2.4.2.8-1: NOTE 4: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** "Individual VNF instance" resource representing the NF Deployment is in INSTANTIATED state.

1. The SMO sends a POST request to the "Change VNF flavour task" resource that represents the changing NF Deployment flavour lifecycle operation to be executed on the NF Deployment and includes in the message content a data structure of type "ChangeVnfFlavourRequest".
2. The DMS creates a new "Individual VNF LCM operation occurrence" resources for the request, which represents the lifecycle operation occurrence about the changing of NF Deployment flavour.
3. The DMS returns a "202 Accepted" response with an empty message content and a "Location" HTTP header that points to the new "Individual VNF LCM operation occurrence" resource. See note 3.
4. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate the start of the lifecycle management operation occurrence with the "STARTING" state if the SMO is subscribed to such type of notifications. See note 3.
5. The DMS checks the granted resources, placement information and availability of deployment items for the change of NF Deployment flavour. The DMS also checks the specific actions (e.g., allocate, modify, or terminate existing O-Cloud resources) to be performed for the change of NF Deployment flavour based on the information contained in the descriptors of the NF Deployment and the current runtime information held about the NF Deployment.
6. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence enters the "PROCESSING" state if the SMO is subscribed to such type of notifications.
7. The DMS proceeds with the change of NF Deployment flavour with the allocation, modification and/or termination of necessary O-Cloud resources.
8. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the ongoing lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
9. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".
10. The DMS completes the change of NF Deployment flavour.
11. The DMS sends to the SMO a VNF lifecycle management operation occurrence notification to indicate that the lifecycle management operation occurrence is "COMPLETED” if the SMO is subscribed to such type of notifications.
12. If desired, the SMO can poll the "Individual VNF LCM operation occurrence" resource to obtain information about the completed lifecycle operation by sending to the DMS a GET request to the "Individual VNF LCM operation occurrence" resource.
13. The DMS returns to the SMO the information about the operation by providing in the message content a data structure of type "VnfLcmOpOcc".

**Postcondition:** The NF Deployment flavour has been changed and the NF Deployment is in "INSTANTIATED" state.

**Error handling:** If the NF Deployment lifecycle operation fails, error information is provided in the notification messages that reports about the erroneous completion of the procedure. Error information is also available in the resource that represents the "Individual VNF lifecycle management operation occurrence" related to the NF Deployment lifecycle management operation.

### NF Deployment fault management

#### Get alarm list

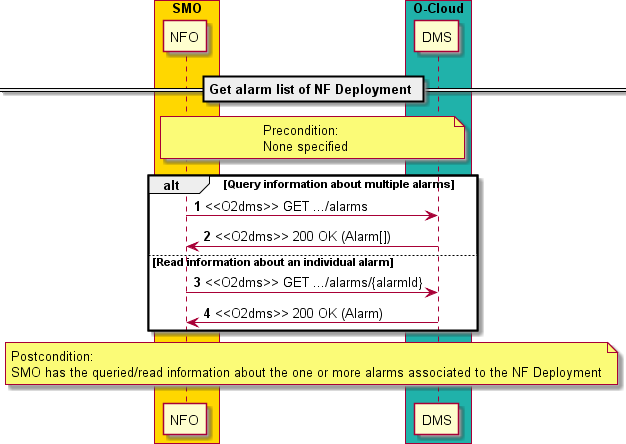
The functionality of getting the list of alarms of a VNF is covered in the ETSI NFV framework by the specific VNF fault management operation named "Get Alarm List". The operation comprises both sub-cases:

* Get information about multiple alarms, and
* Get information about a specific alarm.

To perform the get alarm list associated to an NF Deployment using the present O2dms API profile, the following sub- procedures are applicable:

* 1) Get information of multiple alarms by triggering the operation "Get Alarm List" on the "Alarms" resource following the alternative #1 in the flow illustrated in "Flow of the Get Alarm List operation" in clause 7.3.1 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure is further profiled as illustrated in figure 2.4.3.1-1.
* 2) Query/read information of a specific alarm by triggering the operation "Get Alarm List" on an "Individual alarm" resource, following the alternative #2 in the flow illustrated in "Flow of the Get Alarm List operation" in clause 7.3.1 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure is further profiled as illustrated in figure 2.4.3.1-1.

Figure 2.4.3.1-1 illustrates the procedure flow for get alarm list of an NF Deployment including the sub-procedure introduced above.



##### Figure 2.4.3.1-1: Flow of get alarm list of NF Deployment

The procedure of getting the alarm list of NF Deployment consists of the following steps as illustrated in figure 2.4.3.1- 1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** None specified.

1. If the SMO intends to query multiple alarms, it sends a GET request to the "Alarms" resource.
2. The DMS returns a "200 OK" response to the SMO and includes zero or more data structures of type "Alarm" in the message content, according to query request.
3. If the SMO intends to read the information of a particular alarm, it sends a GET request to the "Individual alarm" resource, by addressing to the appropriate alarm identifier in the resource URI.
4. The DMS returns a "200 OK" response and includes a data structure of type "Alarm" in the message content, according to the query request.

**Postcondition:** The SMO has the queried/read information about the one or more alarms associated to the NF Deployment.

**Error handling:** In case of failure, appropriate error information is provided in the response.

#### Notify

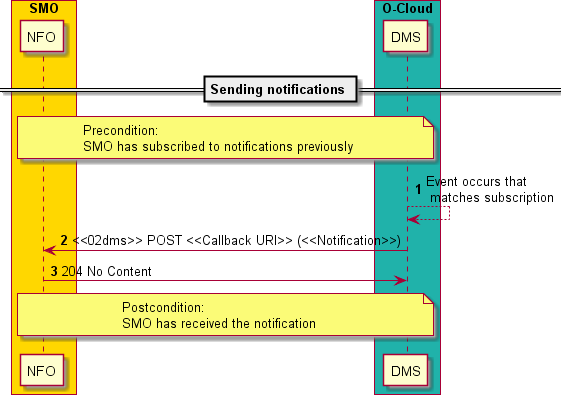
Raising and alarm and sending it to a consumer of fault management of a VNF is covered in the ETSI NFV framework by the specific VNF fault management operation named "Notify". Therefore, alarms are raised and sent by means of notifications. Other kinds of notifications can also be provided related to fault management procedures.

To perform the issuing of fault management notifications associated to an NF Deployment using the present O2dms API profile, the following procedure is applicable:

* 1) Send a notification following the flow illustrated in "Flow of sending notifications" in clause 7.3.4 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the procedure is further profiled as illustrated in figure 2.4.3.2-1.

In addition, to monitor the raised events in an asynchronous mode and be able to receive the corresponding notifications, the API consumer shall request the creation of corresponding subscriptions to NF Deployment fault management events following the flow of "Flow of managing subscriptions" in clause 7.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The API consumer may also monitor the status of active alarms by querying and reading information about them, as described in clause 2.4.3.1.

Figure 2.4.3.2-1 illustrates the procedure flow for sending alarm notification associated to an NF Deployment.



##### Figure 2.4.3.2-1: Flow of fault management notifications associated to an NF Deployment

The procedure of notifying an alarm associated to an NF Deployment consists of the following steps as illustrated in figure 2.4.3.2-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** The SMO has subscribed to notifications previously.

1. An event occurs in the DMS that matches a subscription for notifications.
2. The DMS generates and sends a notification that includes information about the event and sends it in the body of a POST request to the URI which the SMO has registered as part of the subscription request. The variable

<<Notification>> in the flow is a placeholder for the different types of notifications that can be sent via this service, including alarm notifications.

1. The SMO acknowledges the successful delivery of the notification by returning to the DMS a "204 No Content" response.

**Postcondition:** The SMO has received the notification.

**Error handling:** If the DMS does not receive the "204 No Content" response from the SMO, it can retry sending the same notification.

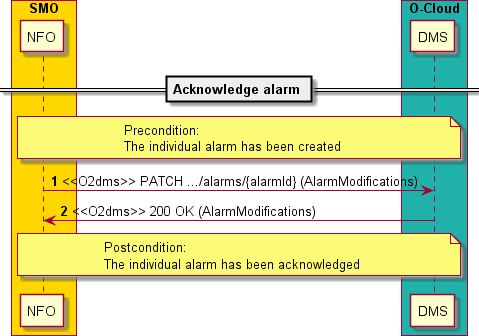
#### Acknowledge alarm

The functionality of acknowledging an individual alarm of a VNF is covered in the ETSI NFV framework by the specific VNF fault management operation named "Acknowledge Alarm".

To perform the acknowledgement of an alarm associated to an NF Deployment using the present O2dms API profile, the following procedure is applicable:

- 1) Acknowledge alarm by triggering the operation "Acknowledge Alarm " on an "Individual alarm" resource following the flow illustrated in "Flow of acknowledging alarm " in clause 7.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the procedure is further profiled as illustrated in figure 2.4.3.3-1.

Figure 2.4.3.3-1 illustrates the procedure flow for acknowledging an alarm of an NF Deployment including the procedure introduced above.



##### Figure 2.4.3.3-1: Flow of acknowledging alarm of NF Deployment

The procedure of acknowledging an alarm associated to an NF Deployment consists of the following steps as illustrated in figure 2.4.3.3-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** The individual alarm has been created.

1. The SMO sends a PATCH request to the individual alarm, by addressing to the appropriate alarm identifier in the resource URI with a message content including a data structure of "AlarmModifications" which includes the requested modification to acknowledge the alarm's acknowledgment status.
2. The DMS returns a "200 OK" response to the SMO and includes a data structure of type "AlarmModifications" in the message content.

**Postcondition:** The individual alarm has been acknowledged.

**Error handling:** In case of failure, appropriate error information is provided in the response.

### NF Deployment performance management

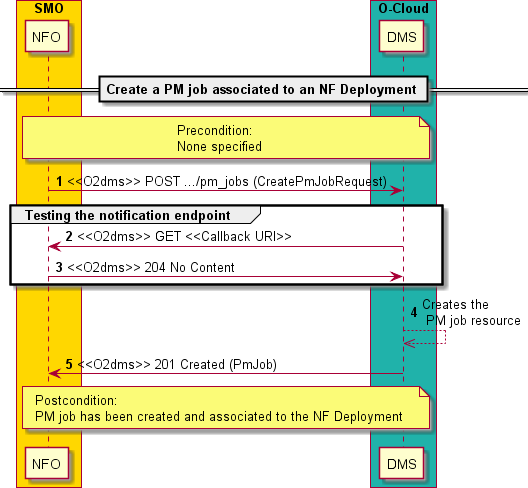
#### Create PM job

The functionality of creating a PM job to collect performance information associated to a VNF is covered in the ETSI NFV framework by the specific VNF performance management operation named "Create PM Job".

To create a PM job associated to an NF Deployment using the present O2dms API profile, the following procedure is applicable:

* 1) Create a PM Job by triggering the operation "Create PM Job" on the "PM Jobs" resource following the flow illustrated in "Flow of creating a PM job" in clause 6.3.1 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the procedure is further profiled as illustrated in figure 2.4.4.1-1.

Figure 2.4.4.1-1 illustrates the procedure flow for creating a PM job associated to an NF Deployment.



##### Figure 2.4.4.1-1: Flow of creating a PM job associated to an NF Deployment

The procedure of creating a PM job associated to an NF Deployment consists of the following steps as illustrated in figure 2.4.4.1-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** None specified.

1. If the SMO intends to create a PM job, it sends a POST request to the "PM Jobs" resource, including one data structure of type "CreatePmJobRequest" in the message content. The request includes information about the notification endpoint to be registered and to which notifications related to events about performance monitoring based on the PM job are expected to be sent.
2. To test the notification endpoint that has been registered, the DMS sends a GET request to the notification endpoint URI.
3. The SMO returns a "204 No Content" response to indicate success in testing the notification endpoint.
4. The DMS proceeds with the creation of the individual PM job resource and its instance (i.e., actual application instance or configuration that implements the PM job process).
5. The DMS returns a "201 Created" response to the SMO and includes in the message content a representation of the PM job that has been created.

**Postcondition:** The PM job has been created and associated to the NF Deployment.

**Error handling:** In case of failure, appropriate error information is provided in the response.

#### Query/read PM job information

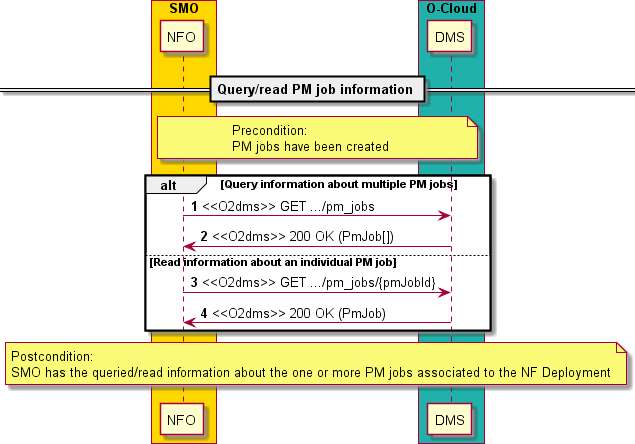
The functionality of querying/reading information of one or more PM jobs associated to a VNF is covered in the ETSI NFV framework by the specific VNF performance management operation named "Query PM Job". The operation comprises two sub-cases:

* Get information about multiple PM jobs, and
* Get information about a specific PM job.

To query information about one or more PM jobs associated to an NF Deployment using the present O2dms API profile, the following sub-procedures are applicable:

* 1) Get information of multiple PM jobs by triggering the operation "Get PM Job" on the "PM Jobs" resource following the alternative #1 in the flow illustrated in "Flow of querying/reading PM jobs" in clause 6.3.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure is further profiled as illustrated in figure 2.4.4.2-1.
* 2) Query/read information of a specific PM job by triggering the operation "Get PM Job" on an "Individual PM Job" resource, following the alternative #2 in the flow illustrated in "Flow of querying/reading PM jobs" in clause 6.3.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure is further profiled as illustrated in figure 2.4.4.2-1.

Figure 2.4.4.2-1 illustrates the procedure flow for querying/reading PM jobs associated to an NF Deployment, including the sub-procedures introduced above.



##### Figure 2.4.4.2-1: Flow of query/reading information of PM jobs associated to an NF Deployment

The procedure of query/reading information of PM jobs associated to an NF Deployment consists of the following steps as illustrated in figure 2.4.4.2-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** PM jobs have been created.

1. If the SMO intends to query multiple PM jobs, it sends a GET request to the "PM jobs" resource.
2. The DMS returns a "200 OK" response to the SMO and includes zero or more data structures of type "PmJob" in the message content, according to the query request.
3. If the SMO intends to read the information of a particular PM job, it sends a GET request to the "Individual PM job" resource, by addressing to the appropriate PM job identifier in the resource URI.
4. The DMS returns a "200 OK" response and includes a data structure of type "PmJob" in the message content, according to the query request.

**Postcondition:** The SMO has the queried/read information about the one or more PM jobs associated to the NF Deployment.

**Error handling:** In case of failure, appropriate error information is provided in the response.

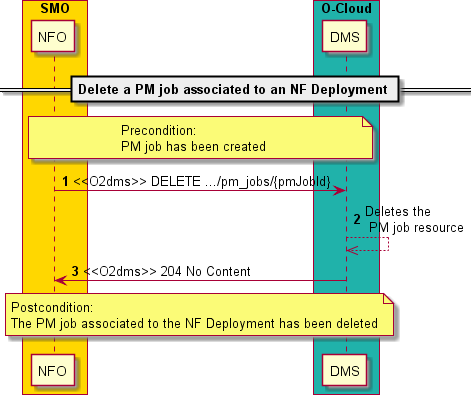
#### Delete a PM job

The functionality of deleting a PM job associated to a VNF is covered in the ETSI NFV framework by the specific VNF performance management operation named "Delete PM Job".

To delete a PM job associated to an NF Deployment using the present O2dms API profile, the following procedure is applicable:

* 1) Delete a PM job by triggering the operation "Delete PM Job" on the specific "Individual PM Job" resource following the flow illustrated in "Flow of deleting a PM job" in clause 6.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the procedure is further profiled as illustrated in figure 2.4.4.3-1.

Figure 2.4.4.3-1 illustrates the procedure flow for deleting a PM job associated to an NF Deployment.



##### Figure 2.4.4.3-1: Flow of deleting a PM job associated to an NF Deployment

The procedure of deleting a PM job associated to an NF Deployment consists of the following steps as illustrated in figure 2.4.4.3-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** The PM job has been created.

1. If the SMO intends to delete a PM job, it sends a DELETE request to the "Individual PM job" resource, addressed by the appropriate PM job identifier in its resource URI.
2. The DMS deletes the "Individual PM job" resource and the PM job instance.
3. The DMS returns to the SMO a "204 No Content" response, with an empty message content.

**Postcondition:** The PM job associated to the NF Deployment has been deleted.

**Error handling:** In case of failure, appropriate error information is provided in the response.

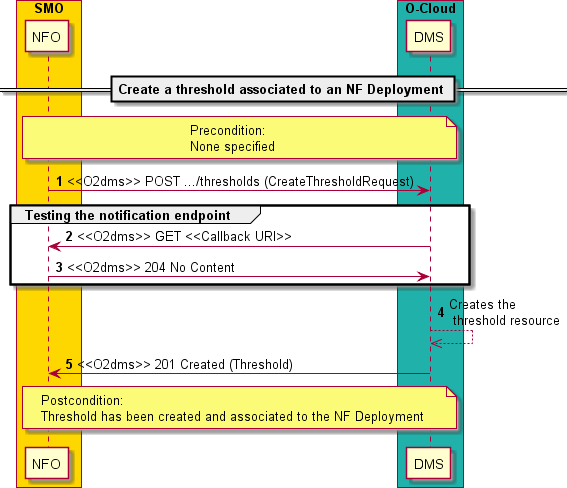
#### Create a threshold

The functionality of creating a threshold to specify threshold levels on specified performance metrics associated to associated to a VNF is covered in the ETSI NFV framework by the specific VNF performance management operation named "Create Threshold ".

To create a threshold associated to an NF Deployment using the present O2dms API profile, the following procedure is applicable:

* 1) Create a threshold by triggering the operation "Create Threshold" on the "Thresholds " resource following the flow illustrated in "Flow of creating a threshold" in clause 6.3.5 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the procedure is further profiled as illustrated in figure 2.4.4.4-1.

Figure 2.4.4.4-1 illustrates the procedure flow for creating a threshold associated to an NF Deployment.



##### Figure 2.4.4.4-1: Flow of creating a threshold associated to an NF Deployment

The procedure of creating a threshold associated to an NF Deployment consists of the following steps as illustrated in figure 2.4.4.4-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** None specified.

1. If the SMO intends to create a threshold, it sends a POST request to the "Thresholds" resource, including one data structure of type "CreateThresholdRequest" in the message content. The request includes information about the notification endpoint to be registered and to which notifications related to events about performance monitoring based on the threshold are expected to be sent.
2. To test the notification endpoint that has been registered, the DMS sends a GET request to the notification endpoint URI.
3. The SMO returns a "204 No Content" response to indicate success in testing the notification endpoint.
4. The DMS proceeds with the creation of the individual threshold resource and its instance (i.e., actual application instance or configuration that implements the threshold process).
5. The DMS returns a "201 Created" response to the SMO and includes in the message content a representation of the threshold that has been created.

**Postcondition:** The threshold has been created and associated to the NF Deployment.

**Error handling:** In case of failure, appropriate error information is provided in the response.

#### Query/read threshold information

The functionality of querying/reading information of one or more thresholds associated to a VNF is covered in the ETSI NFV framework by the specific VNF performance management operation named "Query Threshold". The operation comprises two sub-cases:

* Get information about multiple thresholds, and
* Get information about a specific threshold.

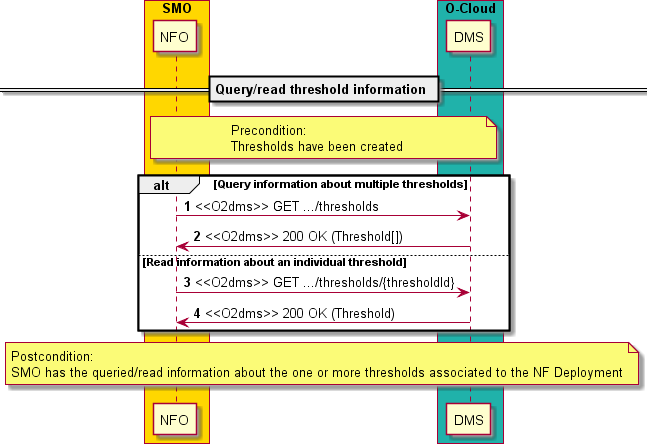
To query information about one or more thresholds associated to an NF Deployment using the present O2dms API profile, the following sub-procedures are applicable:

* 1) Get information of multiple thresholds by triggering the operation "Get Threshold" on the "Thresholds" resource following the alternative #1 in the flow illustrated in "Flow of querying/reading thresholds" in clause

6.3.6 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure is further profiled as illustrated in figure 2.4.4.5-1.

