tmforum

TM Forum Specification

Service Inventory API User Guide

TMF638

Team Approved Date: 05-May-2020

Release Status: Pre-production	Approval Status: Team Approved
Version 4.0.0	IPR Mode: RAND

NOTICE

Direct inquiries to the TM Forum office:

4 Century Drive, Suite 100
Parsippany, NJ 07054, USA
Tel No. +1 973 944 5100
Fax No. +1 973 998 7916
TM Forum Web Page: www.tmforum.org

Table of Contents

Introduction

The following document is intended to provide details of the REST API interface for Service Inventory. The intent of this API is to provide a consistent/standardized mechanism to query and manipulate the Service inventory.

SAMPLE USE CASES

Service inventory query for a customer

The Service Inventory API can be used to query the service instances for a customer via Self Service Portal or the Call Centre operator can query the service instances on behalf of the customer while a customer may have a complaint or a query.

Note: Only the CustomerFacingServices instances will be presented to the customer.

Service inventory update as part of service provisioning

The Service Inventory API can be called by the Service Order Management to create a new service instance/ update an existing service instance in the Service Inventory.

Support of polymorphism and extension patterns

Support of polymorphic collections and types and schema based extension is provided by means of a list of generic meta-attributes that we describe below. Polymorphism in collections occurs when entities inherit from base entities, for instance a TypeAService or TypeBService inheriting properties from the base Service entity.

Generic support of polymorphism and pattern extensions is described in the TMF API Guidelines v3.0 Part 2 document.

The @type attribute provides a way to represent the actual class type of an entity. For example, within a list of Service instances some may be instances of TypeAService where other could be instances of TypeBService. The @type gives this information. All resources and sub-resources of this API have a @type attributes that can be provided when this is useful.

The @referredType can be used within reference entities (like for instance an RelatedParty object) to explicitly denote the actual entity type of the referred class. Notice that in reference entities the @type, when used, denotes the class type of the reference itself, such as RelatedParty, and not the class type of the referred object. However since reference classes are rarely sub-classed, @type is generally not useful in reference objects.

The @schemaLocation property can be used in resources to allow specifying user-defined properties of an Entity or to specify the expected characteristics of an entity.

The @baseType attribute gives a way to provide explicitly the base of class of a given resource that has been extended.

RESOURCE MODEL

Managed Entity and Task Resource Models

Service resource

Service is a base class for defining the Service hierarchy. All Services are characterized as either being possibly visible and usable by a Customer or not. This gives rise to the two subclasses of Service: CustomerFacingService and ResourceFacingService.

Resource model
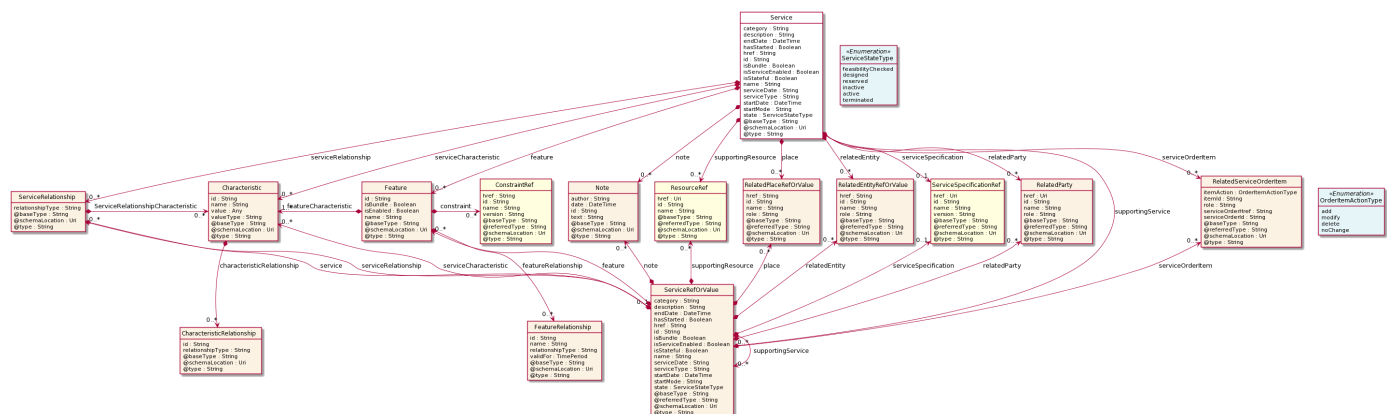
Field descriptions

Service fields

category	A string. Is it a customer facing or resource facing service.
description	A string. Free-text description of the service.
endDate	A date time (DateTime). Date when the service ends.
feature	A list of features (Feature [*]). A list of feature associated with this service.
hasStarted	A boolean. If TRUE, this Service has already been started.
href	A string. Reference of the service.
id	A string. Unique identifier of the service.
isBundle	A boolean. If true, the service is a ServiceBundle which regroup a service hierachy. If false, the service is an 'atomic' service (hierachy leaf).
isServiceEnabled	A boolean. If FALSE and hasStarted is FALSE, this particular Service has NOT been enabled for use - if FALSE and hasStarted is TRUE then the service has failed.
isStateful	A boolean. If TRUE, this Service can be changed without affecting any other services.

