DRAFT - SOL005 - NSD Management Interface

Overview

DRAFT - SOL005 - NSD Management Interface IMPORTANT: Please note that this file might be not aligned to the current version of the ETSI Group Specification it refers to and has not been approved by the ETSI NFV ISG. In case of discrepancies the published ETSI Group Specification takes precedence. Please report bugs to https://forge.etsi.org/bugzilla/buglist.cgi?component=Nfv-Openapis

Version information

Version : 2.4.1

Contact information

Contact : NFV-SOL WG

License information

License : ETSI Forge copyright notice
License URL : https://forge.etsi.org/etsi-forge-copyright-notice.txt
Terms of service : null

Consumes

- `application/json`

Produces

- `application/json`

Paths

Create NSD Info

```
POST /ns_descriptors
```

Description

The POST method is used to create a new NS descriptor resource. This method shall follow the provisions specified in the Tables 5.4.2.3.1-1 and 5.4.2.3.1-2 of GS NFV-SOL 005 for URI query parameters, request and response data structures, and response codes.

Parameters

Type	Name	Schema
Body	**body** *required*	body

body

Name	Description	Schema
CreateNsdInfoRequest *required*	This type creates a completely new NS descriptor resource.	CreateNsdInfoRequest

CreateNsdInfoRequest

Name	Description	Schema
userDefinedData *optional*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object

Responses

HTTP Code	Description	Schema
201	Status 201 **Headers** : `Location` (string) : The HTTP response shall include a "Location" HTTP header that contains the resource URI of the new NS descriptor resource.	Response 201

Response 201

Name	Description	Schema
NsdInfo *optional*	This type represents a response for the query NSD operation. It shall comply with the provisions defined in Table 5.5.2.2-1 of GS NFV-SOL 005.	NsdInfo

NsdInfo

Name	Description	Schema
_links *required*	Links to resources related to this resource.	_links
id *required*	Identifier of the onboarded individual NS descriptor resource. This identifier is allocated by the NFVO.	string
nestedNsdInfoIds *optional*	Identifies the NsdInfo element for the nested NSD referenced by the on-boarded NS descriptor resource. At least one of the attributes – vnfPkgId and nestedNsdInfoId shall be present, after the NSD is on-boarded.	string
nsdDesigner *optional*	Designer of the on-boarded NSD. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdId *optional*	This identifier, which is allocated by the NSD designer, identifies the NSD in a globally unique way. It is copied from the NSD content and shall be present after the NSD content is on-boarded.	object
nsdInvariantId *optional*	This identifier, which is allocated by the NSD designer, identifies an NSD in a version independent manner. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdName *optional*	Name of the onboarded NSD. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdOnboardingState *required*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)

Name	Description	Schema
nsdOperationalState *required*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
nsdUsageState *required*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Version of the on-boarded NSD. The NSD version is a string of variable length.This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
onboardingFailureDetails *optional*	The definition of the general "ProblemDetails" data structure from IETF RFC 7807 [27] is reproduced in Table 4.3.5.3-1 of GS NFV-SOL 005. Compared to the general framework defined in IETF RFC 7807 [27], the "status" and "detail" attributes are mandated to be included by the present document, to ensure that the response contains additional textual information about an error. IETF RFC 7807 [27] foresees extensibility of the "ProblemDetails" type. It is possible that particular APIs in the present document, or particular implementations, define extensions to define additional attributes that provide more information about the error. The description column only provides some explanation of the meaning to facilitate understanding of the design. For a full description, see IETF RFC 7807 [27].	onboardingFailureDetails
pnfdInfoIds *optional*	Identifies the PnfdInfo element for the PNFD referenced by the on-boarded NS descriptor resource.	string

Name	Description	Schema
userDefinedData *optional*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object
vnfPkgIds *optional*	Identifies the VNF package for the VNFD referenced by the on-boarded NS descriptor resource.	string

_links

Name	Description	Schema
nsd_content *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
self *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

onboardingFailureDetails

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

Query NSD Info

```
GET /ns_descriptors
```

Description

The GET method queries information about multiple NS descriptor resources. This method shall follow the provisions specified in the Tables 5.4.2.3.2-1 and 5.4.2.3.2-2 of GS NFV-SOL 005 for URI query parameters, request and response data structures, and response codes.

Responses

HTTP Code	Description	Schema
200	Information about zero or more NS descriptors. The response body shall contain a representation of zero or more NS descriptors, as defined in clause 5.5.2.2 of GS NFV-SOL 005.	Response 200
400	There are two possible scenarios listed below. Error: Invalid attribute-based filtering parameters. The response body shall contain a ProblemDetails structure, in which the "detail" attribute should convey more information about the error. Error: Invalid attribute selector. The response body shall contain a ProblemDetails structure, in which the "detail" attribute should convey more information about the error.	No Content

Response 200

Name	Description	Schema
_links *required*	Links to resources related to this resource.	_links
id *required*	Identifier of the onboarded individual NS descriptor resource. This identifier is allocated by the NFVO.	string
nestedNsdInfoIds *optional*	Identifies the NsdInfo element for the nested NSD referenced by the on-boarded NS descriptor resource. At least one of the attributes – vnfPkgId and nestedNsdInfoId shall be present, after the NSD is on-boarded.	string

Name	Description	Schema
nsdDesigner *optional*	Designer of the on-boarded NSD. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdId *optional*	This identifier, which is allocated by the NSD designer, identifies the NSD in a globally unique way. It is copied from the NSD content and shall be present after the NSD content is on-boarded.	object
nsdInvariantId *optional*	This identifier, which is allocated by the NSD designer, identifies an NSD in a version independent manner. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdName *optional*	Name of the onboarded NSD. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdOnboardingState *required*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *required*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
nsdUsageState *required*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)

Name	Description	Schema
nsdVersion *optional*	Version of the on-boarded NSD. The NSD version is a string of variable length.This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
onboardingFailureDetails *optional*	The definition of the general "ProblemDetails" data structure from IETF RFC 7807 [27] is reproduced in Table 4.3.5.3-1 of GS NFV-SOL 005. Compared to the general framework defined in IETF RFC 7807 [27], the "status" and "detail" attributes are mandated to be included by the present document, to ensure that the response contains additional textual information about an error. IETF RFC 7807 [27] foresees extensibility of the "ProblemDetails" type. It is possible that particular APIs in the present document, or particular implementations, define extensions to define additional attributes that provide more information about the error. The description column only provides some explanation of the meaning to facilitate understanding of the design. For a full description, see IETF RFC 7807 [27].	onboardingFailureDetails
pnfdInfoIds *optional*	Identifies the PnfdInfo element for the PNFD referenced by the on-boarded NS descriptor resource.	string
userDefinedData *optional*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object
vnfPkgIds *optional*	Identifies the VNF package for the VNFD referenced by the on-boarded NS descriptor resource.	string

_links

Name	Description	Schema
nsd_content *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
self *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

onboardingFailureDetails

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

Query NSD Info

```
GET /ns_descriptors/{nsdInfoId}
```

Description

The GET method reads information about an individual NS descriptor. This method shall follow the provisions specified in GS NFV-SOL 005 Tables 5.4.3.3.2-1 and 5.4.3.3.2-2 of GS NFV-SOL 005 for URI query parameters, request and response data structures, and response codes.

Parameters

Type	Name	Schema
Path	**nsdInfoId** *required*	string

Responses

HTTP Code	Description	Schema
200	Information about the individual NS descriptor. The response body shall contain a representation of the individual NS descriptor, as defined in clause 5.5.2.2 of GS NFV-SOL 005.	Response 200

Response 200

Name	Description	Schema
_links *required*	Links to resources related to this resource.	_links
id *required*	Identifier of the onboarded individual NS descriptor resource. This identifier is allocated by the NFVO.	string
nestedNsdInfoIds *optional*	Identifies the NsdInfo element for the nested NSD referenced by the on-boarded NS descriptor resource. At least one of the attributes – vnfPkgId and nestedNsdInfoId shall be present, after the NSD is on-boarded.	string
nsdDesigner *optional*	Designer of the on-boarded NSD. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdId *optional*	This identifier, which is allocated by the NSD designer, identifies the NSD in a globally unique way. It is copied from the NSD content and shall be present after the NSD content is on-boarded.	object
nsdInvariantId *optional*	This identifier, which is allocated by the NSD designer, identifies an NSD in a version independent manner. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdName *optional*	Name of the onboarded NSD. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string

Name	Description	Schema
nsdOnboardingState *required*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *required*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
nsdUsageState *required*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Version of the on-boarded NSD. The NSD version is a string of variable length.This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string