* 2) Query/read information of a specific threshold by triggering the operation "Get Threshold" on an "Individual threshold" resource, following the alternative #2 in the flow illustrated in "Flow of querying/reading thresholds" in clause 6.3.6 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the sub-procedure is further profiled as illustrated in figure 2.4.4.5-1.

Figure 2.4.4.5-1 illustrates the procedure flow for querying/reading thresholds associated to an NF Deployment, including the sub-procedures introduced above.



##### Figure 2.4.4.5-1: Flow of query/reading information of thresholds associated to an NF Deployment

The procedure of query/reading information of thresholds associated to an NF Deployment consists of the following steps as illustrated in figure 2.4.4.5-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** Thresholds have been created.

1. If the SMO intends to query multiple thresholds, it sends a GET request to the "Thresholds" resource.
2. The DMS returns a "200 OK" response to the SMO and includes zero or more data structures of type "Threshold" in the message content, according to the query request.
3. If the SMO intends to read the information of a particular threshold, it sends a GET request to the "Individual threshold" resource, by addressing to the appropriate threshold identifier in the resource URI.
4. The DMS returns a "200 OK" response and includes a data structure of type "Threshold" in the message content, according to the query request.

**Postcondition:** The SMO has the queried/read information about the one or more thresholds associated to the NF Deployment.

**Error handling:** In case of failure, appropriate error information is provided in the response.

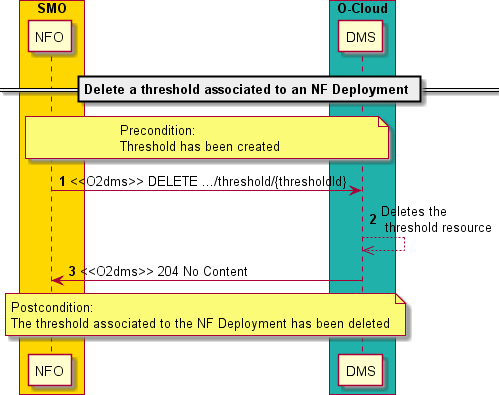
#### Delete a threshold

The functionality of deleting a threshold associated to a VNF is covered in the ETSI NFV framework by the specific VNF performance management operation named "Delete Threshold".

To delete a threshold associated to an NF Deployment using the present O2dms API profile, the following procedure is applicable:

* 1) Delete a threshold by triggering the operation "Delete Threshold" on the specific "Individual threshold" resource following the flow illustrated in "Flow of deleting thresholds" in clause 6.3.7 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the procedure is further profiled as illustrated in figure 2.4.4.6-1.

Figure 2.4.4.6-1 illustrates the procedure flow for deleting a threshold associated to an NF Deployment.



##### Figure 2.4.4.6-1: Flow of deleting a threshold associated to an NF Deployment

The procedure of deleting a threshold associated to an NF Deployment consists of the following steps as illustrated in figure 2.4.4.3-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** The threshold has been created.

1. If the SMO intends to delete a threshold, it sends a DELETE request to the "Individual threshold" resource, addressed by the appropriate threshold identifier in its resource URI.
2. The DMS deletes the "Individual threshold" resource and the threshold instance.
3. The DMS returns to the SMO a "204 No Content" response, with an empty message content.

**Postcondition:** The threshold associated to the NF Deployment has been deleted.

**Error handling:** In case of failure, appropriate error information is provided in the response.

#### Sending performance related notifications

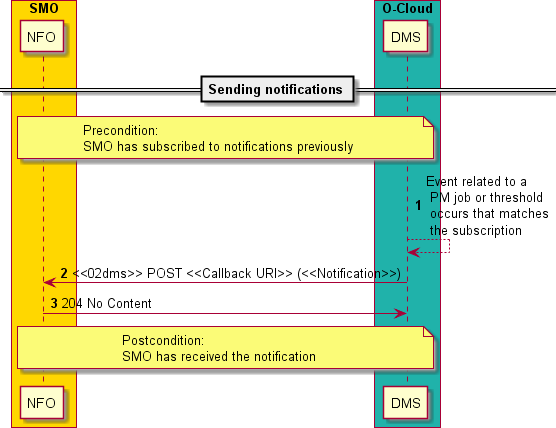
Sending notifications related to performance management of a VNF is covered in the ETSI NFV framework by the specific VNF fault management operation named "Notify". Therefore, performance management events, such as the availability of a performance report or a crossing of a threshold, are raised and sent by means of notifications.

To perform the issuing of performance management notifications associated to an NF Deployment using the present O2dms API profile, the following procedure is applicable:

* 1) Send a notification following the flow illustrated in "Flow of sending notifications" in clause 6.3.9 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the procedure is further profiled as illustrated in figure 2.4.4.7-1.

In addition, to monitor the raised events in an asynchronous mode and be able to receive the corresponding notifications, the API consumer shall request the creation of corresponding subscriptions to NF Deployment performance management events. This is performed by providing in the procedures of creating PM jobs and thresholds the notification endpoint URI in the corresponding request messages.

Figure 2.4.4.7-1 illustrates the procedure flow for sending performance related notification associated to an NF Deployment.



##### Figure 2.4.4.7-1: Flow of performance management notifications associated to an NF Deployment

The procedure of notifying about performance management events associated to an NF Deployment consists of the following steps as illustrated in figure 2.4.4.7-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** The SMO has subscribed to notifications previously.

1. An event related to a PM job or threshold occurs in the DMS that matches a subscription for notifications.
2. The DMS generates and sends a notification that includes information about the event and sends it in the body of a POST request to the URI which the SMO has registered as part of the subscription request. The variable

<<Notification>> in the flow is a placeholder for the different types of notifications that can be send via this service, including performance report availability and threshold crossing notifications.

1. The SMO acknowledges the successful delivery of the notification by returning to the DMS a "204 No Content" response.

**Postcondition:** The SMO has received the notification.

**Error handling:** If the DMS does not receive the "204 No Content" response from the SMO, it can retry sending the same notification.

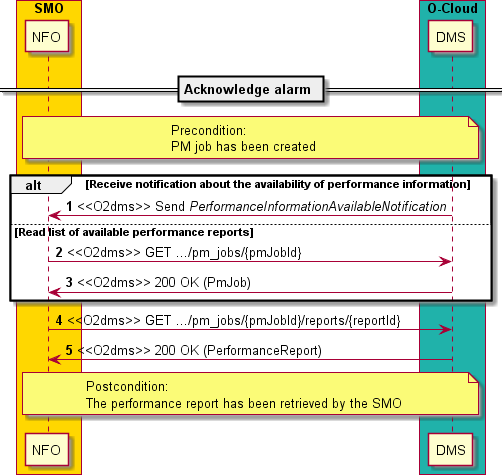
#### Obtaining performance reports

The functionality of obtaining performance reports of a VNF is not covered in the ETSI NFV framework by a specific VNF fault management operation, but rather by the joint functionality provided by the notification and query report procedures.

To obtain the performance reports generated from PM jobs associated to an NF Deployment using the present O2dms API profile, the following sub-procedures are applicable:

* 1) Receive a notification about the availability of performance information" and then query the "Individual performance report" resource following the alternative #1 in the flow illustrated in "Flow of obtaining performance reports" in clause 6.3.4 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the procedure is further profiled as illustrated in figure 2.4.4.8-1.
* 2) Query/read list of available performance reports generated from a PM job, and then query the "Individual performance report" resource following the alternative #2 in the flow illustrated in "Flow of obtaining performance reports" in clause 6.3.4 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) For the case of an NF Deployment, the procedure is further profiled as illustrated in figure 2.4.4.8-1.

Figure 2.4.4.8-1 illustrates the procedure flow for obtaining performance reports of an NF Deployment including the procedure introduced above.



##### Figure 2.4.4.8-1: Flow of obtaining performance reports of NF Deployment

The procedure of obtaining performance reports generated from a PM job associated to the NF Deployment consists of the following steps as illustrated in figure 2.4.4.8-1:

NOTE: The procedure does not illustrate error cases, only successful procedure steps.

**Precondition:** The PM job has been created.

1. The DMS send to the SMO a PerformanceInformationAvailableNotification (see clause 2.4.4.7). The notification indicates the availability of a new performance report. The notification includes a link from which the report can be obtained.
2. Alternatively, the SMO can get the list of performance reports generated from a PM job by sending a GET request to the "Individual PM job" resource. The representation of the "Individual PM job" can include information about performance reports that are available for this PM job, including their links.
3. In case of the SMO requesting the information of the PM job, the DMS returns a "200 OK" response, including a data structure of type "PmJob" in the message content.
4. Based on the information received either from step 1) or step 3), the SMO sends to the DMS a GET request to the URI obtained in those steps. The GET request addresses, by using corresponding identifiers, the particular "Individual PM job" resource and "Individual performance report" of interest.
5. The DMS returns a "200 OK" response to the SMO and includes a data structure of type "PerformanceReport" in the message content.

**Postcondition:** The performance report has been retrieved by the SMO.

**Error handling:** In case of failure, appropriate error information is provided in the response.

## Orchestration aspects with O2dms ETSI NFV profile

### Resource groups and tenancy

For the deployment and lifecycle of NF Deployments either or both, virtualised resources and containerized workloads are necessary. Virtualised resources and/or containerized workloads are allocated from O-Cloud Node Clusters built from O-Cloud resource pools. The O-Cloud resource pools can also be shared or common for the deployment of multiple O-RAN NF. This is typically realized by enabling features such as resource groups and multi-tenancy.

In the ETSI NFV-profiled O2dms interface, resource grouping and multi-tenancy is achieved by conveying over the O2 interface information that identifies the resource groups and tenant users to which the NF Deployments relate to in the form of "VIM connection info". The data type "VimConnectionInfo", defined in clause 3.2.6.3.2, serves such purposes.

The "VimConnectionInfo" contains attributes that support determining the type and specific function to use for resource infrastructure management of the O-Cloud resources assigned to the O-Cloud Node Cluster. Types of infrastructure management solutions that can be used are registered in the ETSI NFV registry of VIM-related information [[19].](#_bookmark21) The structure of the registry is defined in Annex C of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

The ETSI NFV registry of VIM-related information [[19]](#_bookmark21) contains entries to support VM and containerized workloads management such as:

* Registry entries related to VM-based resource management: a) registration of ETSINFV.OPENSTACK\_KEYSTONE.V\_2, i.e., OpenStack® with Keystone V2, and b) registration of ETSINFV.OPENSTACK\_KEYSTONE.V\_3, i.e., OpenStack® with Keystone V3, and
* Registry entries related to containerized workloads management: a) registration of ETSINFV.KUBERNETES.V\_1, i.e., Kubernetes®, b) registration of ETSINFV.DOCKER.V\_2, i.e., Docker® registry as CIR, and c) registration of ETSINFV.HELM.V\_3, i.e., Helm® as MCIOP Repository.

NOTE 1: Void.

NOTE 2: The OpenStack® Word Mark and OpenStack Logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation’s permission.

NOTE 3: The Kubernetes® Word Mark and Kubernetes Logo are either registered trademarks/service marks or trademarks/service marks of The Linux Foundation, in the United States and other countries and are used with the Linux Foundation’s permission.

NOTE 4: The Helm® Word Mark and Helm Logo are either registered trademarks/service marks or trademarks/service marks of The Linux Foundation, in the United States and other countries and are used with the Linux Foundation’s permission.

NOTE 5: The Docker® and the Docker logo are trademarks or registered trademarks of Docker, Inc. in the United States and/or other countries.

Information about "VIM connection info" shall be known to the SMO as pre-requisite to perform NF Deployment lifecycle.

The resources that are granted for consumption from the resource pools can be further grouped by "infrastructure resource groups", which can be mapped to logical groupings of virtualised resources assigned to a tenant (or user) within the O-Cloud. This is represented by the "resourceGroupId" attribute in the "ResourcePlacement" defined in clause 3.2.6.5.3.

In the case of containerized workloads, the logical grouping of a particular set of identifiers, resources, policies and authorizations is realized by means of "namespaces". In addition, namespace quotas define the upper limit on specific types of resources that can be used to deploy containerized workloads within the scope of a namespace. This is represented by the "containerNamespace" attribute in the "ResourcePlacement" defined in clause 3.2.6.5.3.

### Placement information and constraints

Placement information and constraints for NF Deployments are conveyed in the O2dms ETSI NFV profile out-of-band from the O2dms\_LifecycleDeployment API. ETSI GS NFV-SOL 003 [[11]](#_bookmark13) specifies data types which carry relevant information about placement and constraints, and such data types are either exchanged via the granting procedures or provided as placement requirements information in the descriptors.

NOTE 1: In the present document version, granting exchanges are not specified. ETSI GS NFV-SOL 003 [[11]](#_bookmark13) defines the "VNF LCM granting interface" as the mechanism to convey placement and constraints information "in-band" from an API perspective.

NOTE 2: Various forms for conveying placement and constraints information are possible, including: a) using some form of additional grant-less (no grant exchanges) API interactions, b) using declarative descriptors or configuration files, c) using granting exchanges conforming to those defined by ETSI GS NFV-SOL 003 [[11],](#_bookmark13) or d) a combination thereof. However, the present document version does not specify support for any of these specifically and it is left for specific deployment or implementation.

The present clause describes the relevant information as defined in the "ResourcePlacement" type in clause 3.2.6.5.3 and "ResourcePlacementConstraint" type in clause 3.2.6.5.4. The information that can be conveyed using these types shall be made available to the DMS as a pre-condition for the instantiation of an NF Deployment by means not specified by the present document (see also note 2).

Relevant attributes and types are:

* "vimConnectionId": since there can be multiple connections to underlaying infrastructure resource management functions, the scope of management of an entity implementing such a function also determines a level of placement control. For instance, infrastructure resource management entities can be associated to manage resources of specific O-Cloud Node Clusters. Therefore, by indicating the VIM connection to use, a selection of which O-Cloud Node Cluster can be determined for a given NF Deployment to manage.
* "zoneId": resource zones provide a physical segregation of O-Cloud resources within an O-Cloud Site, and by identifying them, it can be controlled to which zone a resource is expected to be deployed.
* "ResourcePlacementConstraint": the type represents information about the placement constraints for two or more resources or containerized workloads of an NF Deployment in terms of affinity and anti-affinity and for diverse scope levels, such as O-Cloud Site, resource zones within an O-Cloud Site, and O-Cloud Nodes. The information about affinity and anti-affinity is typically provided by the deployment descriptors of the NF Deployment.

NOTE 3: In the case of the ETSI GS NFV-SOL 003 [[11],](#_bookmark13) such a descriptor is referred as VNFD. Refer also to the description in table 2.3.2-1 of the present document.

Containerized workloads can also have specific deployment constraints. Examples of deployment constraints are the following: defining specific O-Cloud Nodes to be used for the deployment or capabilities of the O-Cloud Nodes such as support for acceleration. The "mcioConstraints" attribute in the "ResourcePlacement" type defined in clause 3.2.6.5.3 can be used to convey these specific deployment constraints.

### Network connectivity for NF Deployments

Network connectivity information of NF Deployments in the O2dms ETSI NFV profile is exposed and/or conveyed via specific data types. Furthermore, in terms of network connectivity the profile modelling is based on the following key concepts: virtual links and connection points.

A virtual link is an abstraction of a network, which is used by the management and orchestration functions in the ETSI NFV architectural framework. Once applied to the O-RAN framework, virtual links also become an abstraction of connectivity which is exposed and used over the O2dms interface. Virtual links can be realized in various forms such as virtual/overlay networks in an O-Cloud site, or a composition of various networks. Virtual links can be internal or external to an NF Deployment:

* Internal virtual link: it expresses the connectivity between the components or constituents of an NF Deployment.
* External virtual link: it expresses the connectivity external to the NF Deployment, i.e., networks that are used to interconnect among NF Deployments, or between NF Deployments and other entities such as to other O- RAN NF, xApps, or even other systems such as management and orchestration ones.

Internal virtual links can be modelled in two forms depending on the entity that manages the resources realizing the virtual link:

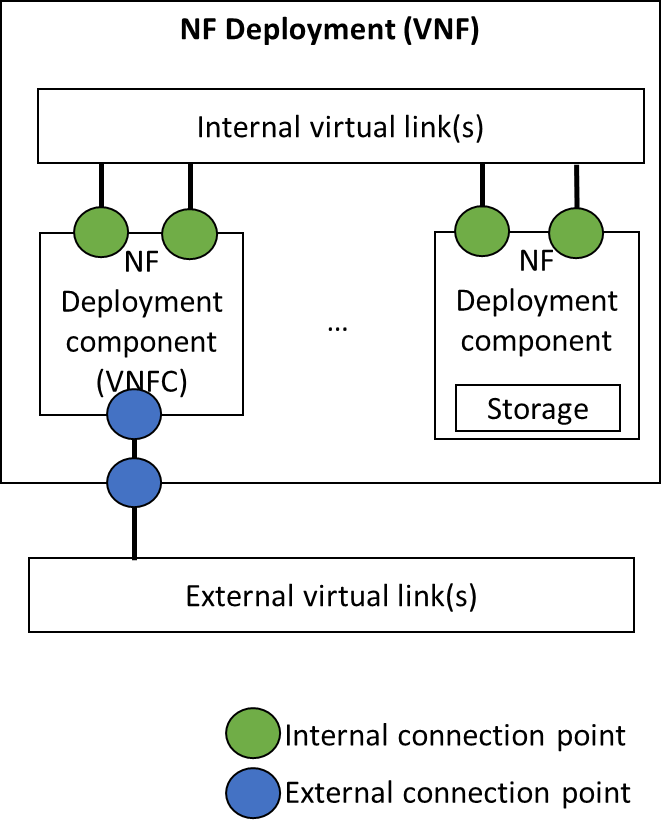
* NF Deployment virtual link: it denotes an internal NF Deployment virtual link that is managed by the DMS. The data type providing information about such kind of virtual link is named "VnfVirtualLinkResourceInfo" (see clause 3.2.6.3.9).
* Externally-managed virtual link: it denotes an internal NF Deployment virtual link that is managed externally to the DMS, e.g., in the present profiling case the SMO. The data type providing information about such kind of virtual link is named "ExtManagedVirtualLinkInfo" (see clause 3.2.6.3.7).

NOTE 1: The reason for differentiating the two types comes from the fact that there are deployment cases where the network resources are managed by organizations or systems different to those managing the NF Deployments.

A connection point is also an abstraction that denotes the port/interface of connectivity of an NF Deployment or a constituent within an NF Deployment to a network. Similarly, as with the virtual link case, connections points are further qualified as internal or external connections points of an NF Deployment.

* Internal connections points: these are the connection points used to connect to the internal virtual links of an NF Deployment. The structured "vnfcCpInfo" attribute in the "VnfcResourceInfo" provides information about such kind of connection point (see clause 3.2.6.3.8).
* External connection points: these are the connection points used to connect the NF Deployment to the external virtual links. The data type providing information about such kind of connection point is named "VnfExtCpInfo" (see clause 3.2.6.3.4).

Figure 2.5.3-1 illustrates the composition of an NF Deployment and the relationship with the networking concepts.



##### Figure 2.5.3-1: NF Deployment and networking concepts

External connection points can be associated to various types of other connection points. For instance, an external connection point can be realized by an actual internal connection point of an NF Deployment, i.e., the internal connection point is re-exposed externally and used for connecting to the external virtual link, or it can also be realized by associating the external connection point directly with an internal virtual link. The former case is the one illustrated in figure 2.5.3-1. The type of the connection point and how it relates to the other constituents of an NF Deployment is denoted by a series of specific attributes in the "VnfExtCpInfo".

In addition, other specific connection points are also defined depending on how they are further realized or mapped to.

* Virtual IP connection point: it denotes a connection point which is basically a Virtual IP (VIP).
* Virtual connection point: it denotes a connection point which does not have a specific virtual resource representation in the infrastructure; instead, it is typically realized by some form of configuration into one or multiple resources (e.g., external load balancers).

EXAMPLE 1: Kubernetes® Service is an example of a virtual connection point.

