**3GPP TSG-CT WG4 Meeting #115eC4-231xyz**

**E-Meeting, 17th– 21st April 2023 (revision of C4-231315)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.2* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **29.510** | **CR** | **0833** | **rev** | **1** | **Current version:** | **18.2.0** |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network |  | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | Content of Profile Update response messages | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Ericsson | | | | | | | | | |
| ***Source to TSG:*** | CT4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | SBIProtoc18 | | | | |  | ***Date:*** | | | 2023-04-03 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **F** |  | | | | | ***Release:*** | | | Rel-18 |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) … Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18) Rel-19 (Release 19)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | The NF Profile Partial Update service operation does not include the "204 No Content" response in the list of possible responses from NRF. However, this is a typical response, and most efficient, in most other APIs in 5GC, when all requested changes in the PATCH request are accepted by the receiver.  Also, the Heart-Beat service operation, which is implemented by means of the same HTTP operation on the same resource, does contain the "204 No Content" as a possible response. This misalignment is confusing and causes inefficiencies.  Last, the procedure described in Annex B for "NF Profile changes in NFRegister and NFUpdate" only covers the NFProfile Complete Replacemente, while it would be beneficial to also use the same mechanism for the NFProfile Partial Update. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | - Include the "204 No Content" as a possible response to the NF Profile Partial Update in the description of the service operation (note that this does not require changes to OpenAPI since this is already allowed in the responses of the PATCH operation)  - Include the possibility to support the feature in Annex B, also for NFProfile Partial Update | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | The signaling between NRF consumers and the NRF is not optimal, and it causes excessive message size in the responses from NRF. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 5.2.2.3.1, 6.1.3.3.3.3, 6.1.6.2.2, A.2, Annex B | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | | This CR does not introduce any changes on the OpenAPI description files. | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

\* \* \* First Change \* \* \* \*

##### 5.2.2.3.1 General

This service operation updates the profile of a Network Function previously registered in the NRF by providing the updated NF profile of the requesting NF to the NRF. The update operation may apply to the whole profile of the NF (complete replacement of the existing profile by a new profile), or it may apply only to a subset of the parameters of the profile (including adding/deleting/replacing services to the NF profile).

To perform a complete replacement of the NF Profile of a given NF Instance, the NF Service Consumer shall issue an HTTP PUT request, as shown in Figure 5.2.2.3.1-1:



Figure 5.2.2.3.1-1: NF Profile Complete Replacement

1. The NF Service Consumer shall send a PUT request to the resource URI representing the NF Instance. The payload body of the PUT request shall contain a representation of the NF Instance to be completely replaced in the NRF.

2a. On success, "200 OK" shall be returned, the payload body of the PUT response shall contain the representation of the replaced resource. The representation of the replaced resource may be a complete NF Profile or a NF Profile just including the mandatory attributes of the NF Profile and the attributes which the NRF added or changed (see Annex B).

2b. On failure or redirection:

- If the update of the NF instance fails at the NRF due to errors in the encoding of the NFProfile JSON object, the NRF shall return "400 Bad Request" status code with the ProblemDetails IE providing details of the error.

- If the update of the NF instance fails at the NRF due to NRF internal errors, the NRF shall return "500 Internal Server Error" status code with the ProblemDetails IE providing details of the error.

- In the case of redirection, the NRF shall return 3xx status code, which shall contain a Location header with an URI pointing to the endpoint of another NRF service instance.

To perform a partial update of the NF Profile of a given NF Instance, the NF Service Consumer shall issue an HTTP PATCH request, as shown in Figure 5.2.2.3.1-2. This partial update shall be used to add/delete/replace individual parameters of the NF Instance, and also to add/delete/replace any of the services (and their parameters) offered by the NF Instance.



Figure 5.2.2.3.1-2: NF Profile Partial Update

1. The NF Service Consumer shall send a PATCH request to the resource URI representing the NF Instance. The payload body of the PATCH request shall contain the list of operations (add/delete/replace) to be applied to the NF Profile of the NF Instance; these operations may be directed to individual parameters of the NF Profile or to the list of services (and their parameters) offered by the NF Instances. In order to leave the NF Profile in a consistent state, all the operations specified by the PATCH request body shall be executed atomically.

The NF Service Consumer should include a "If-Match" HTTP header carrying the latest entity-tag received form NRF for the NF profile to which the PATCH document shall be applied.

2a. On success, if all update operations are accepted by the NRF, "204 No Content" should be returned; the NRF may instead return "200 OK" with the payload body of the PATCH response containing the representation of the replaced resource.

2b. On failure or redirection:

- If the NF Instance, identified by the "nfInstanceID", is not found in the list of registered NF Instances in the NRF's database, the NRF shall return "404 Not Found" status code with the ProblemDetails IE providing details of the error.

- In the case of redirection, the NRF shall return 3xx status code, which shall contain a Location header with an URI pointing to the endpoint of another NRF service instance.

- If "If-Match" header is received with an entity tag different from the entity-tag in NRF for NF profile of the target NF instance, the NRF shall return "412 Precondition Failed" status code with the ProblemDetails IE providing details of the error.

- If no precondition was defined in the request and another confliction has been detected (e.g. to change value of a non-existing IE), the NRF shall return "409 Conflicting" status code with the ProblemDetails IE providing details of the error.

The NRF shall allow updating Vendor-Specific attributes (see 3GPP TS 29.500 [4], clause 6.6.3) that may exist in the NF Profile of a registered NF Instance.

\* \* \* Next Change \* \* \* \*

6.1.3.3.3.3 PATCH

This method updates partially the profile of a given NF instance.

This method shall support the URI query parameters specified in table 6.1.3.3.3.3-1.