Name	Description	Schema
onboardingFailureDetails *optional*	The definition of the general "ProblemDetails" data structure from IETF RFC 7807 [27] is reproduced in Table 4.3.5.3-1 of GS NFV-SOL 005. Compared to the general framework defined in IETF RFC 7807 [27], the "status" and "detail" attributes are mandated to be included by the present document, to ensure that the response contains additional textual information about an error. IETF RFC 7807 [27] foresees extensibility of the "ProblemDetails" type. It is possible that particular APIs in the present document, or particular implementations, define extensions to define additional attributes that provide more information about the error. The description column only provides some explanation of the meaning to facilitate understanding of the design. For a full description, see IETF RFC 7807 [27].	onboardingFailureDetails
pnfdInfoIds *optional*	Identifies the PnfdInfo element for the PNFD referenced by the on-boarded NS descriptor resource.	string
userDefinedData *optional*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object
vnfPkgIds *optional*	Identifies the VNF package for the VNFD referenced by the on-boarded NS descriptor resource.	string

_links

Name	Description	Schema
nsd_content *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
self *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

onboardingFailureDetails

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

Delete NSD

```
DELETE /ns_descriptors/{nsdInfoId}
```

Description

The DELETE method deletes an individual NS descriptor resource. An individual NS descriptor resource can only be deleted when there is no NS instance using it (i.e. usageState = NOT_IN_USE) and has been disabled already (i.e. operationalState = DISABLED). Otherwise, the DELETE method shall fail. This method shall follow the provisions specified in the Tables 5.4.3.3.5-1 and 5.4.3.3.5-2 of GS NFV-SOL 005 for URI query parameters, request and response data structures, and response codes.

Parameters

Type	Name	Schema
Path	**nsdInfoId** *required*	string

Responses

HTTP Code	Description	Schema
204	The operation has completed successfully. The response body shall be empty.	object
409	Status 409	Response 409

Response 409

Name	Description	Schema
ProblemDetails *required*	The definition of the general "ProblemDetails" data structure from IETF RFC 7807 [27] is reproduced in Table 4.3.5.3-1 of GS NFV-SOL 005. Compared to the general framework defined in IETF RFC 7807 [27], the "status" and "detail" attributes are mandated to be included by the present document, to ensure that the response contains additional textual information about an error. IETF RFC 7807 [27] foresees extensibility of the "ProblemDetails" type. It is possible that particular APIs in the present document, or particular implementations, define extensions to define additional attributes that provide more information about the error. The description column only provides some explanation of the meaning to facilitate understanding of the design. For a full description, see IETF RFC 7807 [27].	ProblemDetails

ProblemDetails

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string

Name	Description	Schema
type *optional*	String formatted according to IETF RFC 3986 [10].	object

Update NSD Info

```
PATCH /ns_descriptors/{nsdInfoId}
```

Description

The PATCH method modifies the operational state and/or user defined data of an individual NS descriptor resource. This method can be used to: 1) Enable a previously disabled individual NS descriptor resource, allowing again its use for instantiation of new network service with this descriptor. The usage state (i.e. "IN_USE/NOT_IN_USE") shall not change as a result. 2) Disable a previously enabled individual NS descriptor resource, preventing any further use for instantiation of new network service(s) with this descriptor. The usage state (i.e. "IN_USE/NOT_IN_USE") shall not change as a result. 3) Modify the user defined data of an individual NS descriptor resource. This method shall follow the provisions specified in the Tables 5.4.3.3.4-1 and 5.4.3.3.4-2 for URI query parameters, request and response data structures, and response codes.

Parameters

Type	Name	Schema
Path	**nsdInfoId** *required*	string
Body	**body** *required*	body

body

Name	Description	Schema
NsdInfoModifications *required*	This type represents attribute modifications for an individual NS descriptor resource based on the "NsdInfo" data type. The attributes of "NsdInfo" that can be modified are included in the "NsdInfoModifications" data type. **NOTE** At least one of the attributes - nsdOperationalState and userDefinedData - shall be present.	NsdInfoModifications

NsdInfoModifications

Name	Description	Schema
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
userDefinedData *optional*	Modifications of the "userDefinedData" attribute in "NsdInfo" data type. See note. If present, these modifications shall be applied according to the rules of JSON Merge PATCH (see IETF RFC 7396 [25]). NOTE: At least one of the attributes - nsdOperationalState and userDefinedData - shall be present.	object

Responses

HTTP Code	Description	Schema
200	Status 200	Response 200
409	Status 409	Response 409
412	Status 412	Response 412

Response 200

Name	Description	Schema
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)

Name	Description	Schema
userDefinedData *optional*	Modifications of the "userDefinedData" attribute in "NsdInfo" data type. See note. If present, these modifications shall be applied according to the rules of JSON Merge PATCH (see IETF RFC 7396 [25]). NOTE: At least one of the attributes - nsdOperationalState and userDefinedData - shall be present.	object

Response 409

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

Response 412

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer

Name	Description	Schema
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

Get NSD Content

```
GET /ns_descriptors/{nsdInfoId}/nsd_content
```

Description

The GET method fetches the content of the NSD.

The NSD can be implemented as a single file or as a collection of multiple files. If the NSD is implemented in the form of multiple files, a ZIP file embedding these files shall be returned. If the NSD is implemented as a single file, either that file or a ZIP file embedding that file shall be returned.

The selection of the format is controlled by the "Accept" HTTP header passed in the GET request:

- If the "Accept" header contains only "text/plain" and the NSD is implemented as a single file, the file shall be returned; otherwise, an error message shall be returned.
- If the "Accept" header contains only "application/zip", the single file or the multiple files that make up the NSD shall be returned embedded in a ZIP file.
- If the "Accept" header contains both "text/plain" and "application/zip", it is up to the NFVO to choose the format to return for a single-file NSD; for a multi-file NSD, a ZIP file shall be returned.

NOTE	The structure of the NSD zip file is outside the scope of the present document.

Parameters

Type	Name	Description	Schema
Header	**Accept** *required*	The request shall contain the appropriate entries in the "Accept" HTTP header as defined above.	string

Type	Name	Description	Schema
Header	**Range** *optional*	The request may contain a "Range" HTTP header to obtain single range of bytes from the NSD file. This can be used to continue an aborted transmission. If the NFVO does not support range requests, the NFVO shall ignore the 'Range" header, process the GET request, and return the whole NSD file with a 200 OK response (rather than returning a 4xx error status code).	string
Path	**nsdInfoId** *required*		string

Responses

HTTP Code	Description	Schema
200	On success, the content of the NSD is returned. The payload body shall contain a copy of the file representing the NSD or a ZIP file that contains the file or multiple files representing the NSD, as specified above. The "Content-Type" HTTP header shall be set according to the format of the returned file, i.e. to "text/plain" for a YAML file or to "application/zip" for a ZIP file. **Headers** : `Content-Type` (string)	object
206	On success, if the NFVO supports range requests, a single consecutive byte range from the content of the NSD file is returned. The response body shall contain the requested part of the NSD file. The "Content-Range" HTTP header shall be provided according to IETF RFC 7233 [23]. The "Content-Type" HTTP header shall be set as defined above for the "200 OK" response. **Headers** : `Content-Range` (string) `Content-Type` (string)	No Content
406	Status 406	object
409	Status 409	object

HTTP Code	Description	Schema
416	Status 416	object

Upload NSD

```
PUT /ns_descriptors/{nsdInfoId}/nsd_content
```

Description

The PUT method is used to upload the content of a NSD. The NSD to be uploaded can be implemented as a single file or as a collection of multiple files, as defined in clause 5.4.4.3.2 of GS NFV-SOL 005. If the NSD is implemented in the form of multiple files, a ZIP file embedding these files shall be uploaded. If the NSD is implemented as a single file, either that file or a ZIP file embedding that file shall be uploaded. The "Content-Type" HTTP header in the PUT request shall be set accordingly based on the format selection of the NSD. If the NSD to be uploaded is a text file, the "Content-Type" header is set to "text/plain". If the NSD to be uploaded is a zip file, the "Content-Type" header is set to "application/zip". This method shall follow the provisions specified in the Tables 5.4.4.3.3-1 and 5.4.4.3.3-2 of GS-NFV-SOL 005 for URI query parameters, request and response data structures, and response codes.

Parameters

Type	Name	Description	Schema
Header	**Content-Type** *optional*	The payload body contains a copy of the file representing the NSD or a ZIP file that contains the file or multiple files representing the NSD, as specified above. The request shall set the "Content-Type" HTTP header as defined above.	string
Path	**nsdInfoId** *required*		string
Body	**body** *required*		object

Responses

HTTP Code	Description	Schema
202	Status 202	object

HTTP Code	Description	Schema
204	The NSD content was successfully uploaded and validated (synchronous mode). The response body shall be empty.	No Content
409	Error: The operation cannot be executed currently, due to a conflict with the state of the resource. Typically, this is due to the fact that the NsdOnboardingState has a value other than CREATED. The response body shall contain a ProblemDetails structure, in which the "detail" attribute shall convey more information about the error.	Response 409

Response 409

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

Create PNFD Info

```
POST /pnf_descriptors
```

Description

The POST method is used to create a new PNF descriptor resource.