Depending on the type of connection point, additional elements are modelled, such as the virtual link ports and network attachment definition resources to specify the points of attachment to the virtual links.

* A virtual link port defines the point of attachment of a virtual network interface of a VM-based component of an NF Deployment to a virtual network.

EXAMPLE 2: An OpenStack® Layer 2 Networking Port is an example of virtual link port.

* A network attachment definition resource defines the point of attachment of a virtual network interface of an OS-container component of an NF Deployment to a secondary container cluster network.

EXAMPLE 3: A Network Attachment Definition (NAD) of Multus CNI™ is an example of a network attachment definition resource.

NOTE 2: Void.

NOTE 3: The CNI™ Word Mark is a registered trademarks/service marks of The Linux Foundation, in the United States and other countries and are used with the Linux Foundation’s permission.

The virtual connection points and network attachment definition resources are features either specified or referenced by the Release 4 version of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) and primarily concern to OS container-based deployments. There is no need to define connections points for the OS-container components of an NF Deployment that are connected to the primary container cluster network, because the OS-container components are by default connected to such a network when deployed.

As defined in the present document, the API consumer (i.e., SMO) can convey various sets of network protocol information to the DMS to realize various network connectivity patterns. Two cases of connection point provisioning can be realized as supported by the attributes in the "VnfExtCpConfig" data type specified in clause 3.2.6.3.24:

* The API consumer provisions with the infrastructure management functions in the O-Cloud the connection points in advance and provides subsequently to the DMS references to the virtual link ports or network attachment definition resources that have been provisioned. This case is applicable for both virtual link port- based and network attachment definition resource connectivity.
* The API consumer provides specific protocol layer information to the DMS, and the DMS subsequently uses such information to provision the virtual link ports or virtual connection points, depending on the type of connection point. This is only applicable for the case of virtual link port-based connectivity and virtual connection points.

Regarding the protocol layer information, the "CpProtocolData" and "CpProtocolInfo" data types defined in clauses

3.2.6.3.25 and 3.2.6.3.13, respectively, represent information about the protocol layer(s) that a connection point uses and the associated protocol information. According to the profiled version of ETSI GS NFV-SOL 003 [[11],](#_bookmark13) two types of layer protocol types are supported:

* "IP\_OVER\_ETHERNET": used to provide network address data for IP over Ethernet.
* "IP\_FOR\_VIRTUAL\_CP": used to provide network address data for an external connection point realized as a virtual connection point.

For the IP over Ethernet case, additional data types provide the support to define MAC addresses, IP addresses, and types of segmentation and encapsulation, including VLAN ID support. For the IP for virtual connection point, the data types enable the support to assign concrete fixed IP addresses corresponding to the external load balancers associated to the virtual connection point.

### Software images information

As described in clause 3.5.1 of O-RAN.WG6.O2-GA&P [[4],](#_bookmark6) an O-RAN Cloudified Network Function Software Package is onboarded to further perform software and lifecycle management of O-RAN NF. The NF software package contains software images for the compute resource components of the NF. In the ETSI NFV-based profile, the NF software packaging concept is fulfilled by the "VNF package" artifact.

To instantiate NF Deployments corresponding compute resource components of the NF Deployment have associated software images:

* in the case of a VM-based NF Deployment: VM software images, and
* in the case of a container-based NF Deployment, OS container software images.

NOTE 1: O-Cloud assets are assumed to be conveyed out-of-band because in the present document version, granting exchanges are not specified. ETSI GS NFV-SOL 003 [[11]](#_bookmark13) defines the "VNF LCM granting interface" as the mechanism to convey such information "in-band".

Regarding the NF Deployment lifecycle management, the onboarding of the software images to the O-Cloud and the transfer of related software image information of the NF Deployment shall be conveyed out-of-band of the O2dms\_LifecycleDeployment service API prior to requesting the instantiation of the NF Deployment. The provision of such information is realized in the form of "O-Cloud assets", which are specified in clause 3.2.6.5.5 using the data type "CloudAssets". In the context of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) these are referred as "VIM assets".

VIM assets contain "software image assets" that define the mapping between software images referred in the VNFD (NF Deployment descriptor) and software images managed in the infrastructure resource managers such as the VIM (for VM-based software images) and dedicated repositories such as the CIR (for container-based software images).

NOTE 2: Void.

OS container software images are features either specified or referenced by the Release 4 version of ETSI GS NFV- SOL 003 [[11]](#_bookmark13) and concern to OS container-based deployments.

### Resource management for NF Deployments

NF Deployments are realized by one or multiple instances and types of resources, such as compute, storage and network. In the O2dms ETSI NFV profile, resource handles play a key role in keeping the traceability, inventory and accountability of the resources that are used by an NF Deployment. Resource handles also enable the DMS to manage such resources via the other DMS resource infrastructure functions. The data type providing information about a virtualised resource instance in the infrastructure is named "ResourceHandle" (see clause 3.2.6.3.20). All the constituents of an NF Deployment that have a resource representation in the O-Cloud have an associated resource handle.

A resource handle compiles various types of information, but primarily:

* An identification of the VIM or CISM connection (interface information and credentials) used to manage the resource.
* An identification of the resource within the scope of the infrastructure resource provisioning functions.
* Information about the type of resource within the scope of the infrastructure resource provisioning functions.
* Additional information about the resource which is specific to the resource and its type and that is made available by the underlying infrastructure resource management functions of the DMS (in the ETSI NFV terminology these are referred as VIM and CISM).
* Namespace of the resource, in case of a containerized workload resource.

The types of resources composing the NF Deployment differ depending on the deployment form:

* For VM-based deployments: compute, storage and network resources are realized as "virtual compute" (or VirtualCompute), "virtual storage" (or VirtualStorage) and "virtual network” (or VirtualNetwork).
* For OS container-based deployments: compute, storage and network resources are modeled as "Compute MCIO", "Storage MCIO" and "Network MCIO". MCIO stands for "Managed Container Infrastructure Object" and it models the objects that are managed and exposed by the container infrastructure services management and represents the desired and actual state of a containerized workload.

EXAMPLE: In case of a Kubernetes® based implementation of infrastructure resources management functions for OS container-based deployments, examples of "Compute MCIO" are Deployment, Pod, StatefulSet; examples of "Storage MCIO" are: PersistentVolumeClaim; and examples of "Network MCIO" are Ingress, Service. For a full list of mapped Kubernetes® resource objects mapped to MCIOs, refer to ETSI GS NFV-SOL 018 [[18].](#_bookmark20)

Resources for NF Deployments are also defined based on the content in the deployment descriptors. In the case of the O2dms ETSI NFV profile, it is assumed that the NF Deployment has a corresponding deployment descriptor that provides requirements and describes the characteristics of the resources necessary for the deployment and lifecycle of the NF Deployment.

NOTE 1: In the case of the ETSI GS NFV-SOL 003 [[11],](#_bookmark13) such a descriptor is referred as VNFD. Refer also to the description in table 2.3.2-1 of the present document.

Key characteristics about resources that are of use by the profile are:

* Deployment descriptors are identified. In the case of VNFD, the attribute is typically named as "vnfdId".
* Different types of resources can compose an NF Deployment (see also above). As per the present document version, the following are defined based on the classes defined in the "ResourceDefinition" data type (see clause 3.2.6.5.2).
  + COMPUTE: represents a VM-based compute resource.
  + VL: represents a virtual link, which in turns represents a network resource, e.g., virtual network.
  + STORAGE: represents a storage resource.
  + LINKPORT: represents a link port, i.e., a port of attachment to a network.
  + OSCONTAINER: represents an OS container-based compute resource.
  + VIRTUALCP: represents a virtual connection point.
* Resources have associated resource templates (declarative descriptors). In the case of a VM-based compute resource, the template is further identified by the attribute named "vduId".
* Allocated resources are identifiable using a corresponding "ResourceHandle". NOTE 2: Void.

In the case of OS container-based deployments, additional types defined by the ETSI GS NFV-SOL 003 [[11]](#_bookmark13) are also considered:

* "McioInfo": it provides information on the MCIO(s) representing the constituents of the NF Deployment that are realized by one or a set of OS containers and that are created using the same declarative descriptor.

As introduced in clause 2.5.1, the "VimConnectionInfo" is used to indicate the type of resource infrastructure management to use as well as the necessary information to enable the consumption of such management.

### Acceleration abstraction layer

#### Overview

The O-RAN.WG6.AAL-GAnP.0 [[31]](#_bookmark32) introduces the concepts of acceleration abstraction layer (AAL), whose purpose is to specify a common and consistent interface for hardware accelerators to the applications in order to facilitate the decoupling of an application, e.g., an O-RAN NF, from a specific hardware implementation.

In the context of the O2dms ETSI NFV profile and related orchestration procedures, there are three aspects to be considered:

* The description of acceleration requirements for an NF Deployment,
* Information exchange between the DMS consumer and the DMS about requested acceleration capabilities for an NF Deployment, and
* Injecting acceleration-related configuration into the NF Deployment.

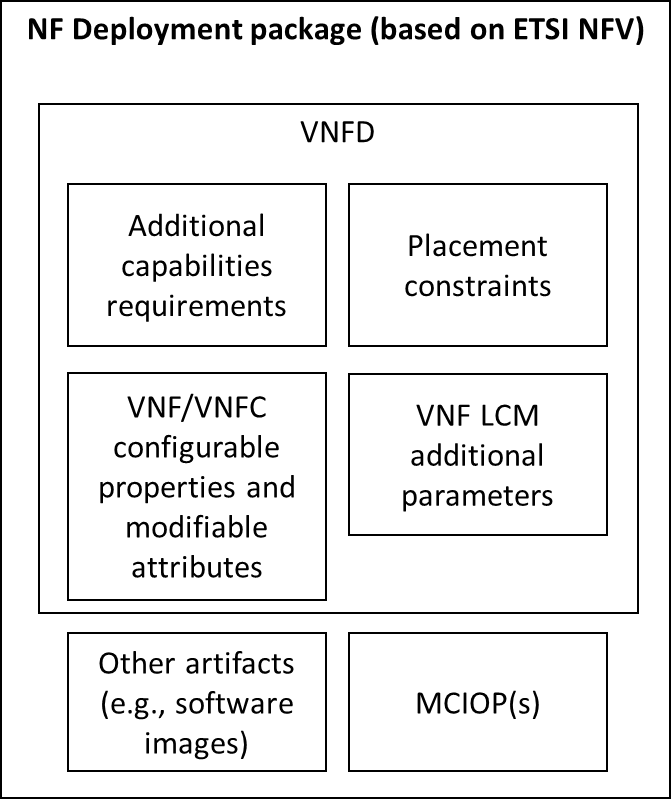
Annex A.2 describes examples related to AAL when using the O2dms ETSI NFV profile related specifications.

#### Description of acceleration requirements

NF Deployment descriptors can contain information and requirements about the acceleration capabilities and resources that are needed by an NF Deployment. The descriptors also provide the means for declaring configurable properties, extensions, or additional lifecycle parameters. These can be used to apply configuration related to acceleration capabilities and resources used by the NF Deployment. Figure 2.5.6.2-1 provides a high-level conceptual illustration of the main elements contained in the descriptors, which are summarized as follows:

* Additional capabilities requirements: description that is used to define requirements the NF Deployment has with respect to additional capabilities, typically, offered by the infrastructure.
* Placement constraints: description that indicates constraints and requirements regarding the placement rules for the NF Deployment and its constituents.
* Configurable properties and modifiable attributes: description about configuration parameters and information that can be modified on an NF Deployment.
* LCM additional parameters: description of parameters that are provided during NF Deployment lifecycle management operations.

These sets of information and relationship with acceleration related aspects are further described in the present and subsequent clauses.



##### Figure 2.5.6.2-1: Representation of elements in the NF Deployment package which can hold relevant configuration, capabilities, and placement requirements

Regarding the declaration of additional capabilities, in the case of the O2dms ETSI NFV profile and the use of the VNFD as defined in ETSI GS NFV-IFA 011 [[12],](#_bookmark14) requirements for additional capabilities of compute-related resources are defined as follows:

* In the case of OS container-based components of an NF Deployment, the "Vdu" information element specified in ETSI GS NFV-IFA 011 [[12]](#_bookmark14) includes the "requestAdditionalCapabilities" attribute, which is used to specify requirements for additional capabilities such as acceleration ones.
* In the case of VM-based components of an NF Deployment, the "VirtualComputeDesc" information element specified in ETSI GS NFV-IFA 011 [[12]](#_bookmark14) includes as well a “requestAdditionalCapabilities” attribute, with the same purpose as described in the item above.

Regarding network related resources, the "VirtualNetworkInterfaceRequirements" defined in the NF Deployment unit connection points (referred as "VduCpd" in ETSI GS NFV-IFA 011 [[12]](#_bookmark14)) and in NF Deployment external connection points (referred as "VnfExtCpd" in ETSI GS NFV-IFA 011 [[12]](#_bookmark14)) enable the NF designer to indicate standard and additional network interface requirements, e.g., to support SR-IOV or secondary container cluster network interfaces.

NOTE: The present document version does not specify the source(s) from where requested additional capabilities and virtual network interface requirements can be referenced into the NF Deployment descriptor.

The "RequestedAdditionalCapability" information element in NF Deployment descriptor can be used to describe for each of the components in the NF Deployment the requirements about what AAL profiles and other associated characteristics are needed by the NF Deployment. This can be described on a per deployable unit of the NF Deployment.

In addition, for the concrete case of OS container-based component of an NF Deployment, the "mcioConstraintsParams" in the "Vdu" information element provides parameter names for constraints expected to be assigned to the MCIO realizing the NF Deployment unit. ETSI GS NFV-IFA 011 [12] specifies the following MCIO constraint parameter values to express associated semantical constraint context and which are related to acceleration capabilities:

* "nodeAdditionalCapabilityDpdk": to indicate that the NF Deployment unit needs a cluster node with DPDK driver capability.
* "nodeAdditionalCapabilitySriov": to indicate that the NF Deployment unit needs a cluster node with SR-IOV card(s).
* "nodeAdditionalCapabilityGpu": to indicate that the NF Deployment unit needs a cluster node with GPU acceleration device(s).
* "nodeAdditionalCapabilityFpga": to indicate that the NF Deployment unit needs a cluster node with FPGA device(s).

More information about the relationship between NF Deployment and the concept of MCIO is available in clause 2.5.5.

The use of configurable properties, modifiable attributes, and LCM additional parameters for conveying relevant acceleration configuration information are further described in subsequent clauses.

#### O2dms interface exchanges

AAL LPU requirements for the NF Deployment as well as other deployment constraints and requirements can be provided to the DMS in two forms:

* On O2dms interface exchanges, and
* NF Deployment descriptors.

Regarding O2dms interface exchanges, acceleration capabilities and resources information cover the following aspects:

* MCIO constraints: these are used to provide values assigned to the MCIOs of an NF Deployment that has OS container-based components. This is provided as part of the "ResourcePlacement" data type specified in clause

3.2.6.5.3. The "key" part of the key-value pair of MCIO constraints is defined in the NF Deployment descriptor (see clause 2.5.6.2), while the "value" part of the key-value pair is the actual key-value pair used in the labelling of the O-Cloud Nodes in the O-Cloud Node Cluster.

* NF Deployment (and units) configurable properties and modifiable attributes: these are used to provide to the DMS runtime configuration and information, e.g., to cater for specific deployment scenarios or state.
* NF Deployment lifecycle additional parameters: these are used to provide to the DMS additional parameters applicable to the specific lifecycle management operation that is performed for the NF Deployment.

The actual requirements about the AAL profiles that are required by an NF Deployment is part of the NF Deployment descriptors (see previous clause 2.5.6.2). The NF Deployment descriptors are made available to the DMS, either as a pre-condition to initiate the lifecycle operation or during the lifecycle management procedures in which components of an NF Deployment (i.e., NF Deployment units) are created, updated, or terminated.

The O2dms ETSI NFV profile also supports the capability to provide runtime configuration related to acceleration capabilities in case that specific deployment information is expected and is not possible to be determined at design time of the NF Deployment via the corresponding descriptors (refer to items #2 and #3 in the list above). Additional description about these attributes is provided in clause 2.5.6.4.

Processing by the DMS of which concrete instances of LPU to allocate is regarded to be an implementation matter of the DMS. However, the DMS implementation shall be capable to process the expressed requirements provided via the NF Deployment descriptors together with the additional input configuration and information provided via the O2dms interface(s). In addition, DMS shall consider if a pre-selection of AAL resources and O-Cloud Nodes have been determined by the SMO. This information can be made available by the SMO to the DMS using the MCIO constraints based on specific values of O-Cloud Nodes labelling (see also clause 2.5.6.2).

#### Configuring NF Deployments with acceleration related information

* + - * 1. Overview

As described in clause 4.2.2 of O-RAN.WG6.AAL-Common-API [[32]](#_bookmark33) and clause 4.1.6 of O-RAN.WG6.AAL-GAnP.0 [[31],](#_bookmark32) AAL applications (as part of the NF Deployment) might need certain configuration about the assigned LPUs (e.g., LPU address in the form of a PCI device identification). In case configuration related to acceleration is to be injected into the NF Deployments via lifecycle management procedures, the O2dms ETSI NFV profile provides the following supporting mechanisms:

* Provisioning configuration data using day-0 mechanisms: in this case, the relevant configuration data is injected during the allocation of the O-Cloud resources and creation of the VM/OS containers realizing the NF Deployment unit.
* Provisioning configuration data using day-1 mechanisms: in this case, the relevant configuration data is injected after the O-Cloud resource has been allocated and the VM/OS container realizing the NF Deployment unit has been created, but before determining that the NF Deployment is fully operational and ready (i.e., instantiation is completed).

ETSI GR NFV-EVE 022 [[33]](#_bookmark34) reports about available VNF configuration mechanisms, and clause 4.2 of ETSI GR NFV-EVE 022 [[33]](#_bookmark34) includes descriptions about the meanings of "day-0" and "day-1" configuration in relation to instantiation process of VNF and VNFC instances, which are mapped to NF Deployment and NF Deployment units, respectively, in the present document.

* + - * 1. Day-0 configuration

For day-0 configuration, two mechanisms for providing configuration related to acceleration capabilities to the NF Deployment are defined depending on the form of deployment of the NF Deployment units constituting the NF Deployment:

* NF Deployment unit based on VM: the "boot data" defined in the NF Deployment descriptor contains strings or URL to files contained in the NF Deployment package that can be used to customize the VM compute resource at boot time. The boot data can contain variable parts that are replaced by deployment specific values during the NF Deployment lifecycle operation before sending the data to the infrastructure manager responsible for creating and setting up the VM.
* NF Deployment unit based on OS container: the configuration data is declared in the NF Deployment descriptor and in other configuration files within the MCIOPs (e.g., Helm® Charts). The technical realization of the mechanism depends on the container infrastructure management solution. When Kubernetes® and Helm® are used, Helm® Charts can contain values.yaml file(s) with key-value pairs. The DMS can create new values.yaml files to override the value in a key-value pair with the value provided by one or more configurable properties. The key-value pairs in the values.yaml can be used to parameterize Kubernetes® objects such as ConfigMap or Secret, or also be used by Init containers or combinations of commands an arguments declared in the Pod specification manifest.

NOTE: The mechanism of NF Deployment unit based on OS container can also be used for day-1 configuration.

Day-0 configuration data values related to acceleration capabilities can be provided to the DMS by the SMO according to the O2dms ETSI NFV profile using specific attributes of the "InstantiateVnfRequest" (see clause 5.5.2.4 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13)) and "VnfInfoModificationRequest" (see clause 5.5.2.12 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13)). The VNFD, as NF Deployment descriptor, specifies and indicates whether to use one or multiple of the attributes above, and which ones to use. Table 2.5.6.4.2-1 describes the relevant attributes for providing day-0 configuration to NF Deployments.