name	A string. Name of the service.
note	A list of notes (Note [*]). A list of notes made on this service.
place	A list of related place ref or values (RelatedPlaceRefOrValue [*]). A list of places (Place [*]). Used to define a place useful for the service (for example a geographical place whre the service is installed).
relatedEntity	A list of related entity ref or values (RelatedEntityRefOrValue [*]). A list of related entity in relationship with this service.
relatedParty	A list of related parties (RelatedParty [*]). A list of related party references (RelatedParty [*]). A related party defines party or party role linked to a specific entity.
serviceCharacteristic	A list of characteristics (Characteristic [*]). A list of characteristics that characterize this service (ServiceCharacteristic [*]).
serviceDate	A string. Date when the service was created (whatever its status).
serviceOrderItem	A list of related service order items (RelatedServiceOrderItem [*]). A list of service order items related to this service.
serviceRelationship	A list of service relationships (ServiceRelationship [*]). A list of service relationships (ServiceRelationship [*]). Describes links with other service(s) in the inventory.
serviceSpecification	A service specification reference (ServiceSpecificationRef). The specification from which this service was instantiated.
serviceType	A string. Business type of the service.
startDate	A date time (DateTime). Date when the service starts.
startMode	A string. This attribute is an enumerated integer that indicates how the Service is started, such as: 0: Unknown; 1: Automatically by the managed environment; 2: Automatically by the owning device; 3: Manually by the Provider of the Service; 4: Manually by a Customer of the Provider; 5: Any of the above.
state	A service state type (ServiceStateType). The life cycle state of the service, such as designed, reserved, active, etc...
supportingResource	A list of resource references (ResourceRef [*]). A list of supporting resources (SupportingResource [*]).Note: only Service of type RFS can be associated with Resources.
supportingService	A list of service ref or values (ServiceRefOrValue [*]). A list of supporting services (SupportingService [*]). A collection of services that support this service (bundling, link CFS to RFS).

Characteristic sub-resource

Describes a given characteristic of an object or entity through a name/value pair.

@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
characteristicRelationship	A list of characteristic relationships (CharacteristicRelationship [*]). Another Characteristic that is related to the current Characteristic;.
id	A string. Unique identifier of the characteristic.
name	A string. Name of the characteristic.
value	An any (Any). The value of the characteristic.
valueType	A string. Data type of the value of the characteristic.

CharacteristicRelationship sub-resource

Another Characteristic that is related to the current Characteristic;.

id	A string. Unique identifier of the characteristic.
relationshipType	A string. The type of relationship.

Feature sub-resource

Configuration feature.

constraint	A list of constraint references (ConstraintRef [*]). This is a list of feature constraints.
featureCharacteristic	A list of characteristics (Characteristic [1..*]). This is a list of Characteristics for a particular feature.
featureRelationship	A list of feature relationships (FeatureRelationship [*]). Configuration feature.
id	A string. Unique identifier of the feature.
isBundle	A boolean. True if this is a feature group. Default is false.
isEnabled	A boolean. True if this feature is enabled. Default is true.
name	A string. This is the name for the feature.

FeatureRelationship sub-resource

Configuration feature.

id	A string. Unique identifier of the target feature.

name	A string. This is the name of the target feature.
relationshipType	A string. This is the type of the feature relationship.
validFor	A time period. The period for which this feature relationship is valid.

Note sub-resource

Extra information about a given entity.

author	A string. Author of the note.
date	A date time (DateTime). Date of the note.
id	A string. Identifier of the note within its containing entity (may or may not be globally unique, depending on provider implementation).
text	A string. Text of the note.

RelatedEntityRefOrValue sub-resource

A reference to an entity, where the type of the entity is not known in advance. A related entity defines a entity described by reference or by value linked to a specific entity. The polymorphic attributes @type, @schemaLocation & @referredType are related to the Entity and not the RelatedEntityRefOrValue class itself.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. Name of the related entity.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
role	A string.

RelatedParty sub-resource

Related Entity reference. A related party defines party or party role linked to a specific entity.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. Name of the related entity.
href	An uri (Uri). Hyperlink reference.

id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
role	A string. Role played by the related party.

RelatedPlaceRefOrValue sub-resource

Related Entity reference. A related place defines a place described by reference or by value linked to a specific entity. The polymorphic attributes @type, @schemaLocation & @referredType are related to the place entity and not the RelatedPlaceRefOrValue class itself.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. A user-friendly name for the place, such as [Paris Store], [London Store], [Main Home].
href	A string. Unique reference of the place.
id	A string. Unique identifier of the place.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
role	A string.

RelatedServiceOrderItem sub-resource

RelatedServiceOrderItem (a ServiceOrder item) .The service order item which triggered service creation/change/termination.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
itemAction	An order item action type (OrderItemActionType). Action of the order item for this service.
itemId	A string. Identifier of the order item where the service was managed.
role	A string. role of the service order item for this service.
serviceOrderHref	A string. Reference of the related entity.

serviceOrderId   A string. Unique identifier of a related entity.