Parameters

Type	Name	Schema
Body	**body** *required*	object

Responses

HTTP Code	Description	Schema
201	Status 201	object

Query PFND Info

```
GET /pnf_descriptors
```

Description

The GET method queries information about multiple PNF descriptor resources.

Parameters

Type	Name	Description	Schema
Query	**all_fields** *optional*	Include all complex attributes in the response. See clause 4.3.3 for details. The NFVO shall support this parameter.	string
Query	**exclude_default** *optional*	Indicates to exclude the following complex attributes from the response. See clause 4.3.3 for details. The NFVO shall support this parameter. The following attributes shall be excluded from the PnfdInfo structure in the response body if this parameter is provided, or none of the parameters "all_fields," "fields", "exclude_fields", "exclude_default" are provided: userDefinedData.	string

Responses

HTTP Code	Description	Schema
200	Status 200	object

Query PNFD Info

```
GET /pnf_descriptors/{pnfdInfoId}
```

Description

The GET method reads information about an individual PNF descriptor. This method shall follow the provisions specified in the Tables 5.4.6.3.2-1 and 5.4.6.3.2-2 of GS NFV-SOL 005 for URI query parameters, request and response data structures, and response codes.

Parameters

Type	Name	Schema
Path	**pnfdInfoId** *required*	string

Responses

HTTP Code	Description	Schema
200	Information about the individual PNFD descriptor. The response body shall contain a representation of the individual PNF descriptor, as defined in clause 5.5.2.5 of GS NFV-SOL 005.	No Content

Delete PNFD

```
DELETE /pnf_descriptors/{pnfdInfoId}
```

Description

The DELETE method deletes an individual PNF descriptor resource. An individual PNF descriptor resource can only be deleted when there is no NS instance using it or there is NSD referencing it. To delete all PNFD versions identified by a particular value of the "pnfdInvariantId" attribute, the procedure is to first use the GET method with filter "pnfdInvariantId" towards the PNF descriptors resource to find all versions of the PNFD. Then, the client uses the DELETE method described in this clause to delete each PNFD version individually. This method shall follow the provisions specified in the Tables 5.4.6.3.5-1 and 5.4.6.3.5-2 of GS NFV-SOL 005 for URI query parameters, request and

response data structures, and response codes.

Parameters

Type	Name	Schema
Path	**pnfdInfoId** *required*	string

Responses

HTTP Code	Description	Schema
204	The operation has completed successfully. The response body shall be empty.	No Content

Update PNFD Info

```
PATCH /pnf_descriptors/{pnfdInfoId}
```

Description

The PATCH method modifies the user defined data of an individual PNF descriptor resource. This method shall follow the provisions specified in the Tables 5.4.6.3.4-1 and 5.4.6.3.4-2 for URI query parameters, request and response data structures, and response codes.

Parameters

Type	Name	Schema
Path	**pnfdInfoId** *required*	string
Body	**body** *required*	body

body

Name	Description	Schema
userDefinedData *required*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object

Responses

HTTP Code	Description	Schema
200	Status 200	Response 200
412	Status 412	Response 412

Response 200

Name	Description	Schema
userDefinedData *required*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object

Response 412

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string

Name	Description	Schema
type *optional*	String formatted according to IETF RFC 3986 [10].	object

Get PNFD Content

```
GET /pnf_descriptors/{pnfdInfoId}/pnfd_content
```

Description

The GET method fetches the content of the PNFD.

Parameters

Type	Name	Schema
Path	**pnfdInfoId** *required*	string

Responses

HTTP Code	Description	Schema
200	On success, the content of the PNFD is returned. The payload body shall contain a copy of the file representing the PNFD. The "Content-Type" HTTP header shall be set to "text/plain".	No Content

Upload PNFD

```
PUT /pnf_descriptors/{pnfdInfoId}/pnfd_content
```

Description

The PUT method is used to upload the content of a PNFD. This method shall follow the provisions specified in the Tables 5.4.7.3.3-1 and 5.4.7.3.3-2 of GS NFV-SOL 005for URI query parameters, request and response data structures, and response codes.

Parameters

Type	Name	Description	Schema
Header	**Content-Type** *optional*	The request shall set the "Content-Type" HTTP header to "text/plain".	string
Path	**pnfdInfoId** *required*		string
Body	**body** *required*		object

Responses

HTTP Code	Description	Schema
204	The PNFD content was successfully uploaded and validated. The response body shall be empty.	No Content
409	Status 409	Response 409

Response 409

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

Subscribe

```
POST /subscriptions
```

Description

The POST method creates a new subscription. This method shall support the URI query parameters, request and response data structures, and response codes, as specified in the Tables 5.4.8.3.1-1 and 5.4.8.3.1-2 of GS-NFV SOL 005. Creation of two subscription resources with the same callbackURI and the same filter can result in performance degradation and will provide duplicates of notifications to the OSS, and might make sense only in very rare use cases. Consequently, the NFVO may either allow creating a subscription resource if another subscription resource with the same filter and callbackUri already exists (in which case it shall return the "201 Created" response code), or may decide to not create a duplicate subscription resource (in which case it shall return a "303 See Other" response code referencing the existing subscription resource with the same filter and callbackUri).

Parameters

Type	Name	Schema
Body	**body** *required*	body

body

Name	Description	Schema
authentication *optional*		authentication
callbackUri *required*	The URI of the endpoint to send the notification to.	string
filter *optional*	This type represents a subscription filter related to notifications about NSD management. It shall comply with the provisions defined in Table 5.5.3.2-1 of GS NFV-SOL 005. At a particular nesting level in the filter structure, the following applies: All attributes shall match in order for the filter to match (logical "and" between different filter attributes). If an attribute is an array, the attribute shall match if at least one of the values in the array matches (logical "or" between the values of one filter attribute).	filter

authentication

Name	Description	Schema
authType *optional*	Defines the types of Authentication/ Authorization the API consumer is willing to accept when receiving a notification. Permitted values: BASIC: In every HTTP request to the notification endpoint, use HTTP Basic authentication with the client credentials. OAUTH2_CLIENT_CREDENTIALS: In every HTTP request to the notification endpoint, use an OAuth 2.0 Bearer token, obtained using the client credentials grant type. TLS_CERT: Every HTTP request to the notification endpoint is sent over a mutually authenticated TLS session. i.e. not only server is authenticated, but also the client is authenticated during the TLS tunnel setup.	enum (BASIC, OAUTH2_CLIENT_CREDENTIALS, TLS_CERT)
paramsBasic *optional*	Parameters for authentication/authorization using BASIC. Shall be present if authType is "BASIC" and the contained information has not been provisioned out of band. Shall be absent otherwise.	paramsBasic
paramsOauth2ClientCredentials *optional*	Parameters for authentication/authorization using OAUTH2_CLIENT_CREDENTIALS. Shall be present if authType is "OAUTH2_CLIENT_CREDENTIALS" and the contained information has not been provisioned out of band. Shall be absent otherwise.	paramsOauth2ClientCredentials

paramsBasic

Name	Description	Schema
password *optional*	Password to be used in HTTP Basic authentication. Shall be present if it has not been provisioned out of band.	string
userName *optional*	Username to be used in HTTP Basic authentication. Shall be present if it has not been provisioned out of band.	string

paramsOauth2ClientCredentials

Name	Description	Schema
clientId *optional*	Client identifier to be used in the access token request of the OAuth 2.0 client credentials grant type. Shall be present if it has not been provisioned out of band.	string

filter

Name	Description	Schema
nestedNsdInfoIds *optional*	Match the NsdInfo identifier for the nested NSD referenced by the on-boarded NSD.	string
notificationTypes *optional*	Match particular notification types. Permitted values: NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification. The permitted values of the "notificationTypes" attribute are spelled exactly as the names of the notification types to facilitate automated code generation systems.	enum (NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification, PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification)
nsdDesigner *optional*	Match the NSD designer of the on-boarded NSD.	string
nsdId *optional*	Match the NSD identifier, which is allocated by the NSD designer. The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInfoId *optional*	Match the NsdInfo identifier which is allocated by the NFVO. Note: The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInvariantId *optional*	Match the NSD invariant identifier which is allocated by the NSD designer and identifies an NSD in a version independent manner.	string
nsdName *optional*	Match the name of the onboarded NSD.	string

Name	Description	Schema
nsdOnboardingState *optional*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
nsdUsageState *optional*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Match the NSD version listed as part of this attribute. The NSD version is a string of variable length.	string
pnfdId *optional*	Match the PNFD identifier which is copied from the PNFD content. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInfoIds *optional*	Match the PnfdInfo identifier for the PNFD referenced by the on-boarded NSD. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInvariantId *optional*	Match the PNFD in a version independent manner.	string