**Table 2.5.6.4.2-1: Interface attributes for providing day-0 configuration to NF Deployments**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data type in the O2dms ETSI NFV profile** | **Associated information elements and attributes in the descriptor (reference to ETSI GS NFV-IFA 011 [12]). See note**  **1** | **Description and usage** |
| additionalParams | InstantiateVnfRequest (see clause 5.5.2.4 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13)) | VnfLcmOperationsConfiguration  -> instantiateVnfOpConfig  -> parameter | Parameters that are specified for its use only during the instantiation of an NF Deployment.  It can be used to provide to the DMS acceleration related configuration only for VM-based NF Deployment units. It can be used to fill/modify as "volatile" values the variable parts of the  boot data. See note 2. |
| extensions | InstantiateVnfRequest and VnfInfoModificationRequest (see clause 5.5.2.12 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13)) | Vnfd.modifiableAttributes  -> extension | Parameters that are used to affect the lifecycle management of the NF Deployment. The attributes and values are stored persistently by the DMS and can be queried and modified. The parameters are intended to be consumed by the DMS or by the lifecycle management scripts during the execution of the NF Deployment lifecycle operation. The values can indirectly affect the configuration of the NF Deployment.  It can be used to provide to the DMS acceleration related configuration only for VM-based NF Deployment units. It can be used to fill/modify as "persistent" values the variable parts of the  boot data. See note 2. |
| vnfConfigurableP roperties | InstantiateVnfRequest and VnfInfoModificationRequest | Vnfd.configurableProperties  -> additionalConfigurableProperty and  Vnfd.vdu.configurableProperties | Parameters that are used to configure an NF Deployment. Modifying the values directly affects the configuration of an existing NF Deployment instance. |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | ->  additionalVnfcConfigurableProper ty | It can be used to provide to the DMS acceleration related configuration for both VM-based and OS container-based NF Deployment units.  In the case of VM-based NF Deployment units, it can be used to fill/modify as "persistent" values the variable parts of the boot data. See note 2.  In the case of OS container- based NF Deployment units, it can be used to provide runtime values to the configuration objects associated to the NF Deployment unit depending on their container infrastructure  management solution. |
| NOTE 1: NFV descriptors based on TOSCA are specified in ETSI GS NFV-SOL 001 [10]. Equivalent data types and attributes in TOSCA are referenced in clause 6.1 of ETSI GS NFV-SOL 001 [10].  NOTE 2: "Volatile" refers to values that are only applicable during the respective NF Deployment lifecycle management operation execution. "Persistent" refers to values that are applicable during the whole lifecycle of the NF Deployment. | | | |

* + - * 1. Day-1 configuration

For day-1 configuration, the O2dms ETSI NFV profile enables the SMO to provide NF Deployment configuration values using the attributes described in table 2.5.6.4.3-1.

##### Table 2.5.6.4.3-1: Interface attributes for providing day-1 configuration to NF Deployments

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Data type in the O2dms ETSI NFV profile** | **Associated information elements and attributes in the descriptor (reference to ETSI GS NFV-IFA 011 [12]). See note** | **Description and usage** |
| vnfConfigurableP roperties | InstantiateVnfRequest and VnfInfoModificationRequest | Vnfd.configurableProperties  -> additionalConfigurableProperty and  Vnfd.vdu.configurableProperties  ->  additionalVnfcConfigurableProper ty | Parameters that are used to configure an NF Deployment. Modifying the values affects directly the configuration of an existing NF Deployment instance.  It can be used to provide to the DMS acceleration related configuration for both VM-based  and OS container-based NF Deployment units. |
| NOTE: NFV descriptors based on TOSCA are specified in ETSI GS NFV-SOL 001 [10]. Equivalent data types and attributes in TOSCA are referenced in clause 6.1 of ETSI GS NFV-SOL 001 [10]. | | | |

The mechanisms by which the DMS applies the configuration depend on the form of deployment of the relevant NF Deployment units, and these are not "visible" over the interfaces specified in the present O2dms ETSI NFV profile. The mechanisms are regarded to be an implementation or integration matter of the different solutions that constitute an DMS. A description of potential solutions is provided for information purposes:

* In the case the NF Deployment is realized by VM-based technology and depending on the mechanism that the NF Deployment supports, possible mechanisms are: a) using the "VNF configuration" interface specified by ETSI GS NFV-SOL 002 [[29],](#_bookmark30) b) using other standard interfaces based on other protocols such as NETCONF,

c) using specific agent-less based configuration tools such as Ansible®.

* In the case the NF Deployment is realized by OS container-based technology, and depending on the mechanism that the NF Deployment supports, possible mechanisms are: a) using the "VNF configuration" interface specified by ETSI GS NFV-SOL 002 [[29],](#_bookmark30) b) using other standard interfaces based on other

protocols such as NETCONF, c) using specific agent-less based configuration tools such as Ansible, and in addition, d) using specific mechanisms supported by the container infrastructure management solution such as ConfigMap and Secrets in case of Kubernetes®.

NOTE: The Ansible® trademark is a registered trademark of Red Hat, Inc. in the United States, and other countries.

### DMS Interactions

#### Overview

As described in clause 2.3.2 and as per the present profile based on "O2-D demarcation #1" (see clause 2.1), the DMS has two main parts:

* NF Deployment management, and
* Resource management of NF Deployments, wherein resources can be virtualised resources and containerized workloads.

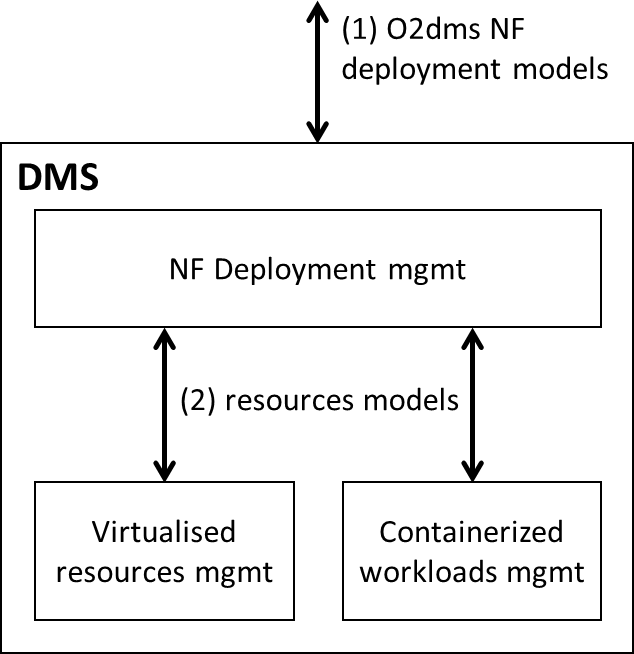
On the northbound, the DMS exposes the interfaces specified by the present document, which are at the NF Deployment level. Internally, the DMS performs the resources management to fulfil the necessary resources for the NF Deployment. Considering the types of resources (virtual resources and containerized workloads), interactions with other

components responsible for the management of these types of resources from the associated O-Cloud Node Clusters, are assumed to take place within the DMS. As described in table 2.3.2-2, according to the ETSI NFV standards, the VIM is the functional block in ETSI NFV responsible for the management of virtualised resources within an O-Cloud Node Cluster, and the CISM the function is responsible for the management of containerized workloads within an O-Cloud Node Cluster.

#### Models in DMS interactions

The present document specifies the protocol and data model of the O2dms interactions. The NF Deployment is the managed object exposed on O2dms, and the models are concerned to such a level of management. As described in clause 2.5.5, NF Deployments are realized by one or multiple instances of types of resources: in VM-based deployment, these are referred as virtualised resources, and for OS container-based deployments, these are referred as MCIOs (but more typically referred as containerized workloads).

The virtualised resources and containerized workloads management parts of the DMS may use specific resources models tailored for the types of resources realizing the NF Deployment. The resources models (indicated by (2) in figure 2.5.7.2-1) can contain information defined at design time of the NF Deployment, but also be complemented by information extracted (coming from) the O2dms interactions based on NF deployment models (indicated by (1) in figure 2.5.7.2-1).



##### Figure 2.5.7.2-1: Models in interactions

The O2dms NF deployment models comprise two main aspects:

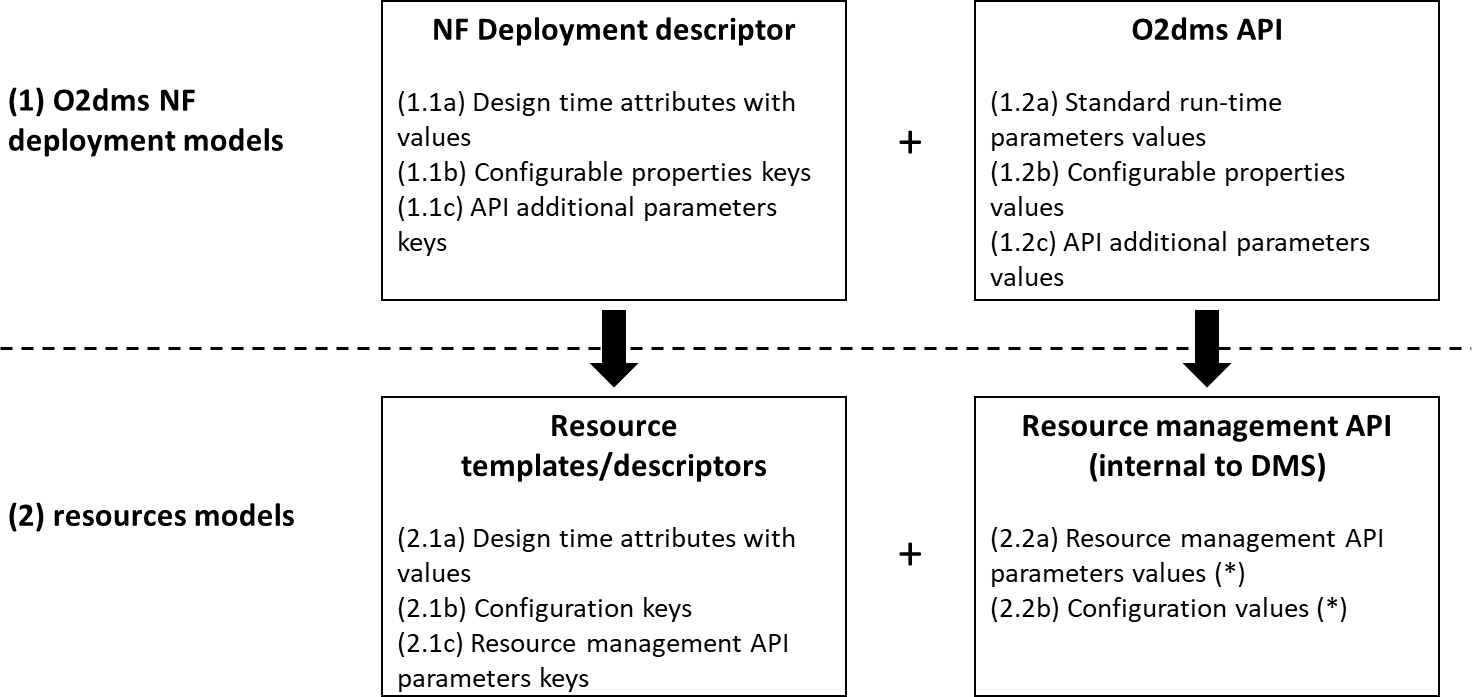
* NF Deployment design-time information: this is also referred as NF Deployment descriptor, and
* NF Deployment run-time information and management interfaces: this corresponds to the O2dms ETSI NFV profile modelling specified in the present document.

The resources models can comprise, as well, two main aspects:

* Resources run-time information and management interfaces: these correspond to interfaces and information provided by the virtualised resources and containerized workloads management parts of the DMS.
* (Optionally) declarative descriptors for resources management: these can be manifests or templates modelling the actual resources and that can be used to facilitate the interactions towards the resources management functions.

According to the present O2dms ETSI NFV profile, the NF Deployment models provide a generic modelling for managing NF Deployments, irrespective of the types of resources that comprise them. It is a responsibility of the NF Deployment management functions to resolve the necessary resource type specific models. For this, the NF Deployment management functions may, but need not, make use of resources models manifests or templates contained in the O- RAN NF software package.

Figure 2.5.7.2-2 illustrates an overview of the set of models and indicative sets of content in respective descriptors/template/manifests and APIs.



##### Figure 2.5.7.2-2: Illustration of content in the O2dms and resource management models

Table 2.5.7.2-1 summarizes the relationship between the sets of content in the O2dms and resource management models.

**Table 2.5.7.2-1: Relationship between sets of content in the o2dms and resource management models**

|  |  |  |  |
| --- | --- | --- | --- |
| **Group** | **Content set** | **Description** | **Relationship** |
| NF Deployment descriptor | 1.1a | Standardized design time attributes with values provided by the NF designer.  Attributes can concern to NF Deployment and resource properties/requirements. | Resource related properties/requirements can relate to 2.1a. |

|  |  |  |  |
| --- | --- | --- | --- |
|  | 1.1b | Configurable properties keys, some can be defined by standard, some can be defined by the NF designer.  Properties can concern to NF Deployment and resource related configuration. | Part of the attributes can relate to 2.1b. |
| 1.1c | Definition of keys for additional parameters defined by the NF designer. Parameters can concern to NF Deployment management and resource  related management. | Part of the parameters can relate to 2.1c. |
| O2dms API | 1.2a | Standardized parameters defined by the API with capability to provide values at run-time.  Parameters can concern to NF Deployment management, and some can concern to resource properties. | Values for the 1.1a.  Part of the parameters can relate to 2.2a. |
| 1.2b | Provides values for the NF Deployment configurable properties. | Values for the 1.1b. |
| 1.2c | Provides values for the API additional parameters. | Values for the 1.1c. |
| Resource templates/descriptors | 2.1a | Attributes and values of resources of the NF Deployment defined according to the  resource templates/descriptors data models. | Some attributes and values can also be defined on the 1.1a. |
| 2.1b | Definition of parameter keys for resource- level configuration. | Some of the configuration values can also be defined in 1.1b. |
| 2.1c | Definition of parameters keys related to resource management defined by the NF designer. | Keys can relate to 2.1a for making parts of the templates parameterizable. |
| Resource management API | 2.2a | Provides values for the resource management API parameters keys. | Values for the 2.1c. See note. |
| 2.2b | Provides values for the resource-level configuration | Values for the 2.1b. See note. |
| NOTE: Depending on the form of the API, this data could be provided via files (e.g., a "values.yaml" in the case of Helm), instead of actual content in the API messages. | | | |

### Monitoring of NF Deployments

#### Overview

NF Deployments are realized by one or multiple instances and types of resources, such as compute, storage, and network. Associated to NF Deployments, as managed objects, performance information and faults can be monitored.

#### Performance monitoring

Performance monitoring is performed via the present O2DMS ETSI NFV profile via the exposure of a corresponding API produced by the DMS, the O2dms\_DeploymentPerformance Service API. The monitoring is performed via:

* PM Jobs, and
* Thresholds.

Collection and reporting of performance information is controlled via PM Jobs.

The performance information is associated to managed object instances, which in the case of the O2DMS, these are the NF Deployments. In addition, sub-object instances can also be further qualified. These are determined depending on the type of metrics that can be monitored from the NF Deployment. For instance, the NF Deployment serves as the main object type, while an NF Deployment unit (aka as VNFC in the case of ETSI NFV) is a kind of sub-object that is part of an NF Deployment. A consumer, such as the SMO, can learn about the managed objects, such as the NF Deployment, via the O2dms\_DeploymentLifecycle Service.

Performance metrics associated to VNF instances, and consequently to NF Deployments (as per the current profile) are specified in ETSI GS NFV-IFA 027 [[34].](#_bookmark35)

#### Fault monitoring

Fault monitoring is performed via the present O2DMS ETSI NFV profile via the exposure of a corresponding API produced by the DMS, the O2dms\_DeploymentFault Service API. The monitoring is performed via Alarms, which are raised to provide information about faults and failures associated to the managed objects of concern.

Like in the case of performance monitoring, NF Deployment serves as the main object type, while other sub-objects are also identified to which specific fault information can be associated. A consumer, such as the SMO, can learn about the managed objects, such as the NF Deployment, via the O2dms\_DeploymentLifecycle Service.

Standard alarms associated to VNF instances, and consequently to NF Deployments (as per the current profile) are specified in ETSI GS NFV-IFA 045 [[35].](#_bookmark36)

# API definitions

## General aspects

### Introduction

The present document defines the protocol and data model for the following O2dms service interfaces in the form of RESTful Application Programming Interface (API) specifications:

* O2dms\_DeploymentLifecycle Service API (as produced by the DMS towards the SMO).
* O2dms\_DeploymentFault Service API (as produced by the DMS towards the SMO).
* O2dms\_DeploymentPerformance Service API (as produced by the DMS towards the SMO). NOTE: Other service APIs are expected to be defined in future versions of the present document.

Table 3.1.1-1 lists the versions of the APIs defined in the present document.

##### Table 3.1.1-1: Versions of the APIs specified in the present document

|  |  |
| --- | --- |
| **API** | **API version** |
| O2dms\_DeploymentLifecycle Service API | 2.12.0 |
| O2dms\_DeploymentFault Service API | 1.12.0 |
| O2dms\_DeploymentPerformance Service API | 2.12.0 |

In clause 3, general aspects are specified that apply to multiple APIs for O2dms services. In addition, the provisions in clauses 3.1.2, 3.1.3, 3.1.4, 3.1.5, 3.1.6, 3.1.7, and 3.1.8 define common aspects of RESTful APIs, and shall apply for all APIs defined in the present document.

In the subsequent clauses, the protocol and data model for the individual interfaces are specified. Per interface, the resource structure with associated HTTP methods is defined and applicable flows are provided, if necessary. Further, the resources and the data model are specified in detail.

Even though the different interfaces defined in the present document are related, implementations shall not assume a particular order of messages that arrive via different interfaces.

### URI structure and supported content formats

This clause specifies the URI prefix and the supported formats applicable to the O2dms RESTful APIs based on the ETSI NFV profile.

All resource URIs of the APIs shall have the following prefix, except the "API versions" resource which shall follow the rules specified in clause 9.3 of ETSI GS NFV-SOL 013 [[15]:](#_bookmark17)

##### {apiRoot}/<apiName>/<apiMajorVersion>/

The request URIs used in HTTP requests from the API service consumer towards the API service producer shall have the Resource URI structure defined in clause 4.1 of ETSI GS NFV-SOL 013 [[15]](#_bookmark17) and further qualified as follows:

##### {apiRoot}/<apiName>/<apiMajorVersion>/<apiSpecificResourceUriPart>

with the following components:

* The {apiRoot} indicates the scheme ("https"), the host name and optional port, and an optional sequence of path segments that together represent a prefix path,
* The <apiName> indicates the API name of the service interface in an abbreviated form. The {apiName} of each interface is defined in the clause specifying the corresponding O2dms RESTful API,
* The <apiMajorVersion> indicates the current major version of the API and is defined in the clause specifying the corresponding O2dms RESTful API, and
* The <apiSpecificResourceUriPart> indicates a resource URI of the API, and shall be set as described in the corresponding O2dms RESTful API for each one of the defined resources.

HTTP/1.1, as defined in IETF RFC 7230 [[24],](#_bookmark26) shall be used. The Transmission Control Protocol (TCP) as specified in IETF RFC 793 [[20]](#_bookmark22) shall be used as transport protocol for HTTP/1.1.

For HTTP requests and responses that have a body, the content format JSON (see IETF RFC 8259 [[25]](#_bookmark27)) shall be supported. The JSON format shall be signalled by the content type "application/json".

All APIs shall support and use HTTP over TLS (also known as HTTPS) (see IETF RFC 2818 [[36]](#_bookmark37)). All APIs shall use TLS version 1.2 as defined by IETF RFC 5246 [[23]](#_bookmark25) or later.

NOTE 1: The HTTP protocol elements mentioned in the O2dms RESTful APIs originate from the HTTP specification; HTTPS runs the HTTP protocol on top of a TLS layer. For simplicity, the O2dms RESTful APIs specifications therefore use the statement above to mention "HTTP request", "HTTP header", etc., without explicitly calling out HTTPS.

NOTE 2: There are a number of best practices and guidelines how to configure and implement TLS 1.2 in a secure manner, as security threats evolve. A detailed specification of those is beyond the scope of the present document; the reader is referred to external documentation such as annex E of ETSI TS 133 310 [[37].](#_bookmark38)

All resource URIs of the API shall comply with the URI syntax as defined in IETF RFC 3986 [[22].](#_bookmark24) An implementation that dynamically generates resource URI parts (individual path segments, sequences of path segments that are separated by "/", query parameter values) shall ensure that these parts only use the character set that is allowed by IETF

RFC 3986 [[22]](#_bookmark24) for these parts.

NOTE 3: This means that characters not part of this allowed set are escaped using percent-encoding as defined by IETF RFC 3986 [[22].](#_bookmark24)

Unless otherwise specified explicitly, all request URI parameters that are part of the path of the resource URI shall be individual path segments, i.e., shall not contain the "/" character.

NOTE 4: A request URI parameter is denoted by a string in curly brackets, e.g. {fooId}.