ServiceRefOrValue sub-resource

A Service to be created defined by value or existing defined by reference. The polymorphic attributes @type, @schemaLocation & @referredType are related to the Service entity and not the RelatedServiceRefOrValue class itself.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. Name of the service.
href	A string. Reference of the service.
id	A string. Unique identifier of the service.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
category	A string. Is it a customer facing or resource facing service.
description	A string. Free-text description of the service.
endDate	A date time (DateTime). Date when the service ends.
hasStarted	A boolean. If TRUE, this Service has already been started.
isBundle	A boolean. If true, the service is a ServiceBundle which regroup a service hierachy. If false, the service is a 'atomic' service (hierachy leaf).
isServiceEnabled	A boolean. If FALSE and hasStarted is FALSE, this particular Service has NOT been enabled for use - if FALSE and hasStarted is TRUE then the service has failed.
isStateful	A boolean. If TRUE, this Service can be changed without affecting any other services.
serviceDate	A string. Date when the service was created (whatever its status).
serviceType	A string. Business type of the service.
startDate	A date time (DateTime). Date when the service starts.
startMode	A string. This attribute is an enumerated integer that indicates how the Service is started, such as: 0: Unknown; 1: Automatically by the managed environment; 2: Automatically by the owning device; 3: Manually by the Provider of the Service; 4: Manually by a Customer of the Provider; 5: Any of the above.
feature	A list of features (Feature [*]). A list of feature associated with this service.

note	A list of notes (Note [*]). A list of notes made on this service.
place	A list of related place ref or values (RelatedPlaceRefOrValue [*]). A list of places (Place [*]). Used to define a place useful for the service (for example a geographical place whre the service is installed).
relatedEntity	A list of related entity ref or values (RelatedEntityRefOrValue [*]). A list of related entity in relationship with this service.
relatedParty	A list of related parties (RelatedParty [*]). A list of related party references (RelatedParty [*]). A related party defines party or party role linked to a specific entity.
serviceCharacteristic	A list of characteristics (Characteristic [*]). A list of characteristics that characterize this service (ServiceCharacteristic [*]).
serviceOrderItem	A list of related service order items (RelatedServiceOrderItem [*]). A list of service order items related to this service.
serviceRelationship	A list of service relationships (ServiceRelationship [*]). A list of service relationships (ServiceRelationship [*]). Describes links with other service(s) in the inventory.
serviceSpecification	A service specification reference (ServiceSpecificationRef). The specification from which this service was instantiated.
state	A service state type (ServiceStateType). The life cycle state of the service, such as designed, reserved, active, etc...
supportingResource	A list of resource references (ResourceRef [*]). A list of supporting resources (SupportingResource [*]).Note: only Service of type RFS can be associated with Resources.
supportingService	A list of service ref or values (ServiceRefOrValue [*]). A list of supporting services (SupportingService [*]). A collection of services that support this service (bundling, link CFS to RFS).

ServiceRelationship sub-resource

relationshipType	A string.
service	A service ref or value (ServiceRefOrValue). A Service to be created defined by value or existing defined by reference. The polymorphic attributes @type, @schemaLocation & @referredType are related to the Service entity and not the RelatedServiceRefOrValue class itself.
serviceRelationshipCharacteristic	A list of characteristics (Characteristic [*]). Describes a given characteristic of an

object or entity through a name/value pair.

ConstraintRef relationship

Constraint reference. The Constraint resource represents a policy/rule applied to an entity or entity spec.

@referredType	A string. The (class) type of the referred constraint.
href	A string. Hyperlink reference to the target constraint.
id	A string. reference id to the target constraint.
name	A string. Name given to the constraint.
version	A string. constraint version.

ResourceRef relationship

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. Name of the related entity.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.

ServiceSpecificationRef relationship

Service specification reference: ServiceSpecification(s) required to realize a ProductSpecification.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. Name of the related entity.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type	A string. When sub-classing, this defines the sub-class Extensible name.
version	A string. Service specification version.