Name	Description	Schema
pnfdName *optional*	Match the name of the on-boarded PNFD.	string
pnfdOnboardingState *optional*	The enumeration PnfdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.6-1 of GS-NFV SOL005. It indicates the onboarding state of the individual PNF descriptor resource. CREATED = The PNF descriptor resource is created. UPLOADING = The associated PNFD content is being uploaded. PROCESSING = The associated PNFD content is being processed, e.g. validation. ONBOARDED = The associated PNFD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDING)
pnfdProvider *optional*	Match the provider of the on-boarded PNFD.	string
pnfdUsageState *optional*	The enumeration PnfdUsageStateType shall comply with the provisions defined in Table 5.5.4.7-1 of GS NFV-SOL005. It indicates the usage state of the resource. IN-USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
pnfdVersion *optional*	Match the PNFD designer of the on-boarded PNFD. The PNFD version is a string of variable length.	string
vnfPkgIds *optional*	Match VNF packages with a package identifier listed in the attribute.	string

Responses

HTTP Code	Description	Schema
201	Status 201 **Headers** : `Location` (string) : The HTTP response shall include a "Location:" HTTP header that points to the created subscription resource.	Response 201

Response 201

Name	Description	Schema
_links *required*	Links to resources related to this resource.	_links
callbackUri *required*	String formatted according to IETF RFC 3986 [10].	object
filter *optional*	This type represents a subscription filter related to notifications about NSD management. It shall comply with the provisions defined in Table 5.5.3.2-1 of GS NFV-SOL 005. At a particular nesting level in the filter structure, the following applies: All attributes shall match in order for the filter to match (logical "and" between different filter attributes). If an attribute is an array, the attribute shall match if at least one of the values in the array matches (logical "or" between the values of one filter attribute).	filter
id *required*	Identifier of this subscription resource	string

_links

Name	Description	Schema
self *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

filter

Name	Description	Schema
nestedNsdInfoIds *optional*	Match the NsdInfo identifier for the nested NSD referenced by the on-boarded NSD.	string

Name	Description	Schema
notificationTypes *optional*	Match particular notification types. Permitted values: NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification. The permitted values of the "notificationTypes" attribute are spelled exactly as the names of the notification types to facilitate automated code generation systems.	enum (NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification, PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification)
nsdDesigner *optional*	Match the NSD designer of the on-boarded NSD.	string
nsdId *optional*	Match the NSD identifier, which is allocated by the NSD designer. The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInfoId *optional*	Match the NsdInfo identifier which is allocated by the NFVO. Note: The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInvariantId *optional*	Match the NSD invariant identifier which is allocated by the NSD designer and identifies an NSD in a version independent manner.	string
nsdName *optional*	Match the name of the onboarded NSD.	string

Name	Description	Schema
nsdOnboardingState *optional*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
nsdUsageState *optional*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Match the NSD version listed as part of this attribute. The NSD version is a string of variable length.	string
pnfdId *optional*	Match the PNFD identifier which is copied from the PNFD content. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInfoIds *optional*	Match the PnfdInfo identifier for the PNFD referenced by the on-boarded NSD. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInvariantId *optional*	Match the PNFD in a version independent manner.	string

Name	Description	Schema
pnfdName *optional*	Match the name of the on-boarded PNFD.	string
pnfdOnboardingState *optional*	The enumeration PnfdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.6-1 of GS-NFV SOL005. It indicates the onboarding state of the individual PNF descriptor resource. CREATED = The PNF descriptor resource is created. UPLOADING = The associated PNFD content is being uploaded. PROCESSING = The associated PNFD content is being processed, e.g. validation. ONBOARDED = The associated PNFD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDING)
pnfdProvider *optional*	Match the provider of the on-boarded PNFD.	string
pnfdUsageState *optional*	The enumeration PnfdUsageStateType shall comply with the provisions defined in Table 5.5.4.7-1 of GS NFV-SOL005. It indicates the usage state of the resource. IN-USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
pnfdVersion *optional*	Match the PNFD designer of the on-boarded PNFD. The PNFD version is a string of variable length.	string
vnfPkgIds *optional*	Match VNF packages with a package identifier listed in the attribute.	string

Query Subscription Information

```
GET /subscriptions
```

Description

The GET method queries the list of active subscriptions of the functional block that invokes the method. It can be used e.g. for resynchronization after error situations. This method shall support the URI query parameters, request and response data structures, and response codes, as specified in the Tables 5.4.8.3.2-1 and 5.4.8.3.2-2 of GS NFV-SOL 005.

Responses

HTTP Code	Description	Schema
200	Status 200	< Response 200 > array
303	A subscription with the same callbackURI and the same filter already exits and the policy of the NFVO is to not create redundant subscriptions. The response body shall be empty. **Headers** : `Location` (string) : The HTTP response shall include a "Location" HTTP header that contains the resource URI of the existing subscription resource.	No Content

Response 200

Name	Description	Schema
_links *required*	Links to resources related to this resource.	_links
callbackUri *required*	String formatted according to IETF RFC 3986 [10].	object
filter *optional*	This type represents a subscription filter related to notifications about NSD management. It shall comply with the provisions defined in Table 5.5.3.2-1 of GS NFV-SOL 005. At a particular nesting level in the filter structure, the following applies: All attributes shall match in order for the filter to match (logical "and" between different filter attributes). If an attribute is an array, the attribute shall match if at least one of the values in the array matches (logical "or" between the values of one filter attribute).	filter
id *required*	Identifier of this subscription resource	string

_links

Name	Description	Schema
self *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

filter

Name	Description	Schema
nestedNsdInfoIds *optional*	Match the NsdInfo identifier for the nested NSD referenced by the on-boarded NSD.	string
notificationTypes *optional*	Match particular notification types. Permitted values: NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification. The permitted values of the "notificationTypes" attribute are spelled exactly as the names of the notification types to facilitate automated code generation systems.	enum (NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification, PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification)
nsdDesigner *optional*	Match the NSD designer of the on-boarded NSD.	string
nsdId *optional*	Match the NSD identifier, which is allocated by the NSD designer. The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInfoId *optional*	Match the NsdInfo identifier which is allocated by the NFVO. Note: The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInvariantId *optional*	Match the NSD invariant identifier which is allocated by the NSD designer and identifies an NSD in a version independent manner.	string
nsdName *optional*	Match the name of the onboarded NSD.	string

Name	Description	Schema
nsdOnboardingState *optional*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
nsdUsageState *optional*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Match the NSD version listed as part of this attribute. The NSD version is a string of variable length.	string
pnfdId *optional*	Match the PNFD identifier which is copied from the PNFD content. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInfoIds *optional*	Match the PnfdInfo identifier for the PNFD referenced by the on-boarded NSD. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInvariantId *optional*	Match the PNFD in a version independent manner.	string

Name	Description	Schema
pnfdName *optional*	Match the name of the on-boarded PNFD.	string
pnfdOnboardingState *optional*	The enumeration PnfdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.6-1 of GS-NFV SOL005. It indicates the onboarding state of the individual PNF descriptor resource. CREATED = The PNF descriptor resource is created. UPLOADING = The associated PNFD content is being uploaded. PROCESSING = The associated PNFD content is being processed, e.g. validation. ONBOARDED = The associated PNFD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDING)
pnfdProvider *optional*	Match the provider of the on-boarded PNFD.	string
pnfdUsageState *optional*	The enumeration PnfdUsageStateType shall comply with the provisions defined in Table 5.5.4.7-1 of GS NFV-SOL005. It indicates the usage state of the resource. IN-USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
pnfdVersion *optional*	Match the PNFD designer of the on-boarded PNFD. The PNFD version is a string of variable length.	string
vnfPkgIds *optional*	Match VNF packages with a package identifier listed in the attribute.	string

Query Subscription Information

```
GET /subscriptions/{subscriptionId}
```

Description

The GET method retrieves information about a subscription by reading an individual subscription resource. This method shall support the URI query parameters, request and response data structures, and response codes, as specified in the Tables 5.4.9.3.2-1 and 5.4.9.3.2-2.