### Usage of HTTP header fields

#### General

HTTP headers are components of the header section of the HTTP request and response messages. They contain the information about the server/client and metadata of the transaction. The use of HTTP header fields shall comply with the provisions defined for those header fields in the specifications referenced from tables 3.1.3.2-1 and 3.1.3.3-1. The following clauses describe more details related to selected HTTP header fields.

#### Request header fields

The request header fields that shall be supported by O2dms RESTful APIs are as specified in clause 4.2.2 of ETSI GS NFV-SOL 013 [[15].](#_bookmark17)

NOTE: ETSI GS NFV-SOL 013 [[15]](#_bookmark17) refers to "RESTful NFV-MANO API" specification; and wherever such reference is provided "O2dms RESTful API" is to be considered instead for the purpose of the present document.

#### Response header fields

The response header fields that shall be supported by O2dms RESTful APIs are as specified in clause 4.2.3 of ETSI GS NFV-SOL 013 [[15].](#_bookmark17)

NOTE: ETSI GS NFV-SOL 013 [[15]](#_bookmark17) refers to "RESTful NFV-MANO API" specification; and wherever such reference is provided "O2dms RESTful API" is to be considered instead for the purpose of the present document.

### Result set control

#### Introduction

This clause specifies procedures that allow to control the size of the result set of GET requests w.r.t. the number of entries in a response list (using attribute-based filtering) or w.r.t. the number of attributes returned in a response (using attribute selection).

#### Attribute-based filtering and selector

The use of attribute-based filtering shall be supported by O2dms RESTful APIs as specified in clause 5.2.2 and 5.3.2 of ETSI GS NFV-SOL 013 [[15].](#_bookmark17)

NOTE: ETSI GS NFV-SOL 013 [[15]](#_bookmark17) refers to "RESTful NFV-MANO API" specification; and wherever such reference is provided "O2dms RESTful API" is to be considered instead for the purpose of the present document.

#### Handling of large query results

The handling of large query results shall be supported by O2dms RESTful APIs as specified in clause 5.4.2 of ETSI GS NFV-SOL 013 [[15].](#_bookmark17)

### Error reporting

In RESTful interfaces, application errors are mapped to HTTP errors. Since HTTP error information is generally not enough to discover the root cause of the error, additional application specific error information is typically delivered.

The error reporting shall be supported by O2dms RESTful APIs as specified in clause 6 of ETSI GS NFV-SOL 013 [[15].](#_bookmark17)

NOTE: ETSI GS NFV-SOL 013 [[15]](#_bookmark17) refers to "RESTful NFV-MANO API" specification; and wherever such reference is provided "O2dms RESTful API" is to be considered instead for the purpose of the present document.

### Common data types

#### Structured data types

* + - * 1. Introduction

This clause defines data structures that are referenced from data structures in multiple O2dms RESTful APIs.

* + - * 1. Introduction

An object contains structured data and shall comply with the provisions of clause 4 of IETF RFC 8259 [[25].](#_bookmark27)

* + - * 1. Type: Link

This type represents a link to a resource using an absolute URI. It shall comply with the provisions defined in table 3.1.6.1.3-1.

**Table 3.1.6.1.3-1: Definition of the Link data type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Cardinality** | **Description** |
| href | Uri | 1 | URI of another resource referenced from a resource. Shall be an absolute URI (i.e., a URI that contains {apiRoot}). |

* + - * 1. Type: NotificationLink

This type represents a link to a resource in a notification, using an absolute or relative URI. It shall comply with the provisions defined in table 3.1.6.1.4-1.

**Table 3.1.6.1.4-1: Definition of the NotificationLink data type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Cardinality** | **Description** |
| href | Uri | 1 | URI of a resource referenced from a notification.  Should be an absolute URI (i.e., a URI that contains {apiRoot}), however, may be a relative URI (i.e., a URI where the {apiRoot} part is omitted) if the  {apiRoot} information is not available. |

* + - * 1. Type: KeyValuePairs

This type represents a list of key-value pairs. The order of the pairs in the list is not significant. In JSON, a set of key- value pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 8259 [25In the following example, a list of key-value pairs with four keys ("aString", "aNumber", "anArray" and "anObject") is provided to illustrate that the values associated with different keys can be of different type.

EXAMPLE:

{

"aString" : "O2dms service", "aNumber" : 0.03,

"anArray" : [1,2,3],

"anObject" : {"organization" : "O-RAN", workingGroup" : "WG6"}

}

* + - * 1. Type: ApiVersionInformation

This type represents API version information. It shall comply with the provisions defined in table 3.1.6.1.6-1.

**Table 3.1.6.1.6-1: ApiVersionInformation data type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Cardinality** | **Description** |
| uriPrefix | String | 1 | Specifies the URI prefix for the API, in the following form  {apiRoot}/{apiName}/{apiMajorVersion}/. |
| apiVersions | Structure (inlined) | 1..N | Version(s) supported for the API signaled by the uriPrefix attribute. |
| >version | String | 1 | Identifies a supported version. The value of the version attribute shall be a version identifier as specified in clause 9.1 of ETSI GS NFV-SOL 013 [[15].](#_bookmark17) |
| >isDeprecated | Boolean | 0..1 | If such information is available, this attribute indicates whether use of the version signaled by the version attribute is deprecated (true) or not (false).  See note. |
| >retirementDate | DateTime | 0..1 | The date and time after which the API version will no longer be supported.  This attribute may be included if the value of the isDeprecated attribute is set to true and shall be absent otherwise. |
| NOTE: A deprecated version is still supported by the API producer but is recommended not to be used any longer.  When a version is no longer supported, it does not appear in the response body. | | | |

#### Simple data types and enumerations

* + - * 1. Introduction

This clause defines simple data types and enumerations that can be referenced from data structures defined in multiple interfaces.

* + - * 1. Simple data types

Table 3.1.6.2.2-1 lists the simple data types that are referenced from multiple interfaces.

**Table 3.1.6.2.2-1: Simple data types**

|  |  |
| --- | --- |
| **Type name** | **Description** |
| Identifier | An identifier with the intention of being globally unique. Representation: string of variable length. See note 1. |
| DateTime | A date-time stamp. Representation: String formatted as defined by the date-time production in IETF RFC 3339 [[21].](#_bookmark23) |
| Uri | A string formatted according to IETF RFC 3986 [[22].](#_bookmark24) |
| Boolean | A data type having two values (true and false). |
| IpAddress | An IPV4 or IPV6 address. Representation: In case of an IPV4 address, string that consists of four  decimal integers separated by dots, each integer ranging from 0 to 255. In case of an IPV6 address, string that consists of groups of zero to four hexadecimal digits, separated by colons. |
| Version | A version. Representation: string of variable length. |
| String | A string as defined in IETF RFC 8259 [[25].](#_bookmark27) |
| Number | A number as defined in IETF RFC 8259 [[25].](#_bookmark27) |
| Integer | An integer, i.e., a number that cannot have a fractional component. See note 2. |
| UnsignedInt | An unsigned integer, i.e., an integer that can't assume negative values. See note 2. |
| NOTE 1: Individual API specifications are assumed to define types for additional identifiers with dedicated scope (e.g., identifiers scoped by some O-Cloud).  NOTE 2: In the JSON instance data model, only the concept of a "number" is used to represent numerical data.  Numbers in JSON can be integral, i.e., have no fractional part, or can include a fractional part. The additional numeric types defined in the present document represent constraints on the general "number" type present in JSON instances which can be enforced e.g., during parsing when processing the JSON instance or expressed as constraints in modelling languages such as JSON Schema [[26]](#_bookmark28) or OpenAPI [[27].](#_bookmark29) | |

### Authorization of API requests and notifications

#### Introduction

The present version of this document does not specify specific authorization mechanisms to be considered for handling API requests and notifications specified in the present document. The provisions specified in clause 8 of ETSI GS NFV- SOL 013 [[15]](#_bookmark17) shall apply. Furthermore, annex G of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) defines authorization scope values for selected APIs that are profiled in the present document.

NOTE: ETSI GS NFV-SOL 013 [[15]](#_bookmark17) refers to "RESTful NFV-MANO API" specification; and wherever such reference is provided "O2dms RESTful API" is to be considered instead for the purpose of the present document.

### Version management

#### Version identifiers and parameters

API version identifiers and rules for incrementing version identifier fields shall be supported by O2dms RESTful APIs as specified in clause 9.1 and 9.2 of ETSI GS NFV-SOL 013 [[15].](#_bookmark17)

NOTE: ETSI GS NFV-SOL 013 [[15]](#_bookmark17) refers to "RESTful NFV-MANO API" specification and OpenAPI specification that ETSI publishes; and wherever such reference is provided "O2dms RESTful API" and OpenAPI that O-RAN publishes is to be considered instead for the purpose of the present document.

Clause 9.2.2 of ETSI GS NFV-SOL 013 [[15]](#_bookmark17) provides examples of backward and non-backward compatible changes.

#### Version information retrieval and signaling

The API producer shall support the dedicated URIs to enable API consumers to retrieve information about API versions supported by an API producer as specified in clause 9.3 of ETSI GS NFV-SOL 013 [[15],](#_bookmark17) and the API consumer shall include the "Version" HTTP header in each HTTP request as specified in clause 9.4 of ETSI GS NFV-SOL 013 [[15].](#_bookmark17)

NOTE: ETSI GS NFV-SOL 013 [[15]](#_bookmark17) refers to "RESTful NFV-MANO API" specification; and wherever such reference is provided "O2dms RESTful API" is to be considered instead for the purpose of the present document.

### HTTP conditional requests

Conditional requests are HTTP (POST, PUT, PATCH and DELETE) requests that include one or more header fields indicating a precondition to be tested before applying the method semantics to the target resource. They are supported by metainformation about the resource that was provided in earlier HTTP responses. The set of HTTP header fields to compose a conditional request allowed in the present version of the present document is defined in clause 4.8 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

## O2dms\_DeploymentLifecycle Service API

### Description

This service interface enables an API consumer (e.g., the SMO) to invoke lifecycle management operations of NF Deployments towards the DMS. The service interface also enables the management of subscriptions to notifications raised by the DMS regarding the lifecycle of the NF Deployments.

### API version

For the O2dms\_DeploymentLifecycle Service API version as specified in the present document, the MAJOR version field shall be "2", the MINOR version field shall be "12", and the PATCH version field shall be "0" as specified in clause 5.1a of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) Consequently, the {apiMajorVersion} URI variable shall be set to "v2".

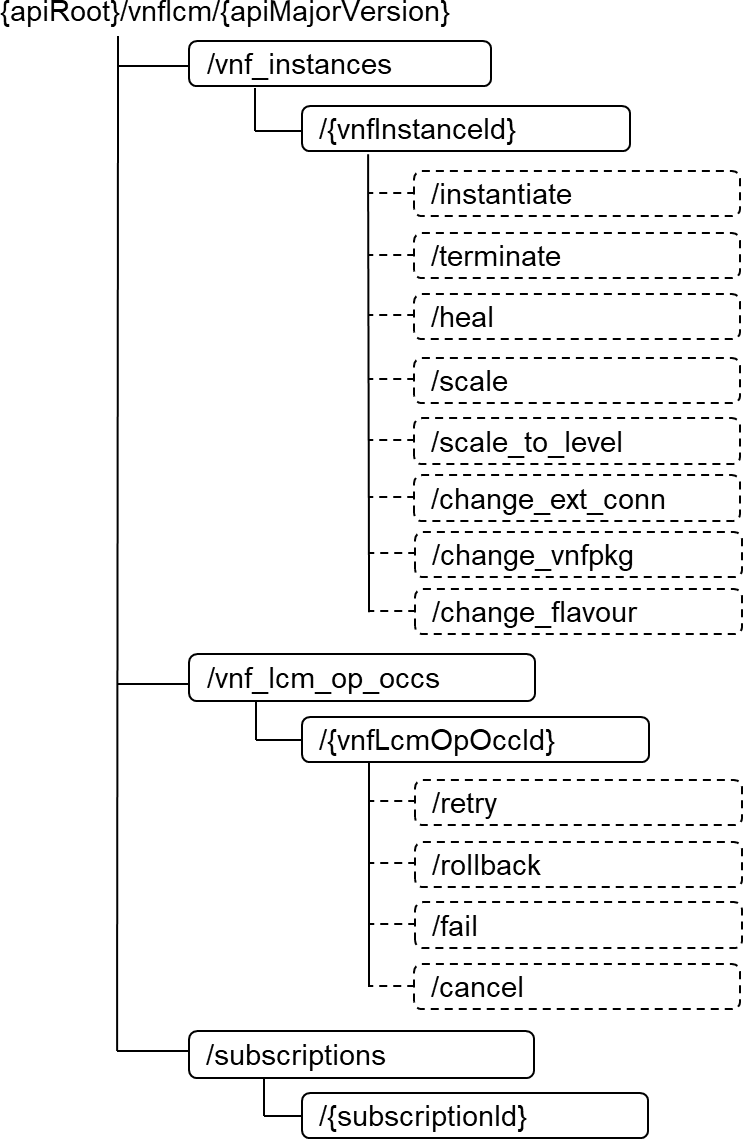
### REST resources structure and methods

All resource URIs of the API shall use the base URI specified defined in clause 3.1.2. The string "vnflcm" shall be used to represent {apiName} as defined in clause 5.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) All resource URIs in clauses below are defined relative to the formed based URI (i.e., {apiRoot}/xyz/{apiVersion}).

When ambiguity is possible, the term REST resource is used to make the distinction between the term resource as understood within the context of REST API from the term resource as understood within the context of O-RAN.

The resource URI structure defined for the O2dms\_DeploymentLifecycle Service API shall be as specified in clause 5.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) Applicable individual REST resources for the present version of the O2dms\_DeploymentLifecycle Service API are listed in table 3.2.3-1.

Figure 3.2.3-1 illustrates the overall resource URI structure defined for the O2dms\_DeploymentLifecycle Service API.



##### Figure 3.2.3-1: Resource URI structure of the O2dms\_DeploymentLifecycle Service API

Table 3.2.3-1 lists the individual REST resources defined and the applicable HTTP methods. The DMS shall support responding to requests for all HTTP methods on the resources in table 3.2.3-1 that are marked as "M" (mandatory) in the "Cat" column.

**Table 3.2.3-1: Resources and methods overview**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Resource name** | **Resource URI** | **HTTP**  **method** | **Cat** | **Description (see note 2)** |
| VNF instances | /vnf\_instances See note 1. | GET | M | Query multiple NF Deployments. |
| POST | M | Create a new "Individual VNF instance" resource which  represents an individual NF Deployment. |
| Individual VNF instance | /vnf\_instances/{vnfInstanceId} See note 1. | GET | M | Read information about an individual NF Deployment. |
| PATCH | M | Modify information of an individual NF Deployment. |
| DELETE | M | Delete an "Individual VNF instance" resource representing an individual NF  Deployment. |
| Instantiate VNF task | /vnf\_instances/{vnfInstanceId}/instantiate  See note 1. | POST | M | Instantiate an NF Deployment. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Terminate VNF task | /vnf\_instances/{vnfInstanceId}/terminate  See note 1. | POST | M | Terminate an NF Deployment instance. |
| Heal VNF task | /vnf\_instances/{vnfInstanceId}/heal  See note 1. | POST | M | Heal an NF Deployment instance. |
| Scale VNF task | /vnf\_instances/{vnfInstanceId}/scale  See note 1. | POST | M | Scale an NF Deployment instance. |
| Scale VNF to Level task | /vnf\_instances/{vnfInstanceId}/scale\_to\_level  See note 1. | POST | M | Scale an NF Deployment instance to a target level. |
| Change external VNF connectivity task | /vnf\_instances/{vnfInstanceId}/change\_ext\_conn  See note 1. | POST | M | Change the external connectivity of an NF Deployment instance. |
| Change current VNF package task | /vnf\_instances/{vnfInstanceId}/change\_vnfpkg  See note 1. | POST | M | Change the current package on which an NF Deployment is  based. |
| Change VNF flavour task | /vnf\_instances/{vnfInstanceId}/change\_flavour  See note 1. | POST | M | Change the flavour of an NF Deployment instance. |
| VNF LCM  operation occurrences | /vnf\_lcm\_op\_occs  See note 1. | GET | M | Query information about multiple lifecycle management operation occurrences of NF Deployments. |
| Individual VNF LCM operation occurrence | /vnf\_lcm\_op\_occs/{vnfLcmOpOccId} See note 1. | GET | M | Query information about an "Individual VNF LCM operation occurrence" corresponding to the lifecycle of an NF  Deployment. |
| Retry operation task | /vnf\_lcm\_op\_occs/(vnfLcmOpOccId}/retry  See note 1. | POST | M | Retry a lifecycle management operation occurrence of an NF  Deployment |
| Rollback operation task | /vnf\_lcm\_op\_occs/(vnfLcmOpOccId}/rollback  See note 1. | POST | M | Rollback a lifecycle management operation occurrence of an NF  Deployment |
| Fail operation task | /vnf\_lcm\_op\_occs/(vnfLcmOpOccId}/fail  See note 1. | POST | M | Fail a lifecycle management operation occurrence of an NF  Deployment |
| Cancel operation task | /vnf\_lcm\_op\_occs/(vnfLcmOpOccId}/cancel  See note 1. | POST | M | Cancel a lifecycle management operation occurrence of an NF  Deployment |
| Subscriptions | /subscriptions See note 1. | POST | M | Subscribe to NF Deployment lifecycle notifications. |
| GET | M | Query information about multiple subscriptions to NF Deployment lifecycle notifications. |
| Individual subscription | /subscriptions/{subscriptionId} See note 1. | GET | M | Read an "Individual subscription" resource which represents information about a subscription to NF Deployment  lifecycle notifications. |
| DELETE | M | Terminate a subscription to NF Deployment lifecycle  notifications. |
| Notification endpoint | (provided by API consumer) See note 1. | POST | See note 3 | Notify about NF Deployment lifecycle events. |
| GET | See note 3 | Test the notification endpoint for NF Deployment lifecycle events. |
| NOTE 1: The resource URI value is the same as the one specified in ETSI GS NFV-SOL 003 [[11]](#_bookmark13).  NOTE 2: The specified meaning of the resource name and supported HTTP method in ETSI GS NFV-SOL 003 [[11]](#_bookmark13) applies, unless otherwise stated. Additional description can be presented in case of further refining its  meaning in the context of O2dms specification. | | | | |

NOTE 3: The DMS shall support invoking the HTTP methods defined for the "Notification endpoint" resource exposed by the API consumer. If the API consumer supports invoking the POST method on the "Subscriptions" resource towards the DMS, it shall also support responding to the HTTP requests defined for the "Notification endpoint" resource.

### REST resources

#### General

The DMS and API consumer of the O2dms\_DeploymentLifecycle Service API shall support all the provisions specified in clause 5.4 of ETSI GS NFV-SOL 003 [[11],](#_bookmark13) with the following considerations:

* NFVO and VNFM entities referred in the ETSI GS NFV-SOL 003 [[11]](#_bookmark13) are mapped to SMO and DMS as defined in clause 2.3.2 of the present document.
* VNF and VNFC referred in the ETSI GS NFV-SOL 003 [[11]](#_bookmark13) are mapped to NF Deployment and NF Deployment unit as defined in clause 2.3.2 of the present document.
* Only the provisions associated to corresponding REST resources and methods listed in table 3.2.3-1 are applicable for the present document. That is, if a REST resource and method is not listed in table 3.2.3-1 of the present document, the corresponding clause in the ETSI GS NFV-SOL 003 [[11]](#_bookmark13) that specifies such REST resource, and its supported methods, is not required to be supported according to the present document version. If specific provisions to REST resources are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.2.4 of the present document.
* For the list of referenced data types for notifications to support, these are defined in clause 3.2.5 of the present document.
* For the list of REST resource data types, structured data types and referenced simple types to be supported, refer to clauses 3.2.6.1, 3.2.6.2, 3.2.6.3 and 3.2.6.4 of the present document.

NOTE 1: From version v07.00 of the present document, several resource and data model clauses have been voided to simplify the profile specification, since no specific provisions and/or considerations had been specified with respect to the baseline specification in ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

As defined in clause 5.4.1.2 of ETSI GS NFV-SOL 003 [[11],](#_bookmark13) to handle the lifecycle operations of the managed entities a number of task resources are defined. These are related to potentially long-running operations such as instantiation and termination, in the case of the present document, of NF Deployments. Each occurrence of the NF Deployment lifecycle operation is represented by an "Individual VNF LCM operation occurrence". In the present document, the pre- conditions and postconditions for the operations defined in table 5.4.1.2-1 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) that apply are only:

* Operation "Instantiate VNF", i.e., "Instantiate NF Deployment".
* Operation "Terminate VNF", i.e., "Terminate NF Deployment".
* Operation “Heal VNF”, i.e., “Heal NF Deployment”.
* Operation "Scale VNF", i.e., "Scale NF Deployment".
* Operation "Scale VNF to Level", i.e., "Scale NF Deployment to Level".
* Operation "Change external VNF connectivity", i.e., "Change external NF Deployment connectivity".
* Operation "Change current VNF package", i.e., "Change current NF Deployment package".
* Operation "Change VNF flavour", i.e., "Change NF Deployment flavour".