Table 6.1.3.3.3.3-1: URI query parameters supported by the PATCH method on this resource

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Data type | P | Cardinality | Description |
| n/a |  |  |  |  |

This method shall support the request data structures specified in table 6.1.3.3.3.3-2 and the response data structures and response codes specified in table 6.1.3.3.3.3-3.

Table 6.1.3.3.3.3-2: Data structures supported by the PATCH Request Body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Data type | P | Cardinality | Description |
| array(PatchItem) | M | 1 | It contains the list of changes to be made to the profile of the NF Instance, according to the JSON PATCH format specified in IETF RFC 6902 [13]. |

Table 6.1.3.3.3.3-3: Data structures supported by the PATCH Response Body on this resource

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Data type | P | Cardinality | Response  codes | Description |
| NFProfile | M | 1 | 200 OK | Upon success, a response body is returned containing the updated profile of the NF Instance. |
| n/a |  |  | 204 No Content | Successful response sent when there is no need to provide a full updated profile of the NF Instance (e.g., in the partial update procedure when all update operations are accepted by the NRF, as described in clause 5.2.2.3.1, or in the Heart-Beat operation response described in clause 5.2.2.3.2). |
| RedirectResponse | O | 0..1 | 307 Temporary Redirect | The NRF shall generate a Location header field containing a URI pointing to the endpoint of another NRF service instance to which the request should be sent.  If an SCP redirects the message to another SCP then the location header field shall contain the same URI or a different URI pointing to the endpoint of the NF service producer to which the request should be sent. |
| RedirectResponse | O | 0..1 | 308 Permanent Redirect | The NRF shall generate a Location header field containing a URI pointing to the endpoint of another NRF service instance to which the request should be sent.  If an SCP redirects the message to another SCP then the location header field shall contain the same URI or a different URI pointing to the endpoint of the NF service producer to which the request should be sent. |
| ProblemDetails | O | 0..1 | 412 Precondition Failed | The modification has failed due to the precondition in the request is not fulfilled. |
| ProblemDetails | O | 0..1 | 409 Conflict | The modification has failed due to confliction (e.g. to change a value of a non-existing IE). |
| NOTE: The mandatory HTTP error status codes for the PATCH method listed in Table 5.2.7.1-1 of 3GPP TS 29.500 [4] other than those specified in the table above also apply, with a ProblemDetails data type (see clause 5.2.7 of 3GPP TS 29.500 [4]). | | | | |

Table 6.1.3.3.3.3-4: Headers supported by the 307 Response Code on this resource

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Data type | P | Cardinality | Description |
| Location | string | M | 1 | A URI pointing to the endpoint of the NRF service instance to which the request should be sent |

Table 6.1.3.3.3.3-5: Headers supported by the 308 Response Code on this resource

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Data type | P | Cardinality | Description |
| Location | string | M | 1 | A URI pointing to the endpoint of the NRF service instance to which the request should be sent |

Table 6.1.3.3.3.3-6: Headers supported by the PATCH method on this resource

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Data type | P | Cardinality | Description |
| If-Match | string | C | 0..1 | Validator for conditional requests, as described in IETF RFC 7232 [19], clause 3.2. |

Table 6.1.3.3.3.3-7: Headers supported by the 200 Response Code on this resource

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Data type | P | Cardinality | Description |
| ETag | string | C | 0..1 | Entity Tag containing a strong validator, described in IETF RFC 7232 [19], clause 2.3. |