Parameters

Type	Name	Schema
Path	**subscriptionId** *required*	string

Responses

HTTP Code	Description	Schema
200	Status 200	Response 200

Response 200

Name	Description	Schema
_links *required*	Links to resources related to this resource.	_links
callbackUri *required*	String formatted according to IETF RFC 3986 [10].	object
filter *optional*	This type represents a subscription filter related to notifications about NSD management. It shall comply with the provisions defined in Table 5.5.3.2-1 of GS NFV-SOL 005. At a particular nesting level in the filter structure, the following applies: All attributes shall match in order for the filter to match (logical "and" between different filter attributes). If an attribute is an array, the attribute shall match if at least one of the values in the array matches (logical "or" between the values of one filter attribute).	filter
id *required*	Identifier of this subscription resource	string

_links

Name	Description	Schema
self *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

filter

Name	Description	Schema
nestedNsdInfoIds *optional*	Match the NsdInfo identifier for the nested NSD referenced by the on-boarded NSD.	string
notificationTypes *optional*	Match particular notification types. Permitted values: NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification. The permitted values of the "notificationTypes" attribute are spelled exactly as the names of the notification types to facilitate automated code generation systems.	enum (NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification, PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification)
nsdDesigner *optional*	Match the NSD designer of the on-boarded NSD.	string
nsdId *optional*	Match the NSD identifier, which is allocated by the NSD designer. The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInfoId *optional*	Match the NsdInfo identifier which is allocated by the NFVO. Note: The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInvariantId *optional*	Match the NSD invariant identifier which is allocated by the NSD designer and identifies an NSD in a version independent manner.	string
nsdName *optional*	Match the name of the onboarded NSD.	string

Name	Description	Schema
nsdOnboardingState *optional*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
nsdUsageState *optional*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Match the NSD version listed as part of this attribute. The NSD version is a string of variable length.	string
pnfdId *optional*	Match the PNFD identifier which is copied from the PNFD content. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInfoIds *optional*	Match the PnfdInfo identifier for the PNFD referenced by the on-boarded NSD. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInvariantId *optional*	Match the PNFD in a version independent manner.	string

Name	Description	Schema
pnfdName *optional*	Match the name of the on-boarded PNFD.	string
pnfdOnboardingState *optional*	The enumeration PnfdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.6-1 of GS-NFV SOL005. It indicates the onboarding state of the individual PNF descriptor resource. CREATED = The PNF descriptor resource is created. UPLOADING = The associated PNFD content is being uploaded. PROCESSING = The associated PNFD content is being processed, e.g. validation. ONBOARDED = The associated PNFD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDING)
pnfdProvider *optional*	Match the provider of the on-boarded PNFD.	string
pnfdUsageState *optional*	The enumeration PnfdUsageStateType shall comply with the provisions defined in Table 5.5.4.7-1 of GS NFV-SOL005. It indicates the usage state of the resource. IN-USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
pnfdVersion *optional*	Match the PNFD designer of the on-boarded PNFD. The PNFD version is a string of variable length.	string
vnfPkgIds *optional*	Match VNF packages with a package identifier listed in the attribute.	string

Terminate Subscription

```
DELETE /subscriptions/{subscriptionId}
```

Description

The DELETE method terminates an individual subscription. This method shall support the URI query parameters, request and response data structures, and response codes, as specified in the Tables 5.4.9.3.5-1 and 5.4.9.3.3-2 of GS NFV-SOL 005.

Parameters

Type	Name	Schema
Path	**subscriptionId** *required*	string

Responses

HTTP Code	Description	Schema
204	The subscription resource was deleted successfully. The response body shall be empty.	No Content

Definitions

CreateNsdInfoRequest

This type creates a completely new NS descriptor resource.

Name	Description	Schema
userDefinedData *optional*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object

CreatePnfdInfoRequest

This type creates a new PNF descriptor resource.

Name	Description	Schema
userDefinedData *optional*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object

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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].

Type : object

Link

This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.

Type : object

NsdChangeNotification

This type represents an NSD management notification, which informs the receiver of a change of the "nsdOperationalState" attribute of an on-boarded NSD. Changes in the value of the "nsdUsageState" and "nsdOnboardingState" attributes are not reported. The notification shall comply with the provisions defined in Table 5.5.2.11-1 of GS NFV-SOL 005. The support of this notification is mandatory. The notification shall be triggered by the NFVO when the value of the "nsdOperationalState" attribute has changed, and the "nsdOperationalState" attribute has the value "ONBOARDED".

Name	Description	Schema
_links *required*	This type represents the links to resources that an NSD management notification can contain.	_links
id *required*	Identifier of this notification. If a notification is sent multiple times due to multiple subscriptions, the "id" attribute of all these notifications shall have the same value.	string
notificationType *required*	Discriminator for the different notification types. Shall be set to "NsdChangeNotification" for this notification type.	string
nsdId *required*	This identifier, which is managed by the service provider, identifies the NSD in a globally unique way. It is copied from the on-boarded NSD.	string
nsdInfoId *required*	Identifier of the NSD information object. This identifier is allocated by the NFVO.	string

Name	Description	Schema
nsdOperationalState *required*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
subscriptionId *required*	Identifier of the subscription that this notification relates to.	string
timeStamp *required*	Date-time of the generation of the notification.	string (date-time)

_links

Name	Description	Schema
nsdInfo *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
subscription *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

NsdDeletionNotification

This type represents an NSD management notification, which informs the receiver of the deletion of an on-boarded NSD. The notification shall comply with the provisions defined in Table 5.5.2.12-1. The support of this notification is mandatory. The notification shall be triggered by the NFVO when it has deleted an on-boarded NSD.

Name	Description	Schema
_links *required*	This type represents the links to resources that an NSD management notification can contain.	_links
id *required*	Identifier of this notification. If a notification is sent multiple times due to multiple subscriptions, the "id" attribute of all these notifications shall have the same value.	string

Name	Description	Schema
notificationType *required*	Discriminator for the different notification types. Shall be set to "NsdDeletionNotification " for this notification type.	string
nsdId *required*	This identifier, which is managed by the service provider, identifies the NSD in a globally unique way. It is copied from the on-boarded NSD.	string
nsdInfoId *required*	Identifier of the NSD information object. This identifier is allocated by the NFVO.	string
subscriptionId *required*	Identifier of the subscription that this notification relates to.	string
timeStamp *required*	Date-time of the generation of the notification.	string (date-time)

_links

Name	Description	Schema
nsdInfo *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
subscription *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

NsdInfo

This type represents a response for the query NSD operation. It shall comply with the provisions defined in Table 5.5.2.2-1 of GS NFV-SOL 005.

Name	Description	Schema
_links *required*	Links to resources related to this resource.	_links
id *required*	Identifier of the onboarded individual NS descriptor resource. This identifier is allocated by the NFVO.	string

Name	Description	Schema
nestedNsdInfoIds *optional*	Identifies the NsdInfo element for the nested NSD referenced by the on-boarded NS descriptor resource. At least one of the attributes – vnfPkgId and nestedNsdInfoId shall be present, after the NSD is on-boarded.	string
nsdDesigner *optional*	Designer of the on-boarded NSD. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdId *optional*	This identifier, which is allocated by the NSD designer, identifies the NSD in a globally unique way. It is copied from the NSD content and shall be present after the NSD content is on-boarded.	object
nsdInvariantId *optional*	This identifier, which is allocated by the NSD designer, identifies an NSD in a version independent manner. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdName *optional*	Name of the onboarded NSD. This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
nsdOnboardingState *required*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *required*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)

Name	Description	Schema
nsdUsageState *required*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Version of the on-boarded NSD. The NSD version is a string of variable length.This information is copied from the NSD content and shall be present after the NSD content is on-boarded.	string
onboardingFailureDetails *optional*	The definition of the general "ProblemDetails" data structure from IETF RFC 7807 [27] is reproduced in Table 4.3.5.3-1 of GS NFV-SOL 005. Compared to the general framework defined in IETF RFC 7807 [27], the "status" and "detail" attributes are mandated to be included by the present document, to ensure that the response contains additional textual information about an error. IETF RFC 7807 [27] foresees extensibility of the "ProblemDetails" type. It is possible that particular APIs in the present document, or particular implementations, define extensions to define additional attributes that provide more information about the error. The description column only provides some explanation of the meaning to facilitate understanding of the design. For a full description, see IETF RFC 7807 [27].	onboardingFailureDetails
pnfdInfoIds *optional*	Identifies the PnfdInfo element for the PNFD referenced by the on-boarded NS descriptor resource.	string
userDefinedData *optional*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object
vnfPkgIds *optional*	Identifies the VNF package for the VNFD referenced by the on-boarded NS descriptor resource.	string

_links

Name	Description	Schema
nsd_content *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
self *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

onboardingFailureDetails

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

NsdInfoModifications

This type represents attribute modifications for an individual NS descriptor resource based on the "NsdInfo" data type. The attributes of "NsdInfo" that can be modified are included in the "NsdInfoModifications" data type.

NOTE	At least one of the attributes - nsdOperationalState and userDefinedData - shall be present.

Name	Description	Schema
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
userDefinedData *optional*	Modifications of the "userDefinedData" attribute in "NsdInfo" data type. See note. If present, these modifications shall be applied according to the rules of JSON Merge PATCH (see IETF RFC 7396 [25]). NOTE: At least one of the attributes - nsdOperationalState and userDefinedData - shall be present.	object

NsdOnboardingFailureNotification

This type represents an NSD management notification, which informs the receiver of the failure of on-boarding an NSD. It shall comply with the provisions defined in Table 5.5.2.10-1. The support of this notification is mandatory. The notification shall be triggered by the NFVO when the on-boarding of an NSD has failed.