In addition, the following specific DMS applicability conditions are considered with respect to the referenced specification regarding the NF Deployment lifecycle:

* As part of the NF Deployment lifecycle processing for all relevant supported LCM operations, the DMS need not proceed with obtaining the grant.

NOTE 2: Granting exchanges between DMS and SMO are not supported in the present document version.

* Selected deployable modules of an NF Deployment can be requested to be instantiated as part of the "Instantiate NF Deployment". However, the deployable modules cannot be changed or additional ones be selected once the NF Deployment is instantiated, since the present document version does not profile the necessary task resource enabling such a capability.

#### REST resource: VNF instances

* + - * 1. Description

This resource represents NF Deployment instances. The API consumer can use this resource to create and query "Individual VNF instance" resources, which represent the NF Deployments.

* + - * 1. Resource definition

The resource definition is defined in clause 5.4.2.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Resource methods

POST

The POST method for this resource is defined in clause 5.4.2.3.1 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

The "VNF package" represents the packaging of the NF Deployment as described in clause 2.3.2 of the present document. The packaging is expected to be onboarded and enabled as a condition for successfully creating the “Individual VNF instance” resource. The specification of how the VNF package is onboarded is outside the scope of the present document.

The VNFD represents the NF Deployment descriptor as described in clause 2.3.2 of the present document. The DMS applies information and metadata from the VNFD into the "Individual VNF instance" resource as specified in clause 5.4.2.3.1 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - 1. Void
      2. Void
      3. Void
      4. Void
      5. REST resource: Retry operation task
         1. Description

This task resource represents the "Retry operation". The API consumer can use this resource to request the retrying of a lifecycle management operation occurrence on an NF Deployment that is in a transient failure state. More information about error handling in NF Deployment lifecycle procedures is available in clause 3.2.7.

* + - * 1. Void
        2. Void

#### REST resource: Rollback operation task

* + - * 1. Description

This task resource represents the "Rollback operation". The API consumer can use this resource to request the rolling back of a lifecycle management operation occurrence on an NF Deployment. More information about error handling in NF Deployment lifecycle procedures is available in clause 3.2.7.

* + - * 1. Resource definition

The resource definition is defined in clause 5.4.15.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Resource methods

POST

The POST method for this resource is defined in clause 5.4.15.3.1 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following specific DMS applicability conditions are considered with respect to the referenced specification clause:

In case of rolling back an occurrence of the NF Deployment instantiation lifecycle operation, the DMS shall request to underlying infrastructure manager the release of the virtualised resources and/or containerized workloads that had been allocated to the NF Deployment.

Void

Void

Void

Void

#### REST resource: Fail operation task

* + - * 1. Description

This task resource represents the "Fail operation". The API consumer can use this resource to request marking the lifecycle management operation occurrence of an NF Deployment as "failed". More information about error handling in NF Deployment lifecycle procedures is available in clause 3.2.7.

* + - * 1. Void
        2. Void

#### REST resource: Cancel operation task

* + - * 1. Description

This task resource represents the "Cancel operation". The API consumer can use this resource to request cancelling an ongoing lifecycle management operation occurrence of an NF Deployment. More information about error handling in NF Deployment lifecycle procedures is available in clause 3.2.7.

* + - * 1. Void
        2. Void
      1. Void
      2. Void
      3. Void
      4. Void

### Notifications

#### General

Notifications issued on the O2dms\_DeploymentLifecycle Service API shall comply to the Subscribe/Notify pattern described in clause 5.9 of ETSI GS NFV-SOL 015 [[17].](#_bookmark19)

Table 3.2.5.1-1 provides an overview of the notifications.

**Table 3.2.5.1-1: O2dms\_DeploymentLifecycle notifications overview**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Clause defined** | **Description** |
| VnfLcmOperationOccurrenceNotification | 3.2.5.2 | This type represents the notification about an NF Deployment lifecycle management operation occurrence. |

|  |  |  |
| --- | --- | --- |
| VnfIdentifierCreationNotification | 5.5.2.18 of ETSI GS  NFV-SOL 003 [[11]](#_bookmark13) | This type represents the notification about the creation of an identifier for the NF Deployment. |
| VnfIdentifierDeletionNotification | 5.5.2.19 of ETSI GS  NFV-SOL 003 [[11]](#_bookmark13) | This type represents the notification about the deletion of an identifier of the NF Deployment, and consequently of  the deletion of the resource representing the NF Deployment. |

#### Type: VnfLcmOperationOccurrenceNotification

This type represents the notification about an NF Deployment lifecycle management operation occurrence. It is used to inform the API consumer about changes in the lifecycle of the NF Deployment which are caused due to an occurrence of a respective lifecycle operation.

The type is defined in clause 5.5.2.17 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* Only the following trigger conditions apply:
  + Instantiation of the VNF, i.e., of the NF Deployment.
  + Termination of the VNF instance, i.e., of the NF Deployment instance.
  + Healing of the VNF instance, i.e., of the NF Deployment instance. This includes both "on-demand healing" and "auto-healing" (see clause 2.4.2.4).
  + Scaling of the VNF instance, i.e., of the NF Deployment instance. This includes both "on-demand scaling" and "auto-scaling" (see clause 2.4.2.5).
  + Changing external connectivity of the VNF instance, i.e., of the NF Deployment instance.
  + Changing of the current VNF package of the VNF instance, i.e., of the NF Deployment instance.
  + Changing of the VNF flavour of the VNF instance, i.e., of the NF Deployment instance.
* Sequence of notifications in relation to "granting" are not applicable

NOTE: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

* + - 1. Void
      2. Void

### Data model

#### REST resource data types

This clause specifies the data types used in resource representations and notifications supported by the present API. Table 3.2.6.1-1 lists the data types.

**Table 3.2.6.1-1: O2dms\_DeploymentLifecycle resource data types**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Clause defined** | **Description** |
| VnfInstance | 3.2.6.2.2 | This type represents an NF Deployment instance. |
| VnfLcmOpOcc | 3.2.6.2.6 | This type represents a lifecycle management operation occurrence of an NF Deployment. |
| LccnSubscription | 3.2.6.2.8 | This type represents a subscription related to notifications about NF Deployment lifecycle changes. |

#### Structured data types

* + - * 1. Introduction

This clause defines the structured data types to be used in resource representations.

All the structured data types specified in clause 5.5.2 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported, with the exception of those that are only used in REST resources and message contents of non-supported REST resources and methods, i.e., that are not listed in table 3.2.3-1. If specific provisions to structured data types are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.2.6.2 of the present document.

* + - * 1. Type: VnfInstance

The "VnfInstance" type represent an NF Deployment instance. The type is defined in clause 5.5.2.2 of ETSI GS NFV- SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* The following attributes need not be supported:
* The following links to resources need not be present even when the current status of the NF Deployment permits its presence:
  + "\_links.indicators",
  + "\_links.operate",
  + "\_links.createSnapshot",
  + "\_links.revertToSnaspshot".

All structure data types referenced (i.e., used as data type of attributes) in the "VnfInstance" shall be supported, with the following exceptions:

* Void.
  + - * 1. Void
        2. Void
        3. Void
        4. Type: VnfLcmOpOcc

This type represents a NF Deployment lifecycle management operation occurrence. The type is defined in clause

5.5.2.13 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* The following attributes are further qualified as follows:
  + "operationParams": supported objects content of the attribute shall correspond only to supported lifecycle management operations on the NF Deployment: INSTANTIATE, TERMINATE, SCALE, SCALE\_TO\_LEVEL, HEAL, CHANGE\_EXT\_CONN, CHANGE\_VNFPKG, CHANGE\_FLAVOUR.
* The following attributes need not be supported:
  + "grantId", which indicates the identifier of the grant related to the operation occurrence (see note 2).
  + “vnfSnapshotInfoId”,
  + “lcmCoordinations”,
  + “rejectedLcmCoordinations”,
* The following links to resources need not be present even when the current status of the NF Deployment permits its presence:
  + “\_links.grant”, and

- “\_links.vnfSnapshot”.

NOTE 1: Void.

NOTE 2: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

NOTE 3: Void.

* + - * 1. Void
        2. Void
        3. Type: HealVnfRequest

The type represents the request parameters for the "Heal NF Deployment" operation (referred as "Heal VNF" in ETSI GS NFV-SOL 003 [[11]](#_bookmark13)). The type is defined in clause 5.5.2.9 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause of ETSI GS NFV-SOL 003 [[11]](#_bookmark13):

* The DMS shall support the following additional attributes in the "HealVnfRequest" message content:
  + "vnfcInstanceId", with data type, cardinality and meaning as defined in clause 5.5.2.9 of ETSI GS NFV- SOL 002 [[29],](#_bookmark30) and
  + "healScript", with data type content, cardinality and meaning as defined in clause 5.5.2.9 of ETSI GS NFV-SOL 002 [[29].](#_bookmark30)

NOTE: The additional attributes from the ETSI GS NFV-SOL 002 [[29]](#_bookmark30) offer the capability to perform a "more granular" healing request, as well as further indications on the scripts or healing actions or set of rules for the healing procedure.

* + - * 1. Void
        2. Void
        3. Void
        4. Void
        5. Void
        6. Void
        7. Void

#### Referenced structured data types

* + - * 1. Introduction

This clause defines the structured data types that are referenced by other data types.

All the referenced structured data types specified in clause 5.5.2 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported, with the exception of those that are only used in REST resources and message contents of non-supported REST resources and methods, i.e., that are not listed in table 3.2.3-1. If specific provisions to referenced structured data types are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.2.6.3 of the present document. In addition, all common structured data types in clause 4.4.1 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported if referenced/used by other data types as indicated above.

* + - * 1. Type: VimConnectionInfo

This type represents parameters for a DMS to establish connectivity with underlying virtualised resource managers, containerized workload managers, container image repositories and MCIOP repositories. The structure is used to convey interface connectivity and credentials related parameters over the O2 interface. Additional parameters may be configured into the DMS by means outside the scope of the present document.

The type is defined in clause 4.4.1.6 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Void
        2. Type: VnfExtCpInfo

This type represents information about an external connection point of an NF Deployment. The type is defined in clause

5.5.3.17 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Void
        2. Void
        3. Type: ExtManagedVirtualLinkInfo

This type represents information about an externally-managed internal virtual link of an NF Deployment. The type is defined in clause 5.5.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

- The following attributes need not be supported:

- “extManagedMultisiteVirtualLinkId”, which indicates the identifier of the externally-managed multi-site virtual link instance in the case that NF Deployment can span more than one O-Cloud site and multi-site connectivity becomes thus necessary.

* + - * 1. Type: VnfcResourceInfo

This type represents information about a component of an NF Deployment and the used virtualised compute and storage resources for VM-based deployment or Compute MCIO and Storage MCIO for OS container-based deployment. The type is defined in clause 5.5.3.5 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

- The following attributes need not be supported:

- "reservationId".

* + - * 1. Type: VnfVirtualLinkResourceInfo

This type represents information about an internal virtual link instance of an NF Deployment and the associated virtualised network resource or Network MCIO. The type is defined in clause 5.5.3.6 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

- The following attributes need not be supported:

- "reservationId".

* + - * 1. Type: VirtualStorageResourceInfo

This type represents information about storage of an NF Deployment and the associated virtualised storage resource or Storage MCIO. The type is defined in clause 5.5.3.7 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

- The following attributes need not be supported:

- "reservationId".

* + - * 1. Void
        2. Void
        3. Void
        4. Void
        5. Type: AffectedVnfc

This type represents information about added, deleted, modified or temporary components of an NF Deployment. The type is defined in clause 5.5.3.13 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

- The following attributes need not be supported:

- "resourceDefinitionId", which identifies resource definitions exchanged via the granting (see note 1).

NOTE 1: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

NOTE 2: Void.

* + - * 1. Type: AffectedVirtualLink

This type represents information about added, deleted, modified or temporary internal virtual links of an NF Deployment. The type is defined in clause 5.5.3.14 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

- The following attributes need not be supported:

- "resourceDefinitionId", which identifies resource definitions exchanged via the granting (see note 1).

NOTE 1: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

NOTE 2: Void.

* + - * 1. Type: AffectedExtLinkPort

This type represents information about added, deleted, modified external link ports attached to external virtual links to an NF Deployment. The type is defined in clause 5.5.3.14a of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

- The following attributes need not be supported:

- "resourceDefinitionId", which identifies resource definitions exchanged via the granting (see note).

NOTE: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

* + - * 1. Type: AffectedVipCp

This type represents information about added, deleted, modified VIP connection points of an NF Deployment. The type is defined in clause 5.5.3.14b of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Type: AffectedVirtualStorage

This type represents information about added, deleted, modified or temporary virtualised storage resources of an NF Deployment. The type is defined in clause 5.5.3.15 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

- The following attributes need not be supported:

- "resourceDefinitionId", which identifies resource definitions exchanged via the granting (see note 1).

NOTE 1: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

NOTE 2: Void.

* + - * 1. Type: ResourceHandle

This type represents information addressing a resource that is used by an NF Deployment instance. Information about the resource is available from underlying O-Cloud infrastructure resource managers part of the DMS.

The type is defined in clause 4.4.1.7 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* The following attributes are further qualified as follows:
  + “vimConnectionId”, the attribute shall be supported since only direct-mode for resource management is supported, i.e., the part of the DMS exposing the present O2dms\_LifecycleDeployment interacts directly with the functions responsible for the infrastructure resource provisioning.
* The following attributes need not be present even when the conditions permit its presence:
  + "resourceProviderId".
    - * 1. Void
        2. Type: ExtVirtualLinkData

This type represents an external virtual link of an NF Deployment. The type is defined in clause 4.4.1.11 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* The following attributes are further qualified as follows:
  + "vimConnectionId", the attribute shall be supported since only direct-mode for resource management is supported, i.e., the part of the DMS exposing the present O2dms\_LifecycleDeployment interacts directly with the functions responsible for the infrastructure resource provisioning.
* The following attributes need not be present even when the conditions permit its presence:
  + "resourceProviderId".
    - * 1. Void
        2. Type: VnfExtCpConfig

This type represents network address information per instance of an external connection point of the NF Deployment.

In the case of VM-based NF Deployment, the DMS will have to either use existing link ports to the virtualised networks enabled for the external connectivity or create the link ports itself.

In the case of OS container-based deployment of a component of the NF Deployment exposing the external connection point, the DMS will have to use the network attachment definition resource of secondary container cluster network when connecting the connection point to the external VL.

In all cases, including the case of using a Virtual CP as an external connection point of the NF Deployment, the "cpProtocolData" contained in the "VnfExtCpConfig" supports providing configuration of the network protocols (e.g., IP address data).

The type is defined in clause 4.4.1.10a of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Void
        2. Void
        3. Type: ExtManagedVirtualLinkData

This type represents an externally-managed internal virtual link of an NF Deployment. The type is defined in clause

4.4.1.12 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* The following attributes are further qualified as follows:
  + "vimConnectionId", the attribute shall be supported since only direct-mode for resource management is supported, i.e., the part of the DMS exposing the present O2dms\_LifecycleDeployment interacts directly with the functions responsible for the infrastructure resource provisioning.
* The following attributes need not be supported:
  + "extManagedMultisiteVirtualLinkId", which indicates an externally-managed multi-site virtual link instance (see note).
* The following attributes need not be present even when the conditions permit its presence:
* "resourceProviderId".
  + - * 1. Type: ExtLinkPortData

This type represents an externally provided link port to be used to connect an external connection point of the NF Deployment to an external virtual link. This type is used in case of VM-based NF Deployments which use ports to connect to the virtual networks. The type is defined in clause 4.4.1.14 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Type: VnfLinkPortData

This type represents an externally provided link port to be used to connect a connection point of a constituent of the NF Deployment to an internal virtual link of the NF Deployment that is managed externally. This type is used in case of VM-based NF Deployments which use ports to connect to the virtual networks. The type is defined in clause 5.5.3.18 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Type: VirtualCpAddressData

This type represents network address data for a virtual CP. The type is defined in clause 4.4.1.10d of ETSI GS NFV- SOL 003 [[11].](#_bookmark13)

Refer to the provisions specified in clause 4.4.1.10d of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) regarding the handling of providing IP address data for the external load balancer realizing the virtual CP, such as the case when no IP address is provided, and it is assigned by the container cluster.

* + - * 1. Void
        2. Void
        3. Type: McioInfo

This type represents information about an MCIO representing the set of component instances of an NF Deployment that are realized by one or a set of OS containers and that have been created based on the same declarative descriptor. The type is defined in clause 5.5.3.24 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Void
        2. Void
        3. Void
        4. Type: NetAttDefResourceData

This type represents a network attachment definition resource that specifies the interface to be used to connect one or multiple connections points to a secondary container cluster network that realizes a VL. The type is defined in clause

4.4.1.17 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Type: NetAttDefResourceInfo

This type represents the information related to a network attachment definition resource providing the information about an interface that is used to connect one or multiple connections points to a secondary container cluster network. The type is defined in clause 5.5.3.28 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Type: AdditionalResourceInfo

This type represents additional resource information that is resource and resource type specific and that is made available by the underlying infrastructure resource managers of the DMS. The type is defined in clause 4.4.1.16 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Type: IntVnfCpData

This type represents input data related to one or more internal connection points of the NF Deployment created based on the same declarative descriptor. The type is defined in clause 4.4.1.18 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

NOTE: The referenced ETSI GS NFV-SOL 003 [[11]](#_bookmark13) only supports referencing to internal connection points of the NF Deployment when attached to secondary container cluster networks, i.e., only applicable for connectivity of containerized workloads of an NF Deployment.

* + - * 1. Void

#### Simple data types and enumerations

* + - * 1. Introduction

The present clause defines simple data types and enumerations that can be referenced from data structures defined in other clauses. The simple data types defined in a request body, response body, or a structure type are defined in ETSI GS NFV-SOL 013 [[15]](#_bookmark17) or the present clause of the specification.

All the simple data types and enumerations specified in clause 5.5.4 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported with the exception of those that are only used in REST resources and message contents of non-supported REST resources and methods, i.e., that are not listed in table 3.2.3-1. If specific provisions to simple data types and enumerations are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.3.6.4 of the present document. In addition, all common simple data types and enumerations in clause 4.4.2

4.4.1 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported if referenced/used by other data types as indicated above.

* + - * 1. Simple data types

The simple data types defined in table 3.2.6.4.2-1 shall be supported.

**Table 3.2.6.4.2-1: Simple data types**

|  |  |  |
| --- | --- | --- |
| **Type name** | **Type definition** | **Description** |
| IdentifierInVnfd | N/A | An identifier that is unique within a deployment descriptor of an NF Deployment. Representation: string of variable length. See note. |
| IdentifierInVim | N/A | An identifier maintained by the functions responsible for the infrastructure resource provisioning in the DMS. It is expected to be unique within a function responsible for the infrastructure  resource provisioning. Representation: string of variable length. See note. |
| IdentifierInVnf | N/A | An identifier that is unique for the respective type within an NF Deployment instance, but that need not be globally unique.  Representation: string of variable length. See note. |
| NOTE: See also definition in clause 4.4.2.2 of ETS GS NFV-SOL 003 [[11].](#_bookmark13) | | |

* + - * 1. Void
        2. Void
        3. Void
        4. Enumeration: LcmOperationType

The enumeration is defined in clause 5.5.4.7 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

- Only the following values shall be supported: INSTANTIATE, TERMINATE, SCALE, SCALE\_TO\_LEVEL, HEAL, CHANGE\_EXT\_CONN, CHANGE\_VNFPKG, and CHANGE\_FLAVOUR, as defined in table 5.5.4.7-1 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

* + - * 1. Void
        2. Void
        3. Enumeration: CancelModeType

The enumeration is defined in clause 5.5.4.6 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following specific DMS applicability conditions are considered with respect to the referenced specification clause:

- GRACEFUL: In the case the lifecycle management operation occurrence is in "STARTING" state, the DMS need not wait for a granting request

NOTE: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

#### Other data types

* + - * 1. Introduction

The present clause defines data types that are not referenced explicitly by the O2dms\_LifecycleDeployment API but that are necessary for representing additional information to be considered by the orchestration procedures between the SMO and the DMS.