Json representation sample

We provide below the json representation of an example of a 'Service' resource object

```json
{
  "id": "5351",
  "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5351",
  "serviceType": "Cloud",
  "name": "vCPE serial 1355615",
  "description": "Instantiation of vCPE",
  "state": "Active",
  "category": "CFS",
  "isServiceEnabled": true,
  "hasStarted": true,
  "startMode": "1",
  "isStateful": true,
  "startDate": "2018-01-15T12:26:11.747Z",
  "serviceSpecification": {
    "id": "1212",
    "href": "https://mycsp.com:8080/tmf-api/serviceCatalogManagement/v4/serviceSpecification/1212",
    "name": "vCPE",
    "version": "1.0.0",
    "@referredType": "ServiceSpecification"
  },
  "feature": [
    {
      "id": "Feat1",
      "isEnabled": true,
      "name": "ElasticBandwith",
      "featureCharacteritic": [
        {
          "name": "isCapped",
          "value": true,
          "id": "45gh-fg",
          "valueType": "boolean"
        }
      ]
    }
  ],
  "serviceCharacteristic": [
    {
      "id": "452-gh6",
      "name": "vCPE",
      "valueType": "object",
      "value": {
        "@type": "JSONSpecification",
        "@schemaLocation": "http://my.schemas/vCPE.schema.json",
        "vCPE_IP": "193.218.236.21",
        "MaxTxRate": 300,
        "TransmitPower": "11 dBm",
        "maxTream": "OFF"
```

```
      }
    }
  ],
  "serviceRelationship": [
    {
      "relationshipType": "DependentOn",
      "ServiceRelationshipCharacteristic": [
        {
          "id": "126",
          "name": "CrossRef",
          "value": "44-11-h",
          "valueType": "string"
        }
      ],
      "service": {
        "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5645",
        "id": "5645",
        "@type": "ServiceRef",
        "@referredType": "Service"
      }
    }
  ],
  "supportingService": [
    {
      "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5885",
      "id": "5885",
      "@type": "ServiceRef",
      "@referredType": "Service"
    }
  ],
  "supportingResource": [
    {
      "id": "6161",
      "href": "https://mycsp.com:8080/tmf-api/resourceInventoryManagement/v4/resource/5351",
      "name": "GenInfra",
      "@referredType": "VirtualResource"
    },
    {
      "id": "7171",
      "href": "https://mycsp.com:8080/tmf-api/resourceInventoryManagement/v4/resource/7171",
      "name": "BNG_MUX",
      "value": "01 25 65",
      "@referredType": "VNF"
    }
  ],
  "relatedParty": [
    {
      "href": "https://mycsp.com:8080/tmf-api/partyManagement/v4/party/456",
      "id": "456",
      "name": "John Doe",
      "role": "user",
      "@referredType": "Individual"
    }
  ],
  "serviceOrderItem": [
    {
```

```
      "serviceOrderHref": "https://mycsp.com:8080/tmf-api/serviceOrder/v4/serviceOrder/42",
      "serviceOrderId": "42",
      "role": "initiator",
      "@referredType": "ServiceOrder",
      "itemId": "1",
      "itemAction": "add"
    },
    {
      "serviceOrderHref": "https://mycsp.com:8080/tmf-api/serviceOrder/v4/serviceOrder/896",
      "serviceOrderId": "896",
      "role": "activation",
      "@referredType": "ServiceOrder",
      "itemId": "4",
      "itemAction": "change"
    }
  ],
  "place": [
    {
      "href": "https://mycsp.com:8080/tmf-api/geographicAddressManagement/v4/geographicAddress/2435",
      "id": "2435",
      "name": "Customer primary location",
      "@referredType": "GeographicAddress",
      "role": "InstallationAddress"
    }
  ],
  "note": [
    {
      "author": "Harvey Poupon",
      "date": "2018-01-15T12:26:11.748Z",
      "text": "This service was installed automatically, no issues were noted in testing."
    }
  ],
  "@type": "Service",
  "@schemaLocation": "https://mycsp.com:8080/tmf-api/schema/Service/vCPE.schema.json"
}
```

Lifecycle of Service

Here is the state machine diagram for a Service:
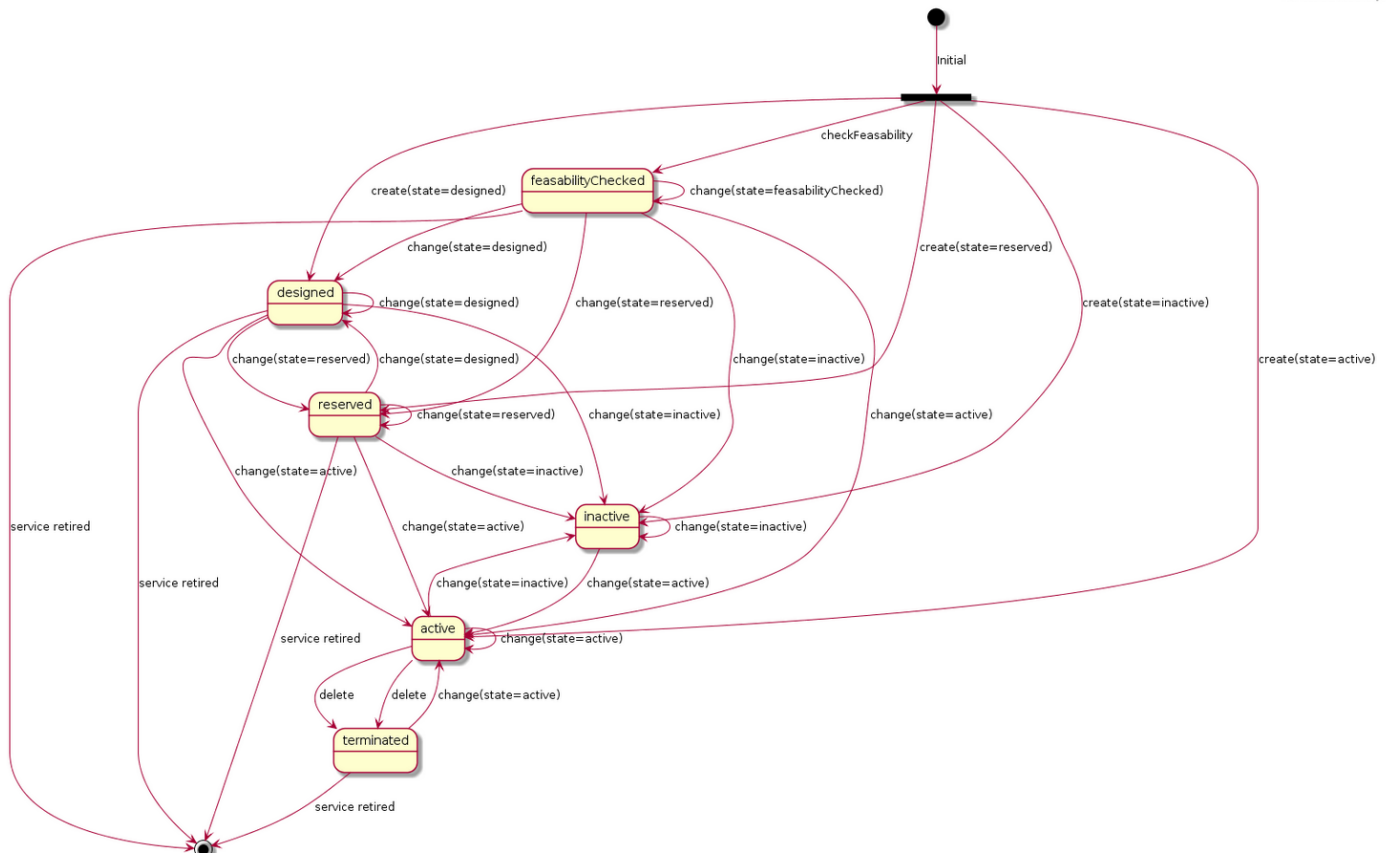
The following table provided service state explanation:

A service is in state...	when the Service Component Activation Interface shall support the following operations
feasabilityChecked	The service component OS is requested to determine whether the necessary resources are available and sufficient for the installation of a given service. It should be noted that this is just an initial feasibility check and no RFS is created.
designed	The Service Component OS is requested to design the service. This could be use to allocate resource but nothing is supposed to be reserved.
reserved	The Service Component OS is requested to reserve a resource, or a set of resources, required for a service. This situation allows for the requesting OS to determine whether the underlying resources are available and reserve them in order to support a service (a RFS). As a result of the reservation request, a service is instantiated & reserved.
inactive	The service is deactivated - The Resource Facing service is deactivated and thus is no longer available for service. It remains allocated to the CFS that is managed by the high level service activation OS
active	The service component OS is requested to activate a given resource facing service such that the component is fully available and active as part of the CFS. When this request is complete, all RFS component shall be in active state
terminated	The service is 'logically deleted'. The resource facing service is deleted and thus allocated from the CFS. All associated resources are freed and made available for service to other users.

Notification Resource Models

4 notifications are defined for this API

Notifications related to Service:
   - ServiceCreateEvent
   - ServiceAttributeValueChangeEvent
   - ServiceStateChangeEvent
   - ServiceDeleteEvent

The notification structure for all notifications in this API follow the pattern depicted by the figure below.
A notification event resource (depicted by "SpecificEvent" placeholder) is a sub class of a generic Event structure containing at least an id of the event occurrence (eventId), an event timestamp (eventTime), and the name of the resource (eventType).
This notification structure owns an event payload structure ("SpecificEventPayload" placeholder) linked to the resource concerned by the notification using the resource name as access field ("resourceName" placeholder).
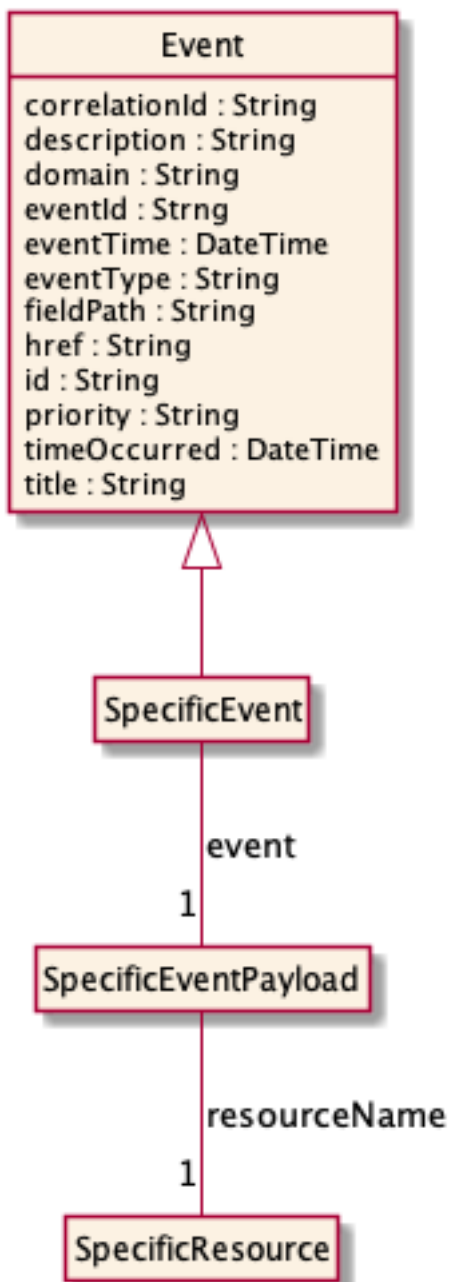
Service Create Event

Notification ServiceCreateEvent case for resource Service

Json representation sample

We provide below the json representation of an example of a 'ServiceCreateEvent' notification event object

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"ServiceCreateEvent",
  "event": {
    "service" :
```