Name	Description	Schema
_links *required*	This type represents the links to resources that an NSD management notification can contain.	_links
id *required*	Identifier of this notification. If a notification is sent multiple times due to multiple subscriptions, the "id" attribute of all these notifications shall have the same value.	string
notificationType *required*	Discriminator for the different notification types. Shall be set to "NsdOnboardingFailureNotification" for this notification type.	string
nsdId *optional*	This identifier, which is managed by the service provider, identifies the NSD in a globally unique way.	string
nsdInfoId *required*	Identifier of the NSD information object. This identifier is allocated by the NFVO.	string

Name	Description	Schema
onboardingFailureDetails *required*	The definition of the general "ProblemDetails" data structure from IETF RFC 7807 [27] is reproduced in Table 4.3.5.3-1 of GS NFV-SOL 005. Compared to the general framework defined in IETF RFC 7807 [27], the "status" and "detail" attributes are mandated to be included by the present document, to ensure that the response contains additional textual information about an error. IETF RFC 7807 [27] foresees extensibility of the "ProblemDetails" type. It is possible that particular APIs in the present document, or particular implementations, define extensions to define additional attributes that provide more information about the error. The description column only provides some explanation of the meaning to facilitate understanding of the design. For a full description, see IETF RFC 7807 [27].	onboardingFailureDetails
subscriptionId *optional*	Identifier of the subscription that this notification relates to.	string
timeStamp *required*	Date-time of the generation of the notification.	string (date-time)

_links

Name	Description	Schema
nsdInfo *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
subscription *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

onboardingFailureDetails

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object

Name	Description	Schema
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

NsdOnboardingNotification

This type represents an NSD management notification, which informs the receiver of the successful on-boarding of an NSD. It shall comply with the provisions defined in Table 5.5.2.9-1 of GS NFV-SOL 005. The support of this notification is mandatory. The notification shall be triggered by the NFVO when the "nsdOnboardingState" attribute of a new NSD has changed to "ONBOARDED".

Name	Description	Schema
_links *required*	This type represents the links to resources that an NSD management notification can contain.	_links
id *required*	Identifier of this notification. If a notification is sent multiple times due to multiple subscriptions, the "id" attribute of all these notifications shall have the same value.	string
notificationType *required*	Discriminator for the different notification types. Shall be set to "NsdOnboardingNotification" for this notification type.	string
nsdId *required*	This identifier, which is managed by the service provider, identifies the NSD in a globally unique way. It is copied from the on-boarded NSD.	string
nsdInfoId *required*	Identifier of the NSD information object. This identifier is allocated by the NFVO.	string
subscriptionId *required*	Identifier of the subscription that this notification relates to.	string

Name	Description	Schema
timeStamp *required*	Date-time of the generation of the notification.	string (date-time)

_links

Name	Description	Schema
nsdInfo *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
subscription *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

NsdOnboardingStateType

The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD.

CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.

Type : enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)

NsdOperationalStateType

The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource.

ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.

Type : enum (ENABLED, DISABLED)

NsdUsageStateType

The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource.

IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.

Type : enum (IN_USE, NOT_IN_USE)

NsdmLinks

This type represents the links to resources that an NSD management notification can contain.

Name	Description	Schema
nsdInfo *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
subscription *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

NsdmNotificationsFilter

This type represents a subscription filter related to notifications about NSD management. It shall comply with the provisions defined in Table 5.5.3.2-1 of GS NFV-SOL 005. At a particular nesting level in the filter structure, the following applies: All attributes shall match in order for the filter to match (logical "and" between different filter attributes). If an attribute is an array, the attribute shall match if at least one of the values in the array matches (logical "or" between the values of one filter attribute).

Name	Description	Schema
nestedNsdInfoIds *optional*	Match the NsdInfo identifier for the nested NSD referenced by the on-boarded NSD.	string
notificationTypes *optional*	Match particular notification types. Permitted values: NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification. The permitted values of the "notificationTypes" attribute are spelled exactly as the names of the notification types to facilitate automated code generation systems.	enum (NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification, PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification)

Name	Description	Schema
nsdDesigner *optional*	Match the NSD designer of the on-boarded NSD.	string
nsdId *optional*	Match the NSD identifier, which is allocated by the NSD designer. The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInfoId *optional*	Match the NsdInfo identifier which is allocated by the NFVO. Note: The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInvariantId *optional*	Match the NSD invariant identifier which is allocated by the NSD designer and identifies an NSD in a version independent manner.	string
nsdName *optional*	Match the name of the onboarded NSD.	string
nsdOnboardingState *optional*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)

Name	Description	Schema
nsdUsageState *optional*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Match the NSD version listed as part of this attribute. The NSD version is a string of variable length.	string
pnfdId *optional*	Match the PNFD identifier which is copied from the PNFD content. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInfoIds *optional*	Match the PnfdInfo identifier for the PNFD referenced by the on-boarded NSD. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInvariantId *optional*	Match the PNFD in a version independent manner.	string
pnfdName *optional*	Match the name of the on-boarded PNFD.	string
pnfdOnboardingState *optional*	The enumeration PnfdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.6-1 of GS-NFV SOL005. It indicates the onboarding state of the individual PNF descriptor resource. CREATED = The PNF descriptor resource is created. UPLOADING = The associated PNFD content is being uploaded. PROCESSING = The associated PNFD content is being processed, e.g. validation. ONBOARDED = The associated PNFD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDING)
pnfdProvider *optional*	Match the provider of the on-boarded PNFD.	string

Name	Description	Schema
pnfdUsageState *optional*	The enumeration PnfdUsageStateType shall comply with the provisions defined in Table 5.5.4.7-1 of GS NFV-SOL005. It indicates the usage state of the resource. IN-USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
pnfdVersion *optional*	Match the PNFD designer of the on-boarded PNFD. The PNFD version is a string of variable length.	string
vnfPkgIds *optional*	Match VNF packages with a package identifier listed in the attribute.	string

NsdmSubscription

This type represents a subscription related to notifications about NSD management.

Name	Description	Schema
_links *required*	Links to resources related to this resource.	_links
callbackUri *required*	String formatted according to IETF RFC 3986 [10].	object
filter *optional*	This type represents a subscription filter related to notifications about NSD management. It shall comply with the provisions defined in Table 5.5.3.2-1 of GS NFV-SOL 005. At a particular nesting level in the filter structure, the following applies: All attributes shall match in order for the filter to match (logical "and" between different filter attributes). If an attribute is an array, the attribute shall match if at least one of the values in the array matches (logical "or" between the values of one filter attribute).	filter
id *required*	Identifier of this subscription resource	string

_links

Name	Description	Schema
self *optional*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

filter

Name	Description	Schema
nestedNsdInfoIds *optional*	Match the NsdInfo identifier for the nested NSD referenced by the on-boarded NSD.	string
notificationTypes *optional*	Match particular notification types. Permitted values: NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification. The permitted values of the "notificationTypes" attribute are spelled exactly as the names of the notification types to facilitate automated code generation systems.	enum (NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification, PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification)
nsdDesigner *optional*	Match the NSD designer of the on-boarded NSD.	string
nsdId *optional*	Match the NSD identifier, which is allocated by the NSD designer. The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInfoId *optional*	Match the NsdInfo identifier which is allocated by the NFVO. Note: The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string

Name	Description	Schema
nsdInvariantId *optional*	Match the NSD invariant identifier which is allocated by the NSD designer and identifies an NSD in a version independent manner.	string
nsdName *optional*	Match the name of the onboarded NSD.	string
nsdOnboardingState *optional*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
nsdUsageState *optional*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Match the NSD version listed as part of this attribute. The NSD version is a string of variable length.	string
pnfdId *optional*	Match the PNFD identifier which is copied from the PNFD content. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string

Name	Description	Schema
pnfdInfoIds *optional*	Match the PnfdInfo identifier for the PNFD referenced by the on-boarded NSD. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInvariantId *optional*	Match the PNFD in a version independent manner.	string
pnfdName *optional*	Match the name of the on-boarded PNFD.	string
pnfdOnboardingState *optional*	The enumeration PnfdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.6-1 of GS-NFV SOL005. It indicates the onboarding state of the individual PNF descriptor resource. CREATED = The PNF descriptor resource is created. UPLOADING = The associated PNFD content is being uploaded. PROCESSING = The associated PNFD content is being processed, e.g. validation. ONBOARDED = The associated PNFD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDING)
pnfdProvider *optional*	Match the provider of the on-boarded PNFD.	string
pnfdUsageState *optional*	The enumeration PnfdUsageStateType shall comply with the provisions defined in Table 5.5.4.7-1 of GS NFV-SOL005. It indicates the usage state of the resource. IN-USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
pnfdVersion *optional*	Match the PNFD designer of the on-boarded PNFD. The PNFD version is a string of variable length.	string
vnfPkgIds *optional*	Match VNF packages with a package identifier listed in the attribute.	string

NsdmSubscriptionRequest

This type represents a subscription request related to notifications about NSD management.