\* \* \* Next Change \* \* \* \*

##### 6.1.6.2.2 Type: NFProfile

Table 6.1.6.2.2-1: Definition of type NFProfile

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute name | Data type | P | Cardinality | Description |
| nfInstanceId | NfInstanceId | M | 1 | Unique identity of the NF Instance. |
| nfType | NFType | M | 1 | Type of Network Function |
| nfStatus | NFStatus | M | 1 | Status of the NF Instance (NOTE 5) (NOTE 16) |
| collocatedNfInstances | array(CollocatedNfInstance) | O | 1..N | Information related to collocated NF type(s) and corresponding NF Instances when the NF is collocated with NFs supporting other NF types.  (NOTE 21)  In this release of the specification, following collocation scenarios are supported (see clause 6.1.6.2.99): - a MB-SMF collocated with a SMF;  - a MB-UPF collocated with a UPF. |
| nfInstanceName | string | O | 0..1 | Human readable name of the NF Instance |
| heartBeatTimer | integer | C | 0..1 | Time in seconds expected between 2 consecutive heart-beat messages from an NF Instance to the NRF.  It may be included in the registration request. When present in the request it shall contain the heartbeat time proposed by the NF service consumer.  It shall be included in responses from NRF to registration requests (PUT) or in NF profile updates (PUT or PATCH). If the proposed heartbeat time is acceptable by the NRF based on the local configuration, it shall use the same value as in the registration request; otherwise the NRF shall override the value using a preconfigured value. |
| plmnList | array(PlmnId) | C | 1..N | PLMN(s) of the Network Function (NOTE 7).  This IE shall be present if this information is available for the NF.  If neither the plmnList IE nor the snpnList IE are provided, PLMN ID(s) of the PLMN of the NRF are assumed for the NF. |
| snpnList | array(PlmnIdNid) | C | 1..N | SNPN(s) of the Network Function.  This IE shall be present if the NF pertains to one or more SNPNs. |
| sNssais | array(ExtSnssai) | O | 1..N | S-NSSAIs of the Network Function.  If not provided, and if the perPlmnSnssaiList attribute is not present, the NF can serve any S-NSSAI.  When present this IE represents the list of S-NSSAIs supported in all the PLMNs listed in the plmnList IE.  If the sNSSAIs attribute is provided in at least one NF Service, the S-NSSAIs supported by the NF Profile shall be the set or a superset of the S-NSSAIs of the NFService(s). |
| perPlmnSnssaiList | array(PlmnSnssai) | O | 1..N | This IE may be included when the list of S-NSSAIs supported by the NF for each PLMN it is supporting is different. When present, this IE shall include the S-NSSAIs supported by the Network Function for each PLMN supported by the Network Function. When present, this IE shall override sNssais IE. (NOTE 9)  If the perPlmnSnssaiList attribute is provided in at least one NF Service, the S-NSSAIs supported per PLMN in the NF Profile shall be the set or a superset of the perPlmnSnssaiList of the NFService(s). |
| nsiList | array(string) | O | 1..N | NSI identities of the Network Function.  If not provided, the NF can serve any NSI. |
| fqdn | Fqdn | C | 0..1 | FQDN of the Network Function (NOTE 1) (NOTE 2) (NOTE 18). For AMF, the FQDN registered with the NRF shall be that of the AMF Name (see 3GPP TS 23.003 [12] clause 28.3.2.5). |
| interPlmnFqdn | Fqdn | C | 0..1 | If the NF needs to be discoverable by other NFs in a different PLMN, then an FQDN that is used for inter-PLMN routing as specified in 3GPP TS 23.003 [12] shall be registered with the NRF (NOTE 8).  A change of this attribute shall result in triggering a "NF\_PROFILE\_CHANGED" notification from NRF towards subscribing NFs located in the same or a different PLMN, but in the latter case the new value shall be notified as a change of the "fqdn" attribute. |
| ipv4Addresses | array(Ipv4Addr) | C | 1..N | IPv4 address(es) of the Network Function (NOTE 1) (NOTE 2) (NOTE 18) |
| ipv6Addresses | array(Ipv6Addr) | C | 1..N | IPv6 address(es) of the Network Function (NOTE 1) (NOTE 2) (NOTE 18) |
| allowedPlmns | array(PlmnId) | O | 1..N | PLMNs allowed to access the NF instance.  If not provided, any PLMN is allowed to access the NF.  This attribute shall not be included in profile change notifications to subscribed NFs, unless the subscribing entity explicitly requested so, in the "completeProfileSubscription" attribute in the subscription request message, and the NRF authorized such a request (see clauses 5.2.2.6.2 and 6.1.6.2.17). (NOTE 17) |
| allowedSnpns | array(PlmnIdNid) | O | 1..N | SNPNs allowed to access the NF instance.  If this attribute is present in the NFService and in the NF profile, the attribute from the NFService shall prevail.  The absence of this attribute in both the NFService and in the NF profile indicates that no SNPN, other than the SNPN(s) registered in the snpnList attribute of the NF Profile (if the NF pertains to an SNPN), is allowed to access the service instance.  This attribute shall not be included in profile change notifications to subscribed NFs, unless the subscribing entity explicitly requested so, in the "completeProfileSubscription" attribute in the subscription request message, and the NRF authorized such a request (see clauses 5.2.2.6.2 and 6.1.6.2.17). (NOTE 17) |
| allowedNfTypes | array(NFType) | O | 1..N | Type of the NFs allowed to access the NF instance.  If not provided, any NF type is allowed to access the NF.  This attribute shall not be included in profile change notifications to subscribed NFs, unless the subscribing entity explicitly requested so, in the "completeProfileSubscription" attribute in the subscription request message, and the NRF authorized such a request (see clauses 5.2.2.6.2 and 6.1.6.2.17). (NOTE 17) |
| allowedNfDomains | array(string) | O | 1..N | Pattern (regular expression according to the ECMA-262 dialect [8]) representing the NF domain names within the PLMN of the NRF allowed to access the NF instance.  If not provided, any NF domain is allowed to access the NF.  This attribute shall not be included in profile change notifications to subscribed NFs, unless the subscribing entity explicitly requested so, in the "completeProfileSubscription" attribute in the subscription request message, and the NRF authorized such a request (see clauses 5.2.2.6.2 and 6.1.6.2.17). (NOTE 17) |
| allowedNssais | array(ExtSnssai) | O | 1..N | S-NSSAI of the allowed slices to access the NF instance.  If not provided, any slice is allowed to access the NF.  This attribute shall not be included in profile change notifications to subscribed NFs, unless the subscribing entity explicitly requested so, in the "completeProfileSubscription" attribute in the subscription request message, and the NRF authorized such a request (see clauses 5.2.2.6.2 and 6.1.6.2.17). (NOTE 17) |
| priority | integer | O | 0..1 | Priority (relative to other NFs of the same type) within the range 0 to 65535, to be used for NF selection; lower values indicate a higher priority. Priority may or may not be present in the nfServiceList parameters, xxxInfo parameters and in this attribute. Priority in the nfServiceList has precedence over the priority in this attribute (NOTE 4).  Priority in xxxInfo parameter shall only be used to determine the relative priority among NF instances with the same priority at NFProfile/NFService.  The NRF may overwrite the received priority value when exposing an NFProfile with the Nnrf\_NFDiscovery service. |