```
            {-- SEE Service RESOURCE SAMPLE --}
    }
}
```

Service Attribute Value Change Event

Notification ServiceAttributeValueChangeEvent case for resource Service

Json representation sample

We provide below the json representation of an example of a 'ServiceAttributeValueChangeEvent' notification event object

```
{
    "eventId":"00001",
    "eventTime":"2015-11-16T16:42:25-04:00",
    "eventType":"ServiceAttributeValueChangeEvent",
    "event": {
        "service" :
            {-- SEE Service RESOURCE SAMPLE --}
    }
}
```

Service State Change Event

Notification ServiceStateChangeEvent case for resource Service

Json representation sample

We provide below the json representation of an example of a 'ServiceStateChangeEvent' notification event object

```
{
    "eventId":"00001",
    "eventTime":"2015-11-16T16:42:25-04:00",
    "eventType":"ServiceStateChangeEvent",
    "event": {
        "service" :
            {-- SEE Service RESOURCE SAMPLE --}
    }
}
```

Service Delete Event

Notification ServiceDeleteEvent case for resource Service

Json representation sample

We provide below the json representation of an example of a 'ServiceDeleteEvent' notification event object

```
{
```

```
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceDeleteEvent",
   "event": {
     "service" :
        {-- SEE Service RESOURCE SAMPLE --}
   }
}
```

API OPERATIONS

Remember the following Uniform Contract:

Operation on Entities	Uniform API Operation	Description
Query Entities	GET Resource	GET must be used to retrieve a representation of a resource.
Create Entity	POST Resource	POST must be used to create a new resource
Partial Update of an Entity	PATCH Resource	PATCH must be used to partially update a resource
Remove an Entity	DELETE Resource	DELETE must be used to remove a resource

Filtering and attribute selection rules are described in the TMF REST Design Guidelines Part 1 document.

Notifications are also described in a subsequent section.

Operations on Service

List services

```
GET /service?fields=...&{filtering}
```

Description

This operation list service entities.
Attribute selection is enabled for all first level attributes.
Filtering may be available depending on the compliance level supported by an implementation.

Usage Samples

Here's an example of a request for retrieving multiple services.

Request

GET /tmf-api/serviceInventory/v4/service?fields=id,category,name,state&category=cloud&relatedParty.id=3643
Accept: application/json

Response

```
200

[
  {
    "id": "5351",
    "name": "vCPE serial 1355615",
    "state": "active",
    "category": "CFS"
  },
  {
    "id": "5352",
    "name": "vDPI serial 1355445",
    "state": "active",
    "category": "CFS"
  }
]
```

Retrieve service

GET /service/{id}?fields=...&{filtering}

Description

This operation retrieves a service entity.
Attribute selection is enabled for all first level attributes.
Filtering on sub-resources may be available depending on the compliance level supported by an implementation.

Usage Samples

Here's an example of a request for retrieving a specific service.

Request

GET /tmf-api/serviceInventory/v4/service/5351
Accept: application/json

Response

```
200

{
  "id": "5351",
  "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5351",
  "serviceType": "Cloud",
  "name": "vCPE serial 1355615",
  "description": "Instantiation of vCPE",
  "state": "Active",
  "category": "CFS",
  "isServiceEnabled": true,
  "hasStarted": true,
  "startMode": "1",
  "isStateful": true,
  "startDate": "2018-01-15T12:26:11.747Z",
  "serviceSpecification": {
    "id": "1212",
    "href": "https://mycsp.com:8080/tmf-api/serviceCatalogManagement/v4/serviceSpecification/1212",
    "name": "vCPE",
    "version": "1.0.0",
    "@referredType": "ServiceSpecification"
  },
  "feature": [
    {
      "id": "Feat1",
      "isEnabled": true,
      "name": "ElasticBandwith",
      "featureCharacteritic": [
        {
          "name": "isCapped",
          "value": true,
          "valueType": "boolean"
        }
      ]
    }
  ],
  "serviceCharacteristic": [
    {
      "id": "452-gh6",
      "name": "vCPE",
      "valueType": "object",
      "value": {
        "@type": "JSONSpecification",
        "@schemaLocation": "http://my.schemas/vCPE.schema.json",
        "vCPE_IP": "193.218.236.21",
        "MaxTxRate": 300,
        "TransmitPower": "11 dBm",
        "maxTream": "OFF"
      }
    }
  ],
  "serviceRelationship": [
    {
```

```
          "relationshipType": "DependentOn",
          "service": {
            "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5645",
            "id": "5645",
            "@type": "ServiceRef",
            "@referredType": "Service"
          }
        }
      ],
      "supportingService": [
        {
          "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5885",
          "id": "5885",
          "@type": "ServiceRef",
          "@referredType": "Service"
        }
      ],
      "relatedParty": [
        {
          "href": "https://mycsp.com:8080/tmf-api/partyManagement/v4/party/456",
          "id": "456",
          "name": "John Doe",
          "role": "user",
          "@referredType": "Individual"
        }
      ],
      "serviceOrderItem": [
        {
          "serviceOrderHref": "https://mycsp.com:8080/tmf-api/serviceOrder/v4/serviceOrder/42",
          "serviceOrderId": "42",
          "role": "initiator",
          "@referredType": "ServiceOrder",
          "itemId": "1",
          "itemAction": "add"
        }
      ],
      "place": [
        {
          "href": "https://mycsp.com:8080/tmf-api/geographicAddressManagement/v4/geographicAddress/2435",
          "id": "2435",
          "name": "Customer primary location",
          "@referredType": "GeographicAddress",
          "role": "InstallationAddress"
        }
      ],
      "note": [
        {
          "author": "Harvey Poupon",
          "date": "2018-01-15T12:26:11.748Z",
          "text": "This service was installed automatically, no issues were noted in testing."
        }
      ],
      "@type": "Service",
      "@schemaLocation": "https://mycsp.com:8080/tmf-api/schema/Service/vCPE.schema.json"
    }
```


Create service

POST /service

Description

This operation creates a service entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a Service, including any possible rule conditions and applicable default values. Notice that it is up to an implementer to add additional mandatory attributes.

Mandatory Attributes	Rule
state	
serviceSpecification.id	

Non Mandatory Attributes	Rule
category	
description	
endDate	
feature	
hasStarted	
isBundle	
isServiceEnabled	
isStateful	
name	
note	
place	
relatedEntity	
relatedParty	
serviceCharacteristic	
serviceDate	
serviceOrderItem	
serviceRelationship	
serviceSpecification	
serviceType	
startDate	
startMode	
supportingResource	
supportingService	

Additional Rules

The following table provides additional rules indicating mandatory fields in sub-resources or relationships when creating a Service resource.

Context	Mandatory Sub-Attributes
relatedParty	id, @referredType
serviceRelationship	relationshipType, service
note	date, author, text
place	role
characteristic	name, value
supportingResource	id
supportingService	id
serviceOrderItem	itemId, serviceOrderId
feature	id
serviceSpecification	id

Usage Samples

Here's an example of a request for creating a specific service. In this example the request only passes mandatory attributes.

Request

```
POST /tmf-api/serviceInventory/v4/service
Content-Type: application/json

{
    "serviceType": "Cloud",
    "name": "vCPE serial 1355615",
    "description": "Instantiation of vCPE",
    "state": "Active",
    "category": "CFS",
    "serviceSpecification": {
        "id": "1212",
        "href": "https://mycsp.com:8080/tmf-api/serviceCatalogManagement/v4/serviceSpecification/1212",
        "name": "vCPE",
        "version": "1.0.0",
        "@referredType": "ServiceSpecification"
    },
    "feature": [
        {
            "id": "Feat1",
            "isEnabled": true,
            "name": "ElasticBandwith",
            "featureCharacteritic": [
                {
                    "name": "isCapped",
                    "value": true,
                    "valueType": "boolean"
                }
            ]
        }
```

```
    ],
    "serviceCharacteristic": [
        {
            "id": "452-gh6",
            "name": "vCPE",
            "valueType": "object",
            "value": {
                "@type": "JSONSpecification",
                "@schemaLocation": "http://my.schemas/vCPE.schema.json",
                "vCPE_IP": "193.218.236.21",
                "MaxTxRate": 300,
                "TransmitPower": "11 dBm",
                "maxTream": "OFF"
            }
        }
    ],
    "serviceRelationship": [
        {
            "relationshipType": "DependentOn",
            "service": {
                "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5645",
                "id": "5645",
                "@type": "ServiceRef",
                "@referredType": "Service"
            }
        }
    ],
    "supportingResource": [
        {
            "id": "7171",
            "href": "https://mycsp.com:8080/tmf-api/resourceInventoryManagement/v4/resource/7171",
            "name": "BNG_MUX",
            "value": "01 25 65",
            "@referredType": "VNF"
        }
    ],
    "relatedParty": [
        {
            "href": "https://mycsp.com:8080/tmf-api/partyManagement/v4/party/456",
            "id": "456",
            "name": "Jean Pontus",
            "role": "user",
            "@referredType": "Individual"
        }
    ],
    "note": [
        {
            "author": "Harvey Poupon",
            "date": "2018-01-15T12:26:11.748Z",
            "text": "This service was installed for a rock star."
        }
    ],
    "@type": "Service",
    "@schemaLocation": "https://mycsp.com:8080/tmf-api/schema/Service/vCPE.schema.json"
}
```

Response

```
201

{
  "id": "5351",
  "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5351",
  "serviceType": "Cloud",
  "name": "vCPE serial 1355615",
  "description": "Instantiation of vCPE",
  "state": "Active",
  "category": "CFS",
  "startDate": "2018-01-15T12:26:11.747Z",
  "serviceSpecification": {
    "id": "1212",
    "href": "https://mycsp.com:8080/tmf-api/serviceCatalogManagement/v4/serviceSpecification/1212",
    "name": "vCPE",
    "version": "1.0.0",
    "@referredType": "ServiceSpecification"
  },
  "feature": [
    {
      "id": "Feat1",
      "isEnabled": true,
      "name": "ElasticBandwith",
      "featureCharacteritic": [
        {
          "name": "isCapped",
          "value": true,
          "valueType": "boolean"
        }
      ]
    }
  ],
  "serviceCharacteristic": [
    {
      "id": "452-gh6",
      "name": "vCPE",
      "valueType": "object",
      "value": {
        "@type": "JSONSpecification",
        "@schemaLocation": "http://my.schemas/vCPE.schema.json",
        "vCPE_IP": "193.218.236.21",
        "MaxTxRate": 300,
        "TransmitPower": "11 dBm",
        "maxTream": "OFF"
      }
    }
  ],
  "serviceRelationship": [
    {
      "relationshipType": "DependentOn",
```