* + - * 1. Type: ResourceDefinition

This type represents information about a resource that is expected to be used or is already associated to an NF Deployment. The type is defined in clause 9.5.3.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

NOTE 1: The present type is not referenced by any of the other types and resource representations of the O2dms\_LifecycleDeployment API.

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* The following attributes are further qualified as follows:
  + "id", the identifier shall be unique within the scope of an NF Deployment and need not be unique within the scope of the referred "GrantRequest" structure.
* The following attributes need not be present even when the conditions permit its presence:
  + "snapshotResDef".

NOTE 2: Void.

* + - * 1. Type: ResourcePlacement

This type represents information about the placement information and constraints for a resource that is expected to be used or is already associated to an NF Deployment.

The data type shall comply with the provisions defined in table 3.2.6.5.3-1.

**Table 3.2.6.5.3-1: Definition of ResourcePlacement data type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Cardinality** | **Description** |

|  |  |  |  |
| --- | --- | --- | --- |
| resourceDefinitionId | Identifier | 1 | Identifier of the related "ResourceDefinition" structure defining the resource to be managed. |
| vimConnectionId | Identifier | 0..1 | Identifier of the connection to the functions responsible for the infrastructure resource provisioning. Shall be present for new resources and shall be absent for resources that have already been allocated.  The applicable "VimConnectionInfo" structure, which is referenced by this attribute, can be obtained from the "vimConnectionInfo" attribute of the "VnfInstance" structure. |
| zoneId | Identifier | 0..1 | Identifier of the resource zone, as managed by the O-Cloud resource management functions, into which the resource is to be placed. Shall be present for new resources and if the resource zones abstraction is applicable to them (e.g., for compute related resources). Shall be absent for resources that  have already been allocated. |
| resourceGroupId | IdentifierInVim | 0..1 | Identifier of the "infrastructure resource group" to be provided when allocating the resource. The "infrastructure resource group" represents a logical grouping of O-Cloud resources assigned to a tenant. Shall be present for new resources.  Shall be absent for resources that have already been allocated. |
| containerNamespace | String | 0..1 | The value of the namespace in which the containerized workloads of an NF Deployment shall be deployed. Shall be present if the underlying resource are managed by functions responsible for the infrastructure resource provisioning  equivalent to a CISM (e.g., a Kubernetes® cluster). |
| mcioConstraints | KeyValuePair | 0..N | The constraint values to be assigned to containerized workloads of an NF Deployment with containerized components.  The key in the key-value pair indicates the parameter name of the constraint in the declarative descriptor of the containerized workload and shall be one of the possible enumeration values of the "mcioConstraintsParams" attribute as specified in clause  7.1.6.2.2 of ETSI GS NFV-IFA 011 [[12].](#_bookmark14) The value in the key- value pair indicates the value to be assigned to the constraint. Shall be present if the underlying resource are managed by functions responsible for the infrastructure resource  provisioning equivalent to a CISM (e.g., a Kubernetes® cluster). |

* + - * 1. Type: ResourcePlacementConstraint

This type represents information about the placement constraints between two or more resources of an NF Deployment. The data type shall comply with the provisions defined in table 3.2.6.5.4-1.

**Table 3.2.6.5.4-1: Definition of ResourcePlacementConstraint data type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Cardinality** | **Description** |
| affinityOrAntiAffinity | Enum (inlined) | 1 | The type of constraint.  Permitted values:   * AFFINITY * ANTI\_AFFINITY |
| scope | Enum (inlined) | 1 | The scope of the resource placement constraint indicating the category of the scope where the constraint applies.  Permitted values:   * NFVI\_POP: it represents the scope of a O-Cloud Site. * ZONE: it represents the scope of a resource zone in an O- Cloud Site. * NFVI\_NODE: it represents the scope of an O-Cloud Node for VM-based deployments. * CIS\_NODE: it represents the scope of a CIS cluster node, i.e., an O-Cloud Node in an O-Cloud Node Cluster for container- based deployments. * CONTAINER\_NAMESPACE: it represents the scope of a container namespace. |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | See note. |
| resource | Structure (inlined) | 2..N | The resources that are involved in the constraint rule. |
| >idType | Enum (inlined) | 1 | The type of the identifier.  Permitted values:   * RES\_MGMT: it represents a resource management level identifier, i.e., an identifier as known/exposed by the functions responsible for the infrastructure management provisioning. * RES\_DEFINITION: it represents an identifier of the "ResourceDefinition" structure. |
| >resourceId | Identifier | 1 | The value of the resource identifier within the scope indicated by the "idType" attribute. |
| NOTE: The "CIS\_NODE" and "CONTAINER\_NAMESPACE" scopes shall only be used to express affinity or anti- affinity relationship between containerized workloads. | | | |

* + - * 1. Type: CloudAssets

This type represents information about O-Cloud assets that can be used or be associated to an NF Deployment. The data type shall comply with the provisions defined in table 3.2.6.5.5-1.

**Table 3.2.6.5.5-1: Definition of CloudAssets data type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Cardinality** | **Description** |
| computeResourceFlavours | VimComputeResourceFlavour | 0..N | Mappings between virtual compute descriptors defined in the NF Deployment descriptor and compute resource flavours managed in the O-Cloud functions responsible for virtualised infrastructure  management. |
| softwareImages | VimSoftwareImage | 0..N | Mappings between software images defined in the NF Deployment descriptor and software images managed in the O-Cloud functions responsible for serving as software  image repository. |
| storageAssets | StorageAsset | 0..N | Mappings between virtual storages defined in the NF Deployment descriptor and virtual  storages managed in the O-Cloud. |
| paasAssets | PaasAsset | 0..N | Mappings between PaaS Services requests defined in the NF Deployment descriptor and PaaS Services managed in the O-Cloud. |

* + - * 1. Type: VimComputeResourceFlavour

This type defines the mapping between a virtual compute descriptor and the corresponding compute resource flavour managed by the O-Cloud functions responsible for virtualised infrastructure management.

The type is defined in clause 9.5.3.9 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* The following attributes are further qualified as follows:
  + "vimConnectionId", the attribute shall be supported since only direct-mode for resource management is supported, i.e., the part of the DMS exposing the present O2dms\_LifecycleDeployment service API interacts directly with the functions responsible for the infrastructure resource provisioning.
* The following attributes need not be present even when the conditions permit its presence:
  + "resourceProviderId".
    - * 1. Type: VimSoftwareImage

This type defines the mapping between a software image definition in the descriptor of the NF Deployment and the corresponding software image managed by the O-Cloud functions performing the serving as image repository and managing the software images.

The type is defined in clause 9.5.3.10 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* The following attributes are further qualified as follows:
  + "vimConnectionId", the attribute shall be supported since only direct-mode for resource management is supported, i.e., the part of the DMS exposing the present O2dms\_LifecycleDeployment service API interacts directly with the functions responsible for the infrastructure resource provisioning.
* The following attributes need not be present even when the conditions permit its presence:
  + "resourceProviderId".
    - * 1. Type: StorageAsset

This type defines the mapping between a virtual storage description in the NF Deployment descriptor and the corresponding virtual storage managed in the O-Cloud.

The type is defined in clause 9.5.3.13 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following DMS applicability conditions are considered with respect to the referenced specification clause:

* The following attributes are further qualified as follows:
  + "vimConnectionId", the attribute shall be supported since only direct-mode for resource management is supported, i.e., the part of the DMS exposing the present O2dms\_LifecycleDeployment service API interacts directly with the functions responsible for the infrastructure resource provisioning.
* The following attributes need not be present even when the conditions permit its presence:
  + "resourceProviderId".
    - * 1. Type: PaasAsset

This type defines the mapping between a PaaS service request in the NF Deployment descriptor (i.e., the VNFD) and the assigned PaaS Service provided by the O-Cloud.

The type is defined as a structure inlined in clause 9.5.2.3 of ETSI GS NFV-SOL 003 [[11],](#_bookmark13) and also represented in the following table 3.2.6.5.9-1.

**Table 3.2.6.5.9-1: Definition of PaasAsset data type**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Cardinality** | **Description** |
| paasServiceType | String | 1 | The type of PaaS Service. The value of this attribute is expected to be matched against values of the registered PaaS Services in the PaaS Services  Repository function. |
| paasServiceId | Identifier | 1 | Identifier of the assigned PaaS Service as managed by the PaaS Services Management function. |
| paasServiceVersion | Version | 0..1 | Version of the PaaS Service. It shall be present if the PaaS Service is versioned. |
| paasServiceRequestId | IdentifierInVnfd | 1 | Identifier of the PaaS Service request in the Deployment Descriptor that maps to the assigned PaaS Service. |
| paasServiceHandle | PaasServiceHandle | 1 | A handle enabling the access and use of the PaaS Service by the NF Deployment. |

### Error handling

#### General

For the O2dms\_DeploymentLifecycle Service API, HTTP error responses shall be supported as specified in clause 3.1.5.

#### Error handling in NF Deployment lifecycle procedures

* + - * 1. Introduction

Clause 5.6.1 in ETSI GS NFV-SOL 003 [[11]](#_bookmark13) describes basic concepts for error handling to be considered during lifecycle management operation occurrences. The same clause also describes what are common failure resolution strategies and the concepts of retry and rollback of lifecycle management operations. In the descriptions, references to the NFV-MANO functional blocks are provided (e.g., NFVO and VNFM), while in the present profile document, the corresponding O-RAN framework elements are to be considered instead according to the mappings in clause 2.3.2 and the specification in clause 3.2.4.1.

* + - * 1. States and state transitions of lifecycle management operation occurrences of NF Deployments

The state and state transition of lifecycle management operation occurrences of NF Deployments shall be supported by the DMS according to the provisions in clause 5.6.2 of ETSI GS NFV-SOL 003 [[11],](#_bookmark13) with the following additional DMS applicability conditions with respect to the referenced specification clause:

- The provisions and descriptions related to "obtaining the grant" need not be supported.

NOTE: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

Based on the above, specific states as supported by the present document version have the following characteristics.

##### about states:

* + STARTING:
    - The VNF lifecycle management operation occurrence of the NF Deployment shall transit into the PROCESSING state once the DMS has checked that a) resources management fulfillment information (without performing a granting exchange), b) placement information, and c) deployment items of the NF Deployment are available to the DMS. The items a), b) and c) in the previous listing are hereafter referred as "NF Deployment lifecycle items"
    - If an error occurs during the checking steps, or the DMS does not have the necessary information to proceed with the infrastructure resources fulfillment for the NF Deployment, the lifecycle management operation occurrence of the NF Deployment shall transit into the "ROLLED\_BACK" state.
  + FAILED\_TEMP:
    - If "NF Deployment lifecycle items" for the lifecycle operation were available, these are still valid.

##### about error handling operations:

* + Retry:
    - The "Retry" shall operate within the bounds of the "NF Deployment lifecycle items" information available to the DMS.
  + Rollback:
    - The "Rollback" shall operate within the bounds of the "NF Deployment lifecycle items" information available to the DMS.
      * 1. Specific cases of error handling

Immediate failure

The description and provisions for the DMS to handle an immediate failure is provided in clause 5.6.3.1 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

Failure in "STARTING" state

The description and provisions for the DMS to handle a failure in "STARTING" state is provided in clause 5.6.3.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The following additional DMS applicability conditions with respect to the referenced specification clause apply:

The provisions and descriptions related to "obtaining the grant" need not be supported, so if the operation fails before the DMS has the necessary "NF Deployment lifecycle items" information, no persistent change to the state of the NF Deployment has happened.

NOTE: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

Failure during the LCM operation execution

The description and provisions for the DMS to handle during the execution of the LCM operation is provided in clause

5.6.3.3 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The following additional DMS applicability conditions with respect to the referenced specification clause apply:

- The provisions and descriptions related to "obtaining the grant" need not be supported.

NOTE: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

Cancellation of LCM operation execution

The description and provisions for the DMS to handle during the cancellation of an LCM operation is provided in clause 5.6.3.4 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) The following additional DMS applicability conditions with respect to the referenced specification clause apply:

The provisions and descriptions related to "obtaining the grant" need not be supported.

NOTE: Granting exchanges between DMS and SMO are not supported in the present document version. Refer also to the conditions described in clause 3.2.4.1.

#### Flows of erroring handling lifecycle management operation occurrences of NF Deployments

* + - * 1. Overview

Clause 3.2.7.b defines and references to flows about error handling of lifecycle management operation occurrences of NF Deployments. In the description, references to already documented flows in ETSI GS NFV-SOL 003 [[11]](#_bookmark13) is provided, with the consideration of the mapping meanings of NFVO and VNFM to the O-RAN framework as specified in clause 3.2.4.1 of the present document.

* + - * 1. Flow of retrying a lifecycle management operation occurrence of an NF Deployment

Clause 5.3.10 in ETSI GS NFV-SOL 003 [[11]](#_bookmark13) describes the flow of retrying a VNF lifecycle management operation, which is also applicable in the case of lifecycle management operations on an NF Deployment.

* + - * 1. Flow of rolling back a lifecycle management operation occurrence of an NF Deployment

Clause 5.3.11 in ETSI GS NFV-SOL 003 [[11]](#_bookmark13) describes the flow of rolling back a VNF lifecycle management operation, which is also applicable in the case of lifecycle management operations on an NF Deployment.

* + - * 1. Flow of failing a lifecycle management operation occurrence of an NF Deployment

Clause 5.3.12 in ETSI GS NFV-SOL 003 [[11]](#_bookmark13) describes the flow of failing a VNF lifecycle management operation, which is also applicable in the case of lifecycle management operations on an NF Deployment.

* + - * 1. Flow of cancelling a lifecycle management operation occurrence of an NF Deployment

Clause 5.3.13 in ETSI GS NFV-SOL 003 [[11]](#_bookmark13) describes the flow of cancelling a VNF lifecycle management operation, which is also applicable in the case of lifecycle management operations on an NF Deployment.

## O2dms\_DeploymentFault Service API

### Description

This service interface enables an API consumer (e.g., the SMO) to invoke fault management operations of NF Deployments towards the DMS. The service interface also enables the management of subscriptions to notifications raised by the DMS regarding the fault management of the NF Deployments.

### API version

For the O2dms\_DeploymentFault Service API version as specified in the present document, the MAJOR version field shall be "1", the MINOR version field shall be "12", and the PATCH version field shall be "0" as specified in clause

7.1a of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) Consequently, the {apiMajorVersion} URI variable shall be set to "v1".

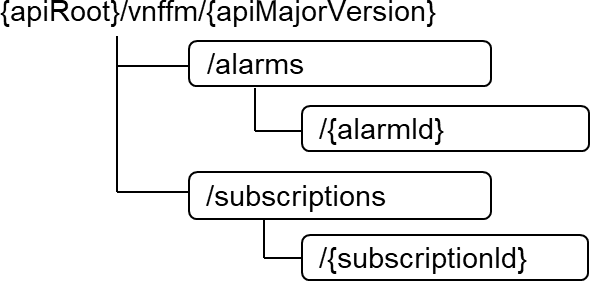
### REST resources structure and methods

All resource URIs of the API shall use the base URI specified defined in clause 3.1.2. The string "vnffm" shall be used to represent {apiName} as defined in clause 7.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) All resource URIs in clauses below are defined relative to the formed based URI (i.e., {apiRoot}/xyz/{apiVersion}).

When ambiguity is possible, the term REST resource is used to make the distinction between the term resource as understood within the context of REST API from the term resource as understood within the context of O-RAN.

The resource URI structure defined for the O2dms\_DeploymentFault Service API shall be as specified in clause 7.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) Applicable individual REST resources for the present version of the O2dms\_DeploymentFault Service API are listed in table 3.3.3-1.

Figure 3.3.3-1 illustrates the overall resource URI structure defined for the O2dms\_DeploymentFault Service API.



##### Figure 3.3.3-1: Resource URI structure of the O2dms\_DeploymentFault Service API

Table 3.3.3-1 lists the individual REST resources defined and the applicable HTTP methods. The DMS shall support responding to requests for all HTTP methods on the resources in table 3.3.3-1 that are marked as "M" (mandatory) in the "Cat" column.

**Table 3.3.3-1: Resources and methods overview**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Resource name** | **Resource URI** | **HTTP**  **method** | **Cat** | **Description (see note 2)** |
| Alarms | /alarms  See note 1. | GET | M | Query multiple alarms related to NF Deployments. |
| Individual alarm | /alarms/{alarmId} See note 1. | GET | M | Read an individual alarm related to an NF Deployment. |
| PATCH | M | Modify information of an individual alarm related to an NF Deployment. |
| Subscriptions | /subscriptions See note 1. | POST | M | Subscribe to fault management  notifications related to NF Deployments. |
| GET | M | Query information about multiple subscriptions to NF Deployment fault management notifications. |
| Individual subscription | /subscriptions/{subscriptionId} See note 1. | GET | M | Read an "Individual subscription" resource which represents information about a subscription to NF Deployment  fault management notifications. |
| DELETE | M | Terminate a subscription to NF Deployment fault management  notifications. |
| Notification endpoint | (provided by API consumer) See note 1. | POST | See note  3 | Notify about NF Deployment fault management events. |
| GET | See note 3 | Test the notification endpoint for NF Deployment fault management events. |
| NOTE 1: The resource URI value is the same as the one specified in ETSI GS NFV-SOL 003 [[11]](#_bookmark13).  NOTE 2: The specified meaning of the resource name and supported HTTP method in ETSI GS NFV-SOL 003 [[11]](#_bookmark13) applies, unless otherwise stated. Additional description can be presented in case of further refining its meaning in the context of O2dms specification.  NOTE 3: The DMS shall support invoking the HTTP methods defined for the "Notification endpoint" resource exposed by the API consumer. If the API consumer supports invoking the POST method on the "Subscriptions" resource towards the DMS, it shall also support responding to the HTTP requests defined for the  "Notification endpoint" resource. | | | | |

### REST resources

#### General

The DMS and API consumer of the O2dms\_DeploymentFault Service API shall support all the provisions specified in clause 7.4 of ETSI GS NFV-SOL 003 [[11],](#_bookmark13) with the following considerations:

* NFVO and VNFM entities referred in the ETSI GS NFV-SOL 003 [[11]](#_bookmark13) are mapped to SMO and DMS as defined in clause 2.3.2 of the present document. Concerning to the managed object types to which alarms can be associated/related to, VNF instance is mapped to NF Deployment, as also defined in clause 2.3.2 of the present document.
* Only the provisions associated to corresponding REST resources and methods listed in table 3.3.3-1 are applicable for the present document. That is, if a REST resource and method is not listed in table 3.3.3-1 of the present document, the corresponding clause in the ETSI GS NFV-SOL 003 [[11]](#_bookmark13) that specifies such REST resource, and its supported methods, is not required to be supported according to the present document version.

If specific provisions to REST resources are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.3.4 of the present document.

* For the list of referenced data types for notifications to support, these are defined in clause 3.3.5 of the present document.
* For the list of REST resource data types, structured data types and referenced simple types to be supported, refer to clauses 3.3.6.1, 3.3.6.2, 3.3.6.3 and 3.3.6.4 of the present document.

### Notifications

#### General

Notifications issued on the O2dms\_DeploymentFault Service API shall comply with the Subscribe/Notify pattern as described in clause 5.9 of ETSI GS NFV-SOL 015 [17].

Table 3.3.5.1-1 provides an overview of the notifications supported by the present API.

**Table 3.3.5.1-1: O2dms\_DeploymentFault notifications overview**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Clause defined** | **Description** |
| AlarmNotification | 7.5.2.5 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) | This type represents an alarm notification about NF Deployment faults. |
| AlarmClearedNotification | 7.5.2.6 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) | This type represents an alarm cleared notification about NF Deployment faults. |
| AlarmListRebuiltNotification | 7.5.2.7 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) | This type represents a notification that the alarm list related to NF Deployments has been rebuilt. |

### Data model

#### REST resource data types

This clause specifies the data types used in resource representations and that shall be supported by the present API. Table 3.3.6.1-1 lists the data types.

**Table 3.3.6.1-1: O2dms\_DeploymentFault resource data types**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Clause defined** | **Description** |
| Alarm | 3.3.6.2.2 | This type represents an alarm associated to an NF Deployment. |
| FmSubscription | 7.5.2.3 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) | This type represents a subscription related to notifications related to fault management of NF Deployments. |

#### Structured data types

* + - * 1. Introduction

This clause defines the structured data types to be used in resource representations.