Name	Description	Schema
authentication *optional*		authentication
callbackUri *required*	The URI of the endpoint to send the notification to.	string
filter *optional*	This type represents a subscription filter related to notifications about NSD management. It shall comply with the provisions defined in Table 5.5.3.2-1 of GS NFV-SOL 005. At a particular nesting level in the filter structure, the following applies: All attributes shall match in order for the filter to match (logical "and" between different filter attributes). If an attribute is an array, the attribute shall match if at least one of the values in the array matches (logical "or" between the values of one filter attribute).	filter

authentication

Name	Description	Schema
authType *optional*	Defines the types of Authentication/ Authorization the API consumer is willing to accept when receiving a notification. Permitted values: BASIC: In every HTTP request to the notification endpoint, use HTTP Basic authentication with the client credentials. OAUTH2_CLIENT_CREDENTIALS: In every HTTP request to the notification endpoint, use an OAuth 2.0 Bearer token, obtained using the client credentials grant type. TLS_CERT: Every HTTP request to the notification endpoint is sent over a mutually authenticated TLS session. i.e. not only server is authenticated, but also the client is authenticated during the TLS tunnel setup.	enum (BASIC, OAUTH2_CLIENT_CREDENTIALS, TLS_CERT)
paramsBasic *optional*	Parameters for authentication/authorization using BASIC. Shall be present if authType is "BASIC" and the contained information has not been provisioned out of band. Shall be absent otherwise.	paramsBasic

Name	Description	Schema
paramsOauth2ClientCredentials *optional*	Parameters for authentication/authorization using OAUTH2_CLIENT_CREDENTIALS. Shall be present if authType is "OAUTH2_CLIENT_CREDENTIALS" and the contained information has not been provisioned out of band. Shall be absent otherwise.	paramsOauth2ClientCredentials

paramsBasic

Name	Description	Schema
password *optional*	Password to be used in HTTP Basic authentication. Shall be present if it has not been provisioned out of band.	string
userName *optional*	Username to be used in HTTP Basic authentication. Shall be present if it has not been provisioned out of band.	string

paramsOauth2ClientCredentials

Name	Description	Schema
clientId *optional*	Client identifier to be used in the access token request of the OAuth 2.0 client credentials grant type. Shall be present if it has not been provisioned out of band.	string

filter

Name	Description	Schema
nestedNsdInfoIds *optional*	Match the NsdInfo identifier for the nested NSD referenced by the on-boarded NSD.	string

Name	Description	Schema
notificationTypes *optional*	Match particular notification types. Permitted values: NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification. The permitted values of the "notificationTypes" attribute are spelled exactly as the names of the notification types to facilitate automated code generation systems.	enum (NsdOnBoardingNotification, NsdOnboardingFailureNotification, NsdChangeNotification, NsdDeletionNotification, PnfdOnBoardingNotification, PnfdOnBoardingFailureNotification, PnfdDeletionNotification)
nsdDesigner *optional*	Match the NSD designer of the on-boarded NSD.	string
nsdId *optional*	Match the NSD identifier, which is allocated by the NSD designer. The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInfoId *optional*	Match the NsdInfo identifier which is allocated by the NFVO. Note: The attributes "nsdId" and "nsdInfoId" are alternatives to reference to a particular NSD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
nsdInvariantId *optional*	Match the NSD invariant identifier which is allocated by the NSD designer and identifies an NSD in a version independent manner.	string
nsdName *optional*	Match the name of the onboarded NSD.	string

Name	Description	Schema
nsdOnboardingState *optional*	The enumeration NsdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.5-1 of GS NFV-SOL 005. It indicates the onboarding state of the NSD. CREATED = The NSD information object is created. UPLOADING = The associated NSD content is being uploaded. PROCESSING = The associated NSD content is being processed, e.g. validation. ONBOARDED = The associated NSD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDED)
nsdOperationalState *optional*	The enumeration NsdOperationalStateType shall comply with the provisions defined in Table 5.5.4.3-1 of GS NFV_SOL 005. It indicates the operational state of the resource. ENABLED = The operational state of the resource is enabled. DISABLED = The operational state of the resource is disabled.	enum (ENABLED, DISABLED)
nsdUsageState *optional*	The enumeration NsdUsageStateType shall comply with the provisions defined in Table 5.5.4.4-1 of GS NFV-SOL 005. It indicates the usage state of the resource. IN_USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
nsdVersion *optional*	Match the NSD version listed as part of this attribute. The NSD version is a string of variable length.	string
pnfdId *optional*	Match the PNFD identifier which is copied from the PNFD content. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInfoIds *optional*	Match the PnfdInfo identifier for the PNFD referenced by the on-boarded NSD. The attributes "pnfdId" and "pnfdInfoId" are alternatives to reference to a particular PNFD in a filter. They should not be used both in the same filter instance, but one alternative should be chosen.	string
pnfdInvariantId *optional*	Match the PNFD in a version independent manner.	string

Name	Description	Schema
pnfdName *optional*	Match the name of the on-boarded PNFD.	string
pnfdOnboardingState *optional*	The enumeration PnfdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.6-1 of GS-NFV SOL005. It indicates the onboarding state of the individual PNF descriptor resource. CREATED = The PNF descriptor resource is created. UPLOADING = The associated PNFD content is being uploaded. PROCESSING = The associated PNFD content is being processed, e.g. validation. ONBOARDED = The associated PNFD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDING)
pnfdProvider *optional*	Match the provider of the on-boarded PNFD.	string
pnfdUsageState *optional*	The enumeration PnfdUsageStateType shall comply with the provisions defined in Table 5.5.4.7-1 of GS NFV-SOL005. It indicates the usage state of the resource. IN-USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
pnfdVersion *optional*	Match the PNFD designer of the on-boarded PNFD. The PNFD version is a string of variable length.	string
vnfPkgIds *optional*	Match VNF packages with a package identifier listed in the attribute.	string

PnfdDeletionNotification

This type represents a PNFD management notification, which informs the receiver of the deletion of an on-boarded PNFD. The notification shall comply with the provisions defined in Table 5.5.2.15-1. The support of this notification is mandatory. The notification is triggered when an on-boarded PNFD is deleted.

Name	Description	Schema
_links *required*	This type represents the links to resources that a PNFD management notification can contain.	_links

Name	Description	Schema
id *required*	Identifier of this notification. If a notification is sent multiple times due to multiple subscriptions, the "id" attribute of all these notifications shall have the same value.	string
notificationType *required*	Discriminator for the different notification types. Shall be set to "PnfdDeletionNotification " for this notification type.	string
pnfdId *required*	This identifier, which is managed by the service provider, identifies the PNFD in a globally unique way. It is copied from the on-boarded PNFD.	string
pnfdInfoId *required*	Identifier of the PNFD information object. This identifier is allocated by the NFVO.	string
subscriptionId *required*	Identifier of this notification. If a notification is sent multiple times due to multiple subscriptions, the "id" attribute of all these notifications shall have the same value.	string
timeStamp *required*	Date-time of the generation of the notification.	string (date-time)

_links

Name	Description	Schema
pnfdInfo *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
subscription *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

PnfdInfo

This type represents a response for the query PNFD operation.

Name	Description	Schema
_links *required*	Links to resources related to this resource.	_links

Name	Description	Schema
id *required*	Identifier of the onboarded individual PNF descriptor resource. This identifier is allocated by the NFVO.	string
onboardingFailureDetails *optional*	The definition of the general "ProblemDetails" data structure from IETF RFC 7807 [27] is reproduced in Table 4.3.5.3-1 of GS NFV-SOL 005. Compared to the general framework defined in IETF RFC 7807 [27], the "status" and "detail" attributes are mandated to be included by the present document, to ensure that the response contains additional textual information about an error. IETF RFC 7807 [27] foresees extensibility of the "ProblemDetails" type. It is possible that particular APIs in the present document, or particular implementations, define extensions to define additional attributes that provide more information about the error. The description column only provides some explanation of the meaning to facilitate understanding of the design. For a full description, see IETF RFC 7807 [27].	onboardingFailureDetails
pnfdId *optional*	This identifier, which is managed by the PNFD designer, identifies the PNFD in a globally unique way. It is copied from the PNFD content and shall be present after the PNFD content is on-boarded.	string
pnfdInvariantId *optional*	Identifies a PNFD in a version independent manner. This attribute is invariant across versions of PNFD.	string
pnfdName *optional*	Name of the onboarded PNFD. This information is copied from the PNFD content and shall be present after the PNFD content is on-boarded.	string
pnfdOnboardingState *required*	The enumeration PnfdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.6-1 of GS-NFV SOL005. It indicates the onboarding state of the individual PNF descriptor resource. CREATED = The PNF descriptor resource is created. UPLOADING = The associated PNFD content is being uploaded. PROCESSING = The associated PNFD content is being processed, e.g. validation. ONBOARDED = The associated PNFD content is on-boarded.	enum (CREATED, UPLOADING, PROCESSING, ONBOARDING)