```
         "service": {
            "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5645",
            "id": "5645",
            "@type": "ServiceRef",
            "@referredType": "Service"
         }
      }
   ],
   "supportingResource": [
      {
         "id": "7171",
         "href": "https://mycsp.com:8080/tmf-api/resourceInventoryManagement/v4/resource/7171",
         "name": "BNG_MUX",
         "value": "01 25 65",
         "@referredType": "VNF"
      }
   ],
   "relatedParty": [
      {
         "href": "https://mycsp.com:8080/tmf-api/partyManagement/v4/party/456",
         "id": "456",
         "name": "Jean Pontus",
         "role": "user",
         "@referredType": "Individual"
      }
   ],
   "note": [
      {
         "author": "Harvey Poupon",
         "date": "2018-01-15T12:26:11.748Z",
         "text": "This service was installed for a rock star."
      }
   ],
   "@type": "Service",
   "@schemaLocation": "https://mycsp.com:8080/tmf-api/schema/Service/vCPE.schema.json"
}
```

Patch service

PATCH /service/{id}

Description

This operation allows partial updates of a service entity. Support of json/merge
(https://tools.ietf.org/html/rfc7386) is mandatory, support of json/patch (http://tools.ietf.org/html/rfc5789) is
optional.

Note: If the update operation yields to the creation of sub-resources or relationships, the same rules concerning
mandatory sub-resource attributes and default value settings in the POST operation applies to the PATCH
operation. Hence these tables are not repeated here.

Patchable and Non Patchable Attributes

The tables below provide the list of patchable and non patchable attributes, including constraint rules on their usage.

Patchable Attributes	Rule
category	
description	
endDate	
feature	
hasStarted	
isBundle	
isServiceEnabled	
isStateful	
name	
note	
place	
relatedEntity	
relatedParty	
serviceCharacteristic	
serviceOrderItem	
serviceRelationship	
serviceSpecification	
serviceType	
startDate	
startMode	
state	
supportingResource	
supportingService	

Non Patchable Attributes	Rule
id	
href	
serviceDate	

Usage Samples

Here's an example of a request for patching a service. In this example, service status is changed to inactive + supporting service changed.

Request
PATCH /tmf-api/serviceInventory/v4/service/5351 Content-Type: application/merge-patch+json

```json
{
   "id": "5351",
   "state": "inactive",
   "supportingService": [
      {
         "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/6500",
         "id": "6500",
         "@type": "ServiceRef",
         "@referredType": "Service"
      }
   ],
   "@type": "Service"
}
```

Response

```
200
```

```json
{
   "id": "5351",
   "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5351",
   "serviceType": "Cloud",
   "name": "vCPE serial 1355615",
   "description": "Instantiation of vCPE",
   "state": "inactive",
   "category": "CFS",
   "startDate": "2018-01-15T12:26:11.747Z",
   "serviceSpecification": {
      "id": "1212",
      "href": "https://mycsp.com:8080/tmf-api/serviceCatalogManagement/v4/serviceSpecification/1212",
      "name": "vCPE",
      "version": "1.0.0",
      "@referredType": "ServiceSpecification"
   },
   "supportingService": [
      {
         "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/6500",
         "id": "6500",
         "@type": "ServiceRef",
         "@referredType": "Service"
      }
   ],
   "relatedParty": [
      {
         "href": "https://mycsp.com:8080/tmf-api/partyManagement/v4/party/456",
         "id": "456",
         "name": "Jean Pontus",
         "role": "user",
         "@referredType": "Individual"
      }
   ],
   "@type": "Service"
}
```

Here's an example of a request for updating a service using json-patch - CIn this example, service status is changed to inactive + supporting service changed.

Request

```
PATCH /tmf-api/serviceInventory/v4/service/5351
Content-Type: application/json-patch+json

[
  {
    "op": "replace",
    "path": "state",
    "value": "inactive"
  },
  {
    "op": "replace",
    "path": "/supportingService/id",
    "value": 6500
  }
]
```

Response

```
200

{
  "id": "5351",
  "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/5351",
  "serviceType": "Cloud",
  "name": "vCPE serial 1355615",
  "description": "Instantiation of vCPE",
  "state": "inactive",
  "category": "CFS",
  "startDate": "2018-01-15T12:26:11.747Z",
  "serviceSpecification": {
    "id": "1212",
    "href": "https://mycsp.com:8080/tmf-api/serviceCatalogManagement/v4/serviceSpecification/1212",
    "name": "vCPE",
    "version": "1.0.0",
    "@referredType": "ServiceSpecification"
  },
  "supportingService": [
    {
      "href": "https://mycsp.com:8080/tmf-api/serviceInventory/v4/service/6500",
      "id": "6500",
      "@type": "ServiceRef",
      "@referredType": "Service"
    }
  ],
```

```
    "relatedParty": [
      {
        "href": "https://mycsp.