All the structured data types specified in clause 7.5.2 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported. If specific provisions to structured data types are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.3.6.2 of the present document.

* + - * 1. Type: Alarm

The "Alarm" type encapsulates the information about an alarm associated to an NF Deployment instance. The type is defined in clause 7.5.2.4 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13)

In addition, the following O-Cloud applicability conditions and clarifications are considered with respect to the referenced specification clause:

* For attributes name descriptions referring to NFVI, in the context of the present profiling specification, NFVI concerns to O-Cloud.

#### Referenced structured data types

* + - * 1. Introduction

This clause defines the structured data types that are referenced by other data types.

All the referenced structured data types specified in clause 7.5.3 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported. If specific provisions to referenced structured data types are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.3.6.3 of the present document. In addition, all common structured data types in clause 4.4.1 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported if referenced/used by other data types as indicated above.

#### Simple data types and enumerations

* + - * 1. Introduction

The present clause defines simple data types and enumerations that can be referenced from data structures defined in other clauses. The simple data types defined in a request body, response body, or a structure type are defined in ETSI GS NFV-SOL 013 [[15]](#_bookmark17) or the present clause of the specification.

All the simple data types and enumerations specified in clause 7.5.4 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported. If specific provisions to simple data types and enumerations are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.3.6.4 of the present document. In addition, all common simple data types and enumerations in clause 4.4.2 4.4.1 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported if referenced/used by other data types as indicated above.

#### Other data types

* + - * 1. Introduction

The present clause defines other data types necessary for representing information to be considered by the orchestration procedures between the SMO and the DMS.

No particular other data type is defined currently for the present API.

## O2dms\_DeploymentPerformance Service API

### Description

This service interface enables an API consumer (e.g., the SMO) to invoke performance management operations of NF Deployments towards the DMS. The service interface also enables the management of subscriptions to notifications raised by the DMS regarding the performance management of the NF Deployments via the management of PM jobs and Threshold.

### API version

For the O2dms\_DeploymentPerformance Service API version as specified in the present document, the MAJOR version field shall be "2", the MINOR version field shall be "12", and the PATCH version field shall be "0" as specified in clause 6.1a of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) Consequently, the {apiMajorVersion} URI variable shall be set to "v2".

### REST resources structure and methods

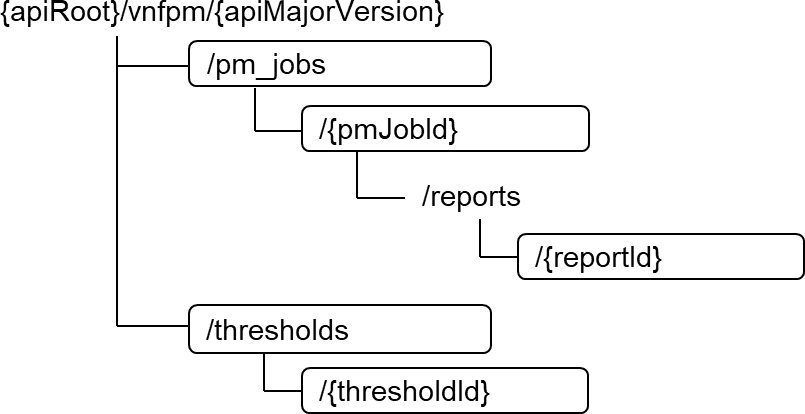
All resource URIs of the API shall use the base URI specified defined in clause 3.1.2. The string "vnfpm" shall be used to represent {apiName} as defined in clause 6.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) All resource URIs in clauses below are defined relative to the formed based URI (i.e., {apiRoot}/xyz/{apiVersion}).

When ambiguity is possible, the term REST resource is used to make the distinction between the term resource as understood within the context of REST API from the term resource as understood within the context of the O-Cloud.

The resource URI structure defined for the O2dms\_DeploymentPerformance Service API shall be as specified in clause

6.2 of ETSI GS NFV-SOL 003 [[11].](#_bookmark13) Applicable individual REST resources for the present version of the O2dms\_DeploymentPerformance Service API are listed in table 3.4.3-1.

Figure 3.4.3-1 illustrates the overall resource URI structure defined for the O2dms\_DeploymentPerformance Service API.



##### Figure 3.4.3-1: Resource URI structure of the O2dms\_DeploymentPerformance Service API

Table 3.4.3-1 lists the individual REST resources defined and the applicable HTTP methods. The DMS shall support responding to requests for all HTTP methods on the resources in table 3.4.3-1 that are marked as "M" (mandatory) in the "Cat" column.

**Table 3.4.3-1: Resources and methods overview**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Resource name** | **Resource URI** | **HTTP**  **method** | **Cat** | **Description (see note 2)** |
| PM jobs | /pm\_jobs  See note 1. | POST | M | Create a PM job related to NF Deployments. |
| GET | M | Query multiple PM jobs related to NF Deployments. |
| Individual PM job | /pm\_jobs/{pmJobId} See note 1. | GET | M | Read an individual PM job related NF Deployments. |
| PATCH | M | Update an individual PM job related to NF Deployments. |
| DELETE | M | Delete an individual PM job related to NF Deployments. |
| Individual  performance report | /pm\_jobs/{pmJobId}/reports/{reportId}  See note 1. | GET | M | Read an individual  performance report related to NF Deployments. |
| Thresholds | /thresholds  See note 1. | POST | M | Create a threshold related to NF Deployments. |
| GET | M | Query multiple thresholds related to NF Deployments. |
| Individual threshold | /thresholds/{thresholdId} See note 1. | GET | M | Read an individual threshold related NF Deployments. |
| PATCH | M | Update an individual threshold related to NF Deployments. |
| DELETE | M | Delete an individual threshold related to NF Deployments. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Notification endpoint | (provided by API consumer) See note 1. | POST | See  note 3 | Notify about NF Deployment  performance management events. |
| GET | See note 3 | Test the notification endpoint for NF Deployment performance management  events. |
| NOTE 1: The resource URI value is the same as the one specified in ETSI GS NFV-SOL 003 [[11]](#_bookmark13).  NOTE 2: The specified meaning of the resource name and supported HTTP method in ETSI GS NFV-SOL 003 [[11]](#_bookmark13) applies, unless otherwise stated. Additional description can be presented in case of further refining its meaning in the context of O2dms specification.  NOTE 3: The DMS shall support invoking the HTTP methods defined for the "Notification endpoint" resource exposed by the API consumer. If the API consumer supports invoking the POST method on the "PM jobs" or  "Thresholds" resource towards the DMS, it shall also support responding to the HTTP requests defined for the "Notification endpoint" resource. | | | | |

### REST resources

#### General

The DMS and API consumer of the O2dms\_DeploymentPerformance Service API shall support all the provisions specified in clause 6.4 of ETSI GS NFV-SOL 003 [[11],](#_bookmark13) with the following considerations:

* NFVO and VNFM entities referred in the ETSI GS NFV-SOL 003 [[11]](#_bookmark13) are mapped to SMO and DMS as defined in clause 2.3.2 of the present document. Concerning to the managed object types to which performance monitoring elements (e.g., PM jobs and thresholds) can be associated/related to, VNF instance is mapped to NF Deployment, as also defined in clause 2.3.2 of the present document.
* Only the provisions associated to corresponding REST resources and methods listed in table 3.4.3-1 are applicable for the present document. That is, if a REST resource and method is not listed in table 3.4.3-1 of the present document, the corresponding clause in the ETSI GS NFV-SOL 003 [[11]](#_bookmark13) that specifies such REST resource, and its supported methods, is not required to be supported according to the present document version. If specific provisions to REST resources are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.4.4 of the present document.
* For the list of referenced data types for notifications to support, these are defined in clause 3.4.5 of the present document.
* For the list of REST resource data types, structured data types and referenced simple types to be supported, refer to clauses 3.4.6.1, 3.4.6.2, 3.4.6.3 and 3.4.6.4 of the present document.

### Notifications

#### General

Notifications issued on the O2dms\_DeploymentPerformance Service API shall comply with the Subscribe/Notify pattern as described in clause 5.9 of ETSI GS NFV-SOL 015 [17].

Table 3.4.5.1-1 provides an overview of the notifications supported by the present API.

**Table 3.4.5.1-1: O2dms\_DeploymentPerformance notifications overview**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Clause defined** | **Description** |
| ThresholdCrossedNotification | 6.5.2.4 of ETSI GS NFV-SOL  003 [[11]](#_bookmark13) | This type represents a notification of a crossed threshold associated to an NF Deployment. |
| PerformanceInformationAvailableNotification | 6.5.2.5 of ETSI GS | This type represents a notification indicating that the performance information (a performance report) is available. |

|  |  |  |
| --- | --- | --- |
|  | NFV-SOL 003 [[11]](#_bookmark13) |  |

### Data model

#### REST resource data types

This clause specifies the data types used in resource representations and that shall be supported by the present API. Table 3.4.6.1-1 lists the data types.

**Table 3.4.6.1-1: O2dms\_DeploymentFault resource data types**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Clause defined** | **Description** |
| PmJob | 6.5.2.7 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) | This type represents a PM job associated to NF Deployments. |
| Threshold | 6.5.2.9 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) | This type represents a threshold. |
| PerformanceReport | 6.5.2.10 of ETSI  GS NFV-SOL 003  [[11]](#_bookmark13) | This type represents a performance report of measurements collected as part of a PM job. |

#### Structured data types

* + - * 1. Introduction

This clause defines the structured data types to be used in resource representations.

All the structured data types specified in clause 6.5.2 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported. If specific provisions to structured data types are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.4.6.2 of the present document.

#### Referenced structured data types

* + - * 1. Introduction

This clause defines the structured data types that are referenced by other data types.

All the referenced structured data types specified in clause 6.5.3 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported. If specific provisions to referenced structured data types are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.4.6.3 of the present document. In addition, all common structured data types in clause 4.4.1 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported if referenced/used by other data types as indicated above.

#### Simple data types and enumerations

* + - * 1. Introduction

The present clause defines simple data types and enumerations that can be referenced from data structures defined in other clauses. The simple data types defined in a request body, response body, or a structure type are defined in ETSI GS NFV-SOL 013 [[15]](#_bookmark17) or the present clause of the specification.

All the simple data types and enumerations specified in clause 6.5.4 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported. If specific provisions to simple data types and enumerations are considered with respect to the profiled API, these are specified in subsequent sub-clauses within clause 3.4.6.4 of the present document. In addition, all common simple data types and enumerations in clause 4.4.2 4.4.1 of ETSI GS NFV-SOL 003 [[11]](#_bookmark13) shall be supported if referenced/used by other data types as indicated above.

#### Other data types

* + - * 1. Introduction

The present clause defines other data types necessary for representing information to be considered by the orchestration procedures between the SMO and the DMS.

No particular other data types are defined currently for the present API.

# Annex A (Informative): Examples

## Overview

Annex A describes examples of usage of the O2dms ETSI NFV profile on various aspects.

## AAL examples

### Example #1: Definition of acceleration capability requirements in NF Deployment descriptors

The following example illustrates how the NF Deployment descriptor based on ETSI NFV’s VNFD, in particular based on TOSCA according to ETSI GS NFV-SOL 001 [[10],](#_bookmark12) can declare acceleration requirements for a VM-based NF Deployment unit by leveraging the definition of the requested additional capabilities. The example describes the use of the requested additional capabilities and the reference to AAL-related key-value.

NOTE 1: The example is not meant to be accurate in terms of necessary AAL parameterization (e.g., the requested capability name, values of "requested\_additional\_capability\_name", version parameters, "profile\_type" and "other\_profile\_params" in the example are just indicative) and its purpose is only to indicate the "placeholder" for such information in the VNFD. The example can be revisited in future versions of the present document based on the progress made in referenced O-RAN WG6 AAL specifications.

NOTE 2: The example only focuses on specific parts of the NF Deployment descriptors.

tosca\_definitions\_version: tosca\_simple\_yaml\_1\_3

description: Template of an NF Deployment example with acceleration requirements imports:

- etsi\_nfv\_sol001\_vnfd\_types.yaml # all of TOSCA VNFD types as defined in ETSI GS NFV-SOL 001

node\_types: MyExample.OranNF.1\_0:

derived\_from: tosca.nodes.nfv.VNF

...

topology\_template: substitution\_mappings:

node\_type: MyExample.OranNF.1\_0 requirements:

...

node\_templates: OranNF:

type: MyExample.OranNF.1\_0 properties:

...

interfaces:

...

NfDeploymentUnitExample:

type: tosca.nodes.nfv.Vdu.Compute properties:

...

capabilities: virtual\_compute:

properties:

...

requested\_additional\_capabilities:

|  |
| --- |
| aal\_profile\_fec: requested\_additional\_capability\_name: "AAL-FEC" support\_mandatory: true min\_requested\_additional\_capability\_version: "1"  preferred\_requested\_additional\_capability\_version: "1" target\_performance\_parameters:  profile\_type: "AAL-FEC" other\_profile\_params: "values"  aal\_profile\_downlink\_high\_phy: requested\_additional\_capability\_name: "AAL-DOWNLINK-High-PHY" support\_mandatory: true min\_requested\_additional\_capability\_version: "1"  preferred\_requested\_additional\_capability\_version: "2" target\_performance\_parameters:  profile\_type: "AAL-DOWNLINK-High-PHY"  other\_profile\_params: "values" |
| aal\_profile\_uplink\_high\_phy:  requested\_additional\_capability\_name: "AAL-UPLINK-High-PHY" support\_mandatory: true min\_requested\_additional\_capability\_version: "1"  preferred\_requested\_additional\_capability\_version: "2" target\_performance\_parameters:  profile\_type: "AAL-UPLINK-High-PHY"  other\_profile\_params: "values" |

In the example above:

* In red, the modelling concerns only to the choice of TOSCA as NF Deployment descriptor according to ETSI GS NFV-SOL 001 [[10]](#_bookmark12) and does not concern to modelling of AAL information or requirements. In other words, the red parts show how the NF designer models the capability within the descriptor by following the ETSI GS NFV-SOL 001 [[10]](#_bookmark12) specification.
* In blue, the modelling concerns to AAL and what is (expected) to be specified by O-RAN. The O-Cloud is expected to be provisioned with such a capability in order for the NF Deployment to be successful. The values here are only an example. The possible values and parameters names ("profile\_type", version number, "other\_profile\_params") are assumed to be referenced from O-RAN specification.
* In black, these are other parts that do not concern specifically to AAL and are only depicted for completeness of the example.

### Example #2: Definition of acceleration related configuration in NF Deployment descriptors and exchange of information on the O2dms interfaces

The following example illustrates how the NF Deployment descriptor based on ETSI NFV’s VNFD, in particular based on TOSCA according to ETSI GS NFV-SOL 001 [[10],](#_bookmark12) can declare configurable properties that can be used for providing runtime configuration information into boot data to be injected into a VM-based NF Deployment unit.

NOTE 1: The example is not meant to be accurate in terms of necessary AAL parameterization and its purpose is only to indicate the "placeholder" for such information in the VNFD. The example can be revisited in future versions of the present document based on the progress made in referenced O-RAN WG6 AAL specifications.

NOTE 2: The example only focuses on specific parts of the NF Deployment descriptors.

tosca\_definitions\_version: tosca\_simple\_yaml\_1\_3

description: Template of an NF Deployment example with acceleration requirements imports:

- etsi\_nfv\_sol001\_vnfd\_types.yaml # all of TOSCA VNFD types as defined in ETSI GS NFV-SOL 001

data\_types: MyExample.datatypes.nfv.VnfInstantiateAdditionalParameters:

derived\_from: tosca.datatypes.nfv.VnfOperationAdditionalParameters properties:

aal\_param\_01: type: string required: true default: value\_1

aal\_param\_02: type: string required: true default: value\_2

MyExample.datatypes.nfv.VnfConfigurableProperties: derived\_from: tosca.datatypes.nfv.VnfConfigurableProperties properties:

additional\_vnf\_configurable\_properties:

type: MyExample.datatypes.nfv.AdditionalVnfConfigurableProperties required: true

MyExample.datatypes.nfv.AdditionalVnfConfigurableProperties derived\_from: tosca.datatypes.nfv.VnfAdditionalConfigurableProperties properties:

aal\_param\_03: type: string required: true default: value\_3

node\_types: MyExample.OranNF.1\_0:

derived\_from: tosca.nodes.nfv.VNF properties:

...

configurable\_properties:

type: MyExample.datatypes.nfv.VnfConfigurableProperties interfaces:

Vnflcm:

type: tosca.interfaces.nfv.Vnflcm operations:

instantiate: inputs:

additional\_parameters:

type: MyExample.datatypes.nfv.VnfInstantiateAdditionalParameters required: false

topology\_template: substitution\_mappings:

node\_type: MyExample.OranNF.1\_0 requirements:

...

node\_templates: OranNF:

type: MyExample.OranNF.1\_0 properties:

...

interfaces: Vnflcm:

operations: instantiate:

implementation: ...

NfDeploymentUnitExample:

type: tosca.nodes.nfv.Vdu.Compute properties:

...

boot\_data: kvp\_data:

data:

aal\_address\_1: { get\_property: [OranNF, configurable\_properties, additional\_configurable\_properties, aal\_param\_03] }

...

In the example above:

* In red, the modelling concerns only to the choice of TOSCA as NF Deployment descriptor according to ETSI GS NFV-SOL 001 [[10]](#_bookmark12) and does not concern to modelling of AAL information or requirements. In other words, the red parts show how the NF designer models the capability within the descriptor by following the ETSI GS NFV-SOL 001 [[10]](#_bookmark12) specification.
* In blue, the modelling concerns to AAL and what is (expected) to be specified by O-RAN regarding relevant AAL configuration parameters. The values here are only an example. The possible values and parameters names ("profile\_type", version number, "other\_profile\_params") are assumed to be referenced from O-RAN specification or simply referenced from the respective O-RAN specification by the ETSI GS NFV-SOL 001 [[10].](#_bookmark12)
* In black, these are other parts that do not concern specifically to AAL and are only depicted for completeness of the example.

It is to be noted that MyExample.datatypes.nfv.VnfInstantiateAdditionalParameters and MyExample.datatypes.nfv.AdditionalVnfConfigurableProperties are datatypes defined by the NF designer and not further specified by ETSI GS NFV-SOL 001 [[10].](#_bookmark12) However, from an O-RAN standpoint, standard parameters related to AAL might be defined.

### Example #3: Exchange of information on the O2dms interfaces related to acceleration

Based on the example described in clause A.2.2, the following related example illustrates how the runtime configuration data can be provided by the SMO to the DMS as part of content message data during the NF Deployment instantiation request.

POST /vnflcm/v2/vnf\_instances/nf-deployment-A/instantiate Host: ocloud.com

Accept: application/json Content-Type: application/json Content-Length: XX

{

"flavourId": "small", "instantiationLevelId": "level-1", "additionalParams": {

"aal\_param\_01": "aal-profile-fec", "aal\_param\_02": "v1"

}

"vnfConfigurableProperties": { "aal\_param\_03": "PCI-address",

}

# Annex (informative): Change History

|  |  |  |
| --- | --- | --- |
| **Date** | **Revision** | **Description** |
| 2023.11.13 | 05.00.01 | Implemented CRs DCM-0024, DCM-0025, DCM-0026, DCM-0027, DCM-0028, DCM- 0029, DCM-0030, DCM-0031, |
| 2023.11.17 | 05.00.02 | Editorial updates based on review comments |
| 2023.11.28 | 06.00 | Final version 06.00 |
| 2024.03.19 | 06.00.01 | Implemented CRs DCM-0032, DCM-0033, DCM-0034 |
| 2024.04.01 | 07.00 | Final version 07.00 |
| 2024.07.09 | 07.00.01 | Implemented approved CRs for July Train:   * DCM-2024.06.25-WG6-CR-0035-O2dms\_ETSI\_NFV\_profile\_update\_API\_versions-v01 * DCM-2024.06.25-WG6-CR-0036-O2dms\_ETSI\_NFV\_profile\_bug\_fixes-v01 Rapporteur actions: * Corrected few occurrences of “error! Reference source not found” in clause 3.1.2 and   3.1.6.1.6. Added the corresponding missing references in the clause 1.2 (i.e. [36] and [37]).  - In reference [18] (SOL018), the notes 1 and 2 can be deleted, because as per CR-0035, a reference to a published version can be made. |
| 2024.07.23 | 07.00.02 | Addressing July Train 2024 review comments:  - Comment #4 on correcting cross-references. |
| 2024.07.29 | 08.00 | Final version for 2024 July Train Upgraded release number to R004. |