| capacity | integer | O | 0..1 | Static capacity information within the range 0 to 65535, expressed as a weight relative to other NF instances of the same type; if capacity is also present in the nfServiceList parameters, those will have precedence over this value. (NOTE 4). |
| load | integer | O | 0..1 | Dynamic load information, within the range 0 to 100, indicates the current load percentage of the NF. |
| loadTimeStamp | DateTime | O | 0..1 | It indicates the point in time in which the latest load information (sent by the NF in the "load" attribute of the NF Profile) was generated at the NF Instance.  If the NF did not provide a timestamp, the NRF should set it to the instant when the NRF received the message where the NF provided the latest load information. |
| locality | string | O | 0..1 | Operator defined information about the location of the NF instance (e.g. geographic location, data center) (NOTE 3) |
| extLocality | map(string) | O | 1..N | Operator defined information about the location of the NF instance. (NOTE 3)  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters, representing a type of locality as defined in clause 6.1.6.3.x.  Example:  {  "DATA\_CENTER": "dc-123",  "CITY": "Los Angeles",  "STATE": "California"  } |
| udrInfo | UdrInfo | O | 0..1 | Specific data for the UDR (ranges of SUPI, group ID …) |
| udrInfoList | map(UdrInfo) | O | 1..N | Multiple entries of UdrInfo. This attribute provides additional information to the udrInfo. udrInfoList may be present even if the udrInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| udmInfo | UdmInfo | O | 0..1 | Specific data for the UDM (ranges of SUPI, group ID…) |
| udmInfoList | map(UdmInfo) | O | 1..N | Multiple entries of UdmInfo. This attribute provides additional information to the udmInfo. udmInfoList may be present even if the udmInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| ausfInfo | AusfInfo | O | 0..1 | Specific data for the AUSF (ranges of SUPI, group ID…) |
| ausfInfoList | map(AusfInfo) | O | 1..N | Multiple entries of AusfInfo. This attribute provides additional information to the ausfInfo. ausfInfoList may be present even if the ausfInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| amfInfo | AmfInfo | O | 0..1 | Specific data for the AMF (AMF Set ID, …) |
| amfInfoList | map(AmfInfo) | O | 1..N | Multiple entries of AmfInfo. This attribute provides additional information to the amfInfo. amfInfoList may be present even if the amfInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| smfInfo | SmfInfo | O | 0..1 | Specific data for the SMF (DNN's, …).  (NOTE 12) |
| smfInfoList | map(SmfInfo) | O | 1..N | Multiple entries of SmfInfo. This attribute provides additional information to the smfInfo. smfInfoList may be present even if the smfInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters.  (NOTE 12) |
| upfInfo | UpfInfo | O | 0..1 | Specific data for the UPF (S-NSSAI, DNN, SMF serving area, interface…) |
| upfInfoList | map(UpfInfo) | O | 1..N | Multiple entries of UpfInfo. This attribute provides additional information to the upfInfo. upfInfoList may be present even if the upfInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| pcfInfo | PcfInfo | O | 0..1 | Specific data for the PCF. |
| pcfInfoList | map(PcfInfo) | O | 1..N | Multiple entries of PcfInfo. This attribute provides additional information to the pcfInfo. pcfInfoList may be present even if the pcfInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| bsfInfo | BsfInfo | O | 0..1 | Specific data for the BSF. |
| bsfInfoList | map(BsfInfo) | O | 1..N | Multiple entries of BsfInfo. This attribute provides additional information to the bsfInfo. bsfInfoList may be present even if the bsfInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| chfInfo | ChfInfo | O | 0..1 | Specific data for the CHF. |
| chfInfoList | map(ChfInfo) | O | 1..N | Multiple entries of ChfInfo. This attribute provides additional information to the chfInfo. chfInfoList may be present even if the chfInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| nefInfo | NefInfo | O | 0..1 | Specific data for the NEF. |
| nrfInfo | NrfInfo | O | 0..1 | Specific data for the NRF. |
| udsfInfo | UdsfInfo | O | 0..1 | Specific data for the UDSF. |
| udsfInfoList | map(UdsfInfo) | O | 1..N | Multiple entries of udsfInfo. This attribute provides additional information to the udsfInfo. udsfInfoList may be present even if the udsfInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| nwdafInfo | NwdafInfo | O | 0..1 | Specific data for the NWDAF. |
| nwdafInfoList | map(NwdafInfo) | O | 1..N | Multiple entries of nwdafInfo. This attribute provides additional information to the nwdafInfo. nwdafInfoList may be present even if the nwdafInfo is absent.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| pcscfInfoList | map(PcscfInfo) | O | 1..N | Specific data for the P-CSCF.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters.  (NOTE 11) |
| hssInfoList | map(HssInfo) | O | 1..N | Specific data for the HSS.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| customInfo | object | O | 0..1 | Specific data for custom Network Functions |
| recoveryTime | DateTime | O | 0..1 | Timestamp when the NF was (re)started (NOTE 5) (NOTE 6) |
| nfServicePersistence | boolean | O | 0..1 | - true: If present, and set to true, it indicates that the different service instances of a same NF Service in this NF instance, supporting a same API version, are capable to persist their resource state in shared storage and therefore these resources are available after a new NF service instance supporting the same API version is selected by a NF Service Consumer (see 3GPP TS 23.527 [27]).  - false (default): Otherwise, it indicates that the NF Service Instances of a same NF Service are not capable to share resource state inside the NF Instance. |
| nfServices | array(NFService) | O | 1..N | List of NF Service Instances. It shall include the services produced by the NF that can be discovered by other NFs, if any. (NOTE 15)  This attribute is deprecated; the attribute "nfServiceList" should be used instead. |
| nfServiceList | map(NFService) | O | 1..N | Map of NF Service Instances, where the "serviceInstanceId" attribute of the NFService object shall be used as the key of the map. (NOTE 15)  It shall include the services produced by the NF that can be discovered by other NFs, if any. |
| nfProfileChangesSupportInd | boolean | O | 0..1 | NF Profile Changes Support Indicator.  See Annex B.  This IE may be present in the NFRegister or NFUpdate (NF Profile Complete Replacement) request and shall be absent in the response.  true: the NF Service Consumer supports receiving NF Profile Changes in the response.  false (default): the NF Service Consumer does not support receiving NF Profile Changes in the response.  Write-Only: true |