Name	Description	Schema
pnfdProvider *optional*	Provider of the onboarded PNFD. This information is copied from the PNFD content and shall be present after the PNFD content is onboarded.	string
pnfdUsageState *required*	The enumeration PnfdUsageStateType shall comply with the provisions defined in Table 5.5.4.7-1 of GS NFV-SOL005. It indicates the usage state of the resource. IN-USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.	enum (IN_USE, NOT_IN_USE)
pnfdVersion *optional*	Version of the onboarded PNFD. The PNFD version is a string of variable length.This information is copied from the PNFD content and shall be present after the PNFD content is on-boarded.	string
userDefinedData *optional*	User defined data for the individual PNF descriptor resource. This attribute can be modified with the PATCH method.	object

_links

Name	Description	Schema
pnfd_content *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
self *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

onboardingFailureDetails

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer

Name	Description	Schema
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

PnfdInfoModifications

This type represents attribute modifications for an individual PNF descriptor resource based on the "PnfdInfo" data type. The attributes of "PnfdInfo" that can be modified are included in the "PnfdInfoModifications" data type.

Name	Description	Schema
userDefinedData *required*	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 keyvalue pairs is represented as an object. It shall comply with the provisions defined in clause 4 of IETF RFC 7159 [20].	object

PnfdOnboardingFailureNotification

This type represents a PNFD management notification, which informs the receiver of the failure of on-boarding a PNFD. It shall comply with the provisions defined in Table 5.5.2.14-1 of GS NFV-SOL 005. The support of this notification is mandatory. The notification is triggered when the on-boarding of a PNFD fails.

Name	Description	Schema
_links *required*	This type represents the links to resources that a PNFD management notification can contain.	_links
id *required*	Identifier of this notification. If a notification is sent multiple times due to multiple subscriptions, the "id" attribute of all these notifications shall have the same value.	string
notificationType *required*	Discriminator for the different notification types. Shall be set to "PnfdOnboardingFailureNotification" for this notification type.	string

Name	Description	Schema
onboardingFailureDetails *required*	The definition of the general "ProblemDetails" data structure from IETF RFC 7807 [27] is reproduced in Table 4.3.5.3-1 of GS NFV-SOL 005. Compared to the general framework defined in IETF RFC 7807 [27], the "status" and "detail" attributes are mandated to be included by the present document, to ensure that the response contains additional textual information about an error. IETF RFC 7807 [27] foresees extensibility of the "ProblemDetails" type. It is possible that particular APIs in the present document, or particular implementations, define extensions to define additional attributes that provide more information about the error. The description column only provides some explanation of the meaning to facilitate understanding of the design. For a full description, see IETF RFC 7807 [27].	onboardingFailureDetails
pnfdId *optional*	This identifier, which is managed by the service provider, identifies the PNFD in a globally unique way.	string
pnfdInfoId *required*	Identifier of the PNFD information object. This identifier is allocated by the NFVO.	string
subscriptionId *required*	Identifier of the subscription that this notification relates to.	string
timeStamp *required*	Date-time of the generation of the notification.	string (date-time)

_links

Name	Description	Schema
pnfdInfo *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
subscription *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

onboardingFailureDetails

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

PnfdOnboardingNotification

This type represents a PNFD management notification, which informs the receiver of the successful on-boarding of a PNFD. It shall comply with the provisions defined in Table 5.5.2.13-1. The support of this notification is mandatory. The notification is triggered when a new PNFD is on-boarded.

Name	Description	Schema
_links *required*	This type represents the links to resources that a PNFD management notification can contain.	_links
id *required*	Identifier of this notification. If a notification is sent multiple times due to multiple subscriptions, the "id" attribute of all these notifications shall have the same value.	string
notificationType *required*	Discriminator for the different notification types. Shall be set to "PnfdOnboardingNotification" for this notification type.	string
pnfdId *required*	This identifier, which is managed by the service provider, identifies the PNFD in a globally unique way. It is copied from the on-boarded PNFD.	string

Name	Description	Schema
pnfdInfoId *required*	Identifier of the PNFD information object. This identifier is allocated by the NFVO.	string
subscriptionId *required*	Identifier of the subscription that this notification relates to.	string
timeStamp *required*	Date-time of the generation of the notification.	string (date-time)

_links

Name	Description	Schema
pnfdInfo *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
subscription *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

PnfdOnboardingStateType

The enumeration PnfdOnboardingStateType shall comply with the provisions defined in Table 5.5.4.6-1 of GS-NFV SOL005. It indicates the onboarding state of the individual PNF descriptor resource.

CREATED = The PNF descriptor resource is created. UPLOADING = The associated PNFD content is being uploaded. PROCESSING = The associated PNFD content is being processed, e.g. validation. ONBOARDED = The associated PNFD content is on-boarded.

Type : enum (CREATED, UPLOADING, PROCESSING, ONBOARDING)

PnfdUsageStateType

The enumeration PnfdUsageStateType shall comply with the provisions defined in Table 5.5.4.7-1 of GS NFV-SOL005. It indicates the usage state of the resource.

IN-USE = The resource is in use. NOT_IN_USE = The resource is not-in-use.

Type : enum (IN_USE, NOT_IN_USE)

PnfdmLinks

This type represents the links to resources that a PNFD management notification can contain.

Name	Description	Schema
pnfdInfo *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object
subscription *required*	This type represents a link to a resource. It shall comply with the provisions defined in Table 4.4.1.3-1 of GS NFV-SOL 005.	object

ProblemDetails

The definition of the general "ProblemDetails" data structure from IETF RFC 7807 [27] is reproduced in Table 4.3.5.3-1 of GS NFV-SOL 005. Compared to the general framework defined in IETF RFC 7807 [27], the "status" and "detail" attributes are mandated to be included by the present document, to ensure that the response contains additional textual information about an error. IETF RFC 7807 [27] foresees extensibility of the "ProblemDetails" type. It is possible that particular APIs in the present document, or particular implementations, define extensions to define additional attributes that provide more information about the error.

The description column only provides some explanation of the meaning to facilitate understanding of the design. For a full description, see IETF RFC 7807 [27].

Name	Description	Schema
detail *required*	A human-readable explanation specific to this occurrence of the problem.	string
instance *optional*	String formatted according to IETF RFC 3986 [10].	object
status *required*	The HTTP status code for this occurrence of the problem.	integer
title *optional*	A short, human-readable summary of the problem type. It should not change from occurrence to occurrence of the problem, except for purposes of localization. If type is given and other than "about:blank", this attribute shall also be provided.	string
type *optional*	String formatted according to IETF RFC 3986 [10].	object

SubscriptionAuthentication

Name	Description	Schema
authType *optional*	Defines the types of Authentication/ Authorization the API consumer is willing to accept when receiving a notification. Permitted values: BASIC: In every HTTP request to the notification endpoint, use HTTP Basic authentication with the client credentials. OAUTH2_CLIENT_CREDENTIALS: In every HTTP request to the notification endpoint, use an OAuth 2.0 Bearer token, obtained using the client credentials grant type. TLS_CERT: Every HTTP request to the notification endpoint is sent over a mutually authenticated TLS session. i.e. not only server is authenticated, but also the client is authenticated during the TLS tunnel setup.	enum (BASIC, OAUTH2_CLIENT_CREDENTIALS, TLS_CERT)
paramsBasic *optional*	Parameters for authentication/authorization using BASIC. Shall be present if authType is "BASIC" and the contained information has not been provisioned out of band. Shall be absent otherwise.	paramsBasic
paramsOauth2ClientCredentials *optional*	Parameters for authentication/authorization using OAUTH2_CLIENT_CREDENTIALS. Shall be present if authType is "OAUTH2_CLIENT_CREDENTIALS" and the contained information has not been provisioned out of band. Shall be absent otherwise.	paramsOauth2ClientCredentials

paramsBasic

Name	Description	Schema
password *optional*	Password to be used in HTTP Basic authentication. Shall be present if it has not been provisioned out of band.	string
userName *optional*	Username to be used in HTTP Basic authentication. Shall be present if it has not been provisioned out of band.	string

paramsOauth2ClientCredentials

Name	Description	Schema
clientId *optional*	Client identifier to be used in the access token request of the OAuth 2.0 client credentials grant type. Shall be present if it has not been provisioned out of band.	string

Uri

String formatted according to IETF RFC 3986 [10].

Type : object