| nfProfilePartialUpdateChangesSupportInd | boolean | O | 0..1 | NF Profile Partial Update Changes Support Indicator.  See Annex B.  This IE may be present in the NFRegister or NFUpdate request and shall be absent in the response.  true: the NF Service Consumer supports receiving NF Profile Changes in the response to an NF Profile Partial Update operation.  false (default): the NF Service Consumer does not support receiving NF Profile Changes in the response to an NF Profile Partial Update operation.  Write-Only: true |
| nfProfileChangesInd | boolean | O | 0..1 | NF Profile Changes Indicator.  See Annex B.  This IE shall be absent in the request to the NRF and may be included by the NRF in NFRegister or NFUpdate response.  true: the NF Profile contains NF Profile changes.  false (default): complete NF Profile.  Read-Only: true |
| defaultNotificationSubscriptions | array(DefaultNotificationSubscription) | O | 1..N | Notification endpoints for different notification types.  (NOTE 10) |
| lmfInfo | LmfInfo | O | 0..1 | Specific data for the LMF. |
| gmlcInfo | GmlcInfo | O | 0..1 | Specific data for the GMLC. |
| nfSetIdList | array(NfSetId) | C | 1..N | NF Set ID defined in clause 28.12 of 3GPP TS 23.003 [12].  At most one NF Set ID shall be indicated per PLMN-ID or SNPN of the NF. At most one combination of an AMF region and an AMF Set ID shall be indicated per PLMN-ID or SNPN in an AMF profile.  This information shall be present if available.  (NOTE 22) (NOTE 23) |
| servingScope | array(string) | O | 1..N | The served area(s) of the NF instance.  The absence of this attribute does not imply that the NF instance can serve every area in the PLMN.  (NOTE 13) |
| lcHSupportInd | boolean | O | 0..1 | This IE indicates whether the NF supports Load Control based on LCI Header (see clause 6.3 of 3GPP TS 29.500 [4]).  - true: the NF supports the feature.  - false (default): the NF does not support the feature. |
| olcHSupportInd | boolean | O | 0..1 | This IE indicates whether the NF supports Overload Control based on OCI Header (see clause 6.4 of 3GPP TS 29.500 [4]).  - true: the NF supports the feature.  - false (default): the NF does not support the feature. |
| nfSetRecoveryTimeList | map(DateTime) | O | 1..N | Map of recovery time, where the key of the map is the *NfSetId* of NF Set(s) that the NF instance belongs to.  When present, the value of each entry of the map shall be the recovery time of the NF Set indicated by the key. |
| serviceSetRecoveryTimeList | map(DateTime) | O | 1..N | Map of recovery time, where the key of the map is the *NfServiceSetId* of the NF Service Set(s) configured in the NF instance.  When present, the value of each entry of the map shall be the recovery time of the NF Service Set indicated by the key. |
| scpDomains | array(string) | O | 1..N | When present, this IE shall carry the list of SCP domains the SCP belongs to, or the SCP domain the NF (other than SCP) or the SEPP belongs to.  (NOTE 14) |
| scpInfo | ScpInfo | O | 0..1 | Specific data for the SCP. |
| seppInfo | SeppInfo | O | 0..1 | Specific data for the SEPP. |
| vendorId | VendorId | O | 0..1 | Vendor ID of the NF instance, according to the IANA-assigned "SMI Network Management Private Enterprise Codes" [38]. |
| supportedVendorSpecificFeatures | map(array(VendorSpecificFeature)) | O | 1..N(1..M) | Map of Vendor-Specific features, where the key of the map is the IANA-assigned "SMI Network Management Private Enterprise Codes" [38]. The string used as key of the map shall contain 6 decimal digits; if the SMI code has less than 6 digits, it shall be padded with leading digits "0" to complete a 6-digit string value.  The value of each entry of the map shall be a list (array) of VendorSpecificFeature objects.  (NOTE 19) |
| aanfInfoList | map(AanfInfo) | O | 1..N | Multiple entries of AanfInfo.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| 5gDdnmfInfo | 5GDdnmfInfo | O | 0..1 | Specific data for the 5G DDNMF (5G DDNMF ID, …) |
| mfafInfo | MfafInfo | O | 0..1 | Specific data for the MFAF |
| easdfInfoList | map(EasdfInfo) | O | 1..N | EASDF specific data.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters.  (NOTE 20) |
| dccfInfo | DccfInfo | O | 0..1 | Specific data for the DCCF. |
| nsacfInfoList | map(NsacfInfo) | O | 1..N | Specific data for the NSACF.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| mbSmfInfoList | map(MbSmfInfo) | O | 1..N | MB-SMF specific data.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| tsctsfInfoList | map(TsctsfInfo) | O | 1..N | Specific data for the TSCTSF.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| mbUpfInfoList | map(MbUpfInfo) | O | 1..N | MB-UPF specific data.  The key of the map shall be a (unique) valid JSON string per clause 7 of IETF RFC 8259 [22], with a maximum of 32 characters. |
| trustAfInfo | TrustAfInfo | O | 0..1 | Specific data for the trusted AF. |
| nssaafInfo | NssaafInfo | O | 0..1 | Specific data for the NSSAAF. |
| hniList | arrary(Fqdn) | C | 1..N | Identifications of Credentials Holder or Default Credentials Server.  This IE shall be present if the NFs are available for the case of access to an SNPN using credentials owned by a Credentials Holder or for the case of SNPN Onboarding using a DCS. |
| iwmscInfo | IwmscInfo | O | 0..1 | Specific data for the SMS-IWMSC. |
| mnpfInfo | MnpfInfo | O | 0..1 | Specific data for the MNPF. |
| smsfInfo | SmsfInfo | O | 0..1 | Specific data for the SMSF. |
| NOTE 1: At least one of the addressing parameters (fqdn, ipv4address or ipv6adress) shall be included in the NF Profile. If the NF supports the NF services with "https" URI scheme (i.e use of TLS is mandatory), then the FQDN shall be provided in the NF Profile or the NF Service profile (see clause 6.1.6.2.3) and it shall be used to construct the target URI (unless overriden by a NFService-specific FQDN). See NOTE 1 of Table 6.1.6.2.3-1 for the use of these parameters. If multiple ipv4 addresses and/or ipv6 addresses are included in the NF Profile, the NF Service Consumer of the discovery service shall select one of these addresses randomly, unless operator defined local policy of IP address selection, in order to avoid overload for a specific ipv4 address and/or ipv6 address.  NOTE 2: If the type of Network Function is UPF, the addressing information is for the UPF N4 interface and, if the UPF registers service instances supporting the UPF Event Exposure service without registering addressing information at these service instances level, also for accessing the UPF Event Exposure service at these service instances. If the type of Network Function is MB-UPF, the addressing information is for the MB-UPF N4mb interface. If the type of Network Function is a P-CSCF and if no Gm FQDN or IP addresses are registered in the pcscfInfoList attribute, the addressing information is also used for the P-CSCF Gm interface.  NOTE 3: A requester NF may use this information to select a NF instance (e.g. a NF instance preferably located in the same data center).  NOTE 4: The capacity and priority parameters, if present, are used for NF selection and load balancing. The priority and capacity attributes shall be used for NF selection in the same way that priority and weight are used for server selection as defined in IETF RFC 2782 [23].  NOTE 5: The NRF shall notify NFs subscribed to receiving notifications of changes of the NF profile, if the NF recoveryTime or the nfStatus is changed. See clause 6.2 of 3GPP TS 23.527 [27].  NOTE 6: A requester NF may consider that all the resources created in the NF before the NF recovery time have been lost. This may be used to detect a restart of a NF and to trigger appropriate actions, e.g. release local resources. See clause 6.2 of 3GPP TS 23.527 [27].  NOTE 7: A NF may register multiple PLMN IDs in its profile within a PLMN comprising multiple PLMN IDs. If so, all the attributes of the NF Profile shall apply to each PLMN ID registered in the plmnList. As an exception, attributes including a PLMN ID, e.g. IMSI-based SUPI ranges, TAIs and GUAMIs, are specific to one PLMN ID and the NF may register in its profile multiple occurrences of such attributes for different PLMN IDs (e.g. the UDM may register in its profile SUPI ranges for different PLMN IDs).  NOTE 8: Other NFs are in a different PLMN if they belong to none of the PLMN ID(s) configured for the PLMN of the NRF.  NOTE 9: This is for the use case where an NF (e.g. AMF) supports multiple PLMNs and the slices supported in each PLMN are different. See clause 9.2.6.2 of 3GPP TS 38.413 [29].  NOTE 10: For notification types that may be associated with a specifc service of the NF Instance receiving the notification (see clause 6.1.6.3.4), if notification endpoints are present both in the profile of the NF instance (NFProfile) and in some of its NF Services (NFService) for a same notification type, the notification endpoint(s) of the NF Services shall be used for this notification type. The defaultNotificationSubscriptions attribute may contain multiple default subscriptions for a same notification type; in that case, those default subscriptions are used as alternative notification endpoints so, for each notification event that needs to be sent, the NF Service Consumer shall select one of such subscriptions and use it to send the notification.  NOTE 11: The absence of the pcscfInfoList attribute in a P-CSCF profile indicates that the P-CSCF can be selected for any DNN and Access Type, and that the P-CSCF Gm addressing information is the same as the addressing information registered in the fqdn, ipv4Addresses and ipv4Addresses attributes of the NF profile.  NOTE 12: The absence of both the smfInfo and smfInfoList attributes in an SMF profile indicates that the SMF can be selected for any S-NSSAI listed in the sNssais and perPlmnSnssaiList IEs, or for any S-NSSAI if neither the sNssais IE nor the perPlmnSnssaiList IE are present, and for any DNN, TAI and access type.  NOTE 13: The servingScope attribute may indicate geographical areas, It may be used e.g. to discover and select NFs in centralized Data Centers that are expected to serve users located in specific region(s) or province(s). It may also be used to reduce the large configuration of TAIs in the NF instances.  NOTE 14: An NF (other than a SCP) can register at most one SCP domain in NF profile, i.e. the NF can belong to only one SCP domain. If an NF (other than a SCP) includes this information in its profile, this indicates that the services produced by this NF should be accessed preferably via an SCP from the SCP domain the NF belongs to.  NOTE 15: If the NF Service Consumer that issues an NF profile retrieval request indicates support for the "Service-Map" feature, the NRF shall return in the NF profile retrieval response the list of NF Service Instances in the "nfServiceList" map attribute. Otherwise, the NRF shall return the list of NF Service Instances in the "nfServices" array attribute.  NOTE 16: The nfStatus also indicate the Status of the NF instance as NF Service Consumer for notification delivery. When a notification is to be delivered to the NF instance and the NF Service Producer (or SCP) has been aware that the NF instance is not operative from the nfStatus in its NF profile, the NF Service producer (or SCP) shall reselect another NF Service Consumer as target if possible, e.g. using binding indication or discovery factors previously provided for the notification. When selecting or reselecting an NF Service Consumer for notification delivery, not operative NF instances shall not be selected as target.  NOTE 17: A change of this attribute shall trigger a "NF\_PROFILE\_CHANGED" notification from NRF, if the change of the NF Profile results in that the NF Instance starts or stops being authorized to be accessed by an NF having subscribed to be notified about NF profile changes.  NOTE 18: For API URIs constructed with an FQDN, the NF Service Consumer may use the FQDN of the target URI to do a DNS query and obtain the IP address(es) to setup the TCP connection, and ignore the IP addresses that may be present in the NFProfile; alternatively, the NF Service Consumer may use those IP addresses to setup the TCP connection, if no service-specific FQDN or IP address is provided in the NFService data and if the NF Service Consumer supports to indicate specific IP address(es) to establish an HTTP/2 connection with an FQDN in the target URI.  NOTE 19: When present, this attribute allows an NF requesting NF Discovery (e.g. an NF Service Consumer) to determine which vendor-specific extensions are supported in a given NF (e.g. an NF Service Producer), so as to select an appropriate NF with specific capability, or to include or not the vendor-specific attributes (see 3GPP TS 29.500 [4] clause 6.6.3) required for a given feature in subsequent messages towards a certain NF. One given vendor-specific feature shall not appear in both NF Profile and NF Service Profile. If one vendor-specific feature is service related, it shall only be included in the NF Service Profile.  NOTE 20: The absence of the easdfInfoList attribute in an EASDF profile indicates that the EASDF can be selected for any S-NSSAI, DNN, DNAI or PSA UPF N6 IP address.  NOTE 21: The NF service consumer when invoking NF services offered by collocated NF service producers shall follow the respective service API in the same manner as if they were not collocated with any other NF type. The NF service consumer shall not assume any optimization of signaling between the NF service consumer and the collocated NF service producers.  NOTE 22: The nfSetIdList attribute shall be present only if all NF service instance(s) of the NF instance are redundant at NF Set level. I.e. any NF service instance shall be redundant (i.e. functionally equivalent, inter-changeable and sharing contexts) with equivalent service instance(s) of every other NF instance(s) within the indicated NF Set or, if the NF service instance belongs to an NF service set, it shall be redundant with NF service instance(s) in an equivalent NF service set of every other NF instance(s) within the indicated NF set.  NOTE 23: The NF Instance shall be removed from an NF set or re-assigned to another NF set ONLY when there is NO ongoing resource/context associated with the NF instance. | | | | |

\* \* \* Next Change \* \* \* \*

# A.2 Nnrf\_NFManagement API

*(... text not shown for clarity ...)*

NFProfile:

description: Information of an NF Instance registered in the NRF

type: object

required:

- nfInstanceId

- nfType

- nfStatus

anyOf:

- required: [ fqdn ]

- required: [ ipv4Addresses ]

- required: [ ipv6Addresses ]

properties:

nfInstanceId:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/NfInstanceId'

nfInstanceName:

type: string

nfType:

$ref: '#/components/schemas/NFType'

nfStatus:

$ref: '#/components/schemas/NFStatus'

collocatedNfInstances:

type: array

items:

$ref: '#/components/schemas/CollocatedNfInstance'

minItems: 1

heartBeatTimer:

type: integer

minimum: 1

plmnList:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/PlmnId'

minItems: 1

snpnList:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/PlmnIdNid'

minItems: 1

sNssais:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/ExtSnssai'

minItems: 1

perPlmnSnssaiList:

type: array

items:

$ref: '#/components/schemas/PlmnSnssai'

minItems: 1

nsiList:

type: array

items:

type: string

minItems: 1

fqdn:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/Fqdn'

interPlmnFqdn:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/Fqdn'

ipv4Addresses:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/Ipv4Addr'

minItems: 1

ipv6Addresses:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/Ipv6Addr'

minItems: 1

allowedPlmns:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/PlmnId'

minItems: 1

allowedSnpns:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/PlmnIdNid'

minItems: 1

allowedNfTypes:

type: array

items:

$ref: '#/components/schemas/NFType'

minItems: 1

allowedNfDomains:

type: array

items:

type: string

minItems: 1

allowedNssais:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/ExtSnssai'

minItems: 1

priority:

type: integer

minimum: 0

maximum: 65535

capacity:

type: integer

minimum: 0

maximum: 65535

load:

type: integer

minimum: 0

maximum: 100

loadTimeStamp:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/DateTime'

locality:

type: string

extLocality:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string serves

as key representing a type of locality

type: object

additionalProperties:

type: string

minProperties: 1

udrInfo:

$ref: '#/components/schemas/UdrInfo'

udrInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of UdrInfo

type: object

additionalProperties:

$ref: '#/components/schemas/UdrInfo'

minProperties: 1

udmInfo:

$ref: '#/components/schemas/UdmInfo'

udmInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of UdmInfo

type: object

additionalProperties:

$ref: '#/components/schemas/UdmInfo'

minProperties: 1

ausfInfo:

$ref: '#/components/schemas/AusfInfo'

ausfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of AusfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/AusfInfo'

minProperties: 1

amfInfo:

$ref: '#/components/schemas/AmfInfo'

amfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of AmfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/AmfInfo'

minProperties: 1

smfInfo:

$ref: '#/components/schemas/SmfInfo'

smfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of SmfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/SmfInfo'

minProperties: 1

upfInfo:

$ref: '#/components/schemas/UpfInfo'

upfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of UpfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/UpfInfo'

minProperties: 1

pcfInfo:

$ref: '#/components/schemas/PcfInfo'

pcfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of PcfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/PcfInfo'

minProperties: 1

bsfInfo:

$ref: '#/components/schemas/BsfInfo'

bsfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of BsfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/BsfInfo'

minProperties: 1

chfInfo:

$ref: '#/components/schemas/ChfInfo'

chfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of ChfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/ChfInfo'

minProperties: 1

nefInfo:

$ref: '#/components/schemas/NefInfo'

nrfInfo:

$ref: '#/components/schemas/NrfInfo'

udsfInfo:

$ref: '#/components/schemas/UdsfInfo'

udsfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of UdsfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/UdsfInfo'

minProperties: 1

nwdafInfo:

$ref: '#/components/schemas/NwdafInfo'

nwdafInfoList:

type: object

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of NwdafInfo

additionalProperties:

$ref: '#/components/schemas/NwdafInfo'

minProperties: 1

pcscfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of PcscfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/PcscfInfo'

minProperties: 1

hssInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of HssInfo

type: object

additionalProperties:

$ref: '#/components/schemas/HssInfo'

minProperties: 1

customInfo:

type: object

recoveryTime:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/DateTime'

nfServicePersistence:

type: boolean

default: false

nfServices:

deprecated: true

type: array

items:

$ref: '#/components/schemas/NFService'

minItems: 1

nfServiceList:

description: >

A map (list of key-value pairs) where serviceInstanceId serves as key of NFService

type: object

additionalProperties:

$ref: '#/components/schemas/NFService'

minProperties: 1

nfProfileChangesSupportInd:

type: boolean

default: false

writeOnly: true

nfProfilePartialUpdateChangesSupportInd:

type: boolean

default: false

writeOnly: true

nfProfileChangesInd:

type: boolean

default: false

readOnly: true

defaultNotificationSubscriptions:

type: array

items:

$ref: '#/components/schemas/DefaultNotificationSubscription'

lmfInfo:

$ref: '#/components/schemas/LmfInfo'

gmlcInfo:

$ref: '#/components/schemas/GmlcInfo'

nfSetIdList:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/NfSetId'

minItems: 1

servingScope:

type: array

items:

type: string

minItems: 1

lcHSupportInd:

type: boolean

default: false

olcHSupportInd:

type: boolean

default: false

nfSetRecoveryTimeList:

description: A map (list of key-value pairs) where NfSetId serves as key of DateTime

type: object

additionalProperties:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/DateTime'

minProperties: 1

serviceSetRecoveryTimeList:

description: >

A map (list of key-value pairs) where NfServiceSetId serves as key of DateTime

type: object

additionalProperties:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/DateTime'

minProperties: 1

scpDomains:

type: array

items:

type: string

minItems: 1

scpInfo:

$ref: '#/components/schemas/ScpInfo'

seppInfo:

$ref: '#/components/schemas/SeppInfo'

vendorId:

$ref: '#/components/schemas/VendorId'

supportedVendorSpecificFeatures:

description: >

The key of the map is the IANA-assigned SMI Network Management Private Enterprise Codes

type: object

additionalProperties:

type: array

items:

$ref: '#/components/schemas/VendorSpecificFeature'

minItems: 1

minProperties: 1

aanfInfoList:

type: object

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of AanfInfo

additionalProperties:

$ref: '#/components/schemas/AanfInfo'

minProperties: 1

5gDdnmfInfo:

$ref: '#/components/schemas/5GDdnmfInfo'

mfafInfo:

$ref: '#/components/schemas/MfafInfo'

easdfInfoList:

type: object

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of EasdfInfo

additionalProperties:

$ref: '#/components/schemas/EasdfInfo'

minProperties: 1

dccfInfo:

$ref: '#/components/schemas/DccfInfo'

nsacfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of NsacfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/NsacfInfo'

minProperties: 1

mbSmfInfoList:

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of MbSmfInfo

type: object

additionalProperties:

$ref: '#/components/schemas/MbSmfInfo'

minProperties: 1

tsctsfInfoList:

type: object

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of TsctsfInfo

additionalProperties:

$ref: '#/components/schemas/TsctsfInfo'

minProperties: 1

mbUpfInfoList:

type: object

description: >

A map (list of key-value pairs) where a (unique) valid JSON string

serves as key of MbUpfInfo

additionalProperties:

$ref: '#/components/schemas/MbUpfInfo'

minProperties: 1

trustAfInfo:

$ref: '#/components/schemas/TrustAfInfo'

nssaafInfo:

$ref: '#/components/schemas/NssaafInfo'

hniList:

type: array

items:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/Fqdn'

minItems: 1

iwmscInfo:

$ref: '#/components/schemas/IwmscInfo'

mnpfInfo:

$ref: '#/components/schemas/MnpfInfo'

smsfInfo:

$ref: '#/components/schemas/SmsfInfo'

*(... text not shown for clarity ...)*

\* \* \* Next Change \* \* \* \*

Annex B (normative):  
NF Profile changes in NFRegister and NFUpdate responses

# B.1 General

In the NFRegister and NFUpdate (NF Profile Complete Replacement and NF Profile Partial Update) service operations, a NF Service Consumer may indicate to the NRF that it supports receiving NF Profile changes in the response from the NRF, by including the nfProfileChangesSupportInd and/or the nfProfilePartialUpdateChangesSupportInd attributes set to "true" in the NFProfile it registers to or replaces in the NRF.

NOTE: For NF Profile Partial Update (which uses the HTTP PATCH operation), the NF Service Consumer indicates its support of this capability during the initial NFRegister operation, or during an NF Profile Complete Replacement (i.e., in the payload body of the corresponding HTTP PUT request).

The NRF may return NF Profile changes, instead of the complete NF Profile, in NFRegister or NFUpdate responses, if the NF Service Consumer has indicated corresponding support in its NFProfile data. When doing so, the NRF shall include in the NF Profile returned in the response:

- attributes that are mandatory to include in the NF Profile; if an optional IE is included (e.g. nfServices), attributes that are mandatory to include in this optional IE (e.g. serviceInstanceId) shall also be included;

- optional or conditional IEs that have been changed or added by the NRF; and

- the nfProfileChangesInd IE set to "true", indicating that the returned profile contains NF profile changes.

EXAMPLE 1: The NRF does not change the NF Profile received in the request.

The NRF response contains a NFProfile with just the following IEs:

- nfInstanceId, nfType, nfStatus; and

- nfProfileChangesInd IE set to "true".

EXAMPLE 2: The NRF modifies or adds the heartbeatTimer attribute to the NF Profile received in the request.

The NRF response contains a NFProfile with just the following IEs:

- nfInstanceId, nfType, nfStatus;

- heartbeatTimer with NRF chosen value;

- nfProfileChangesInd IE set to "true".

\* \* \* End of Changes \* \* \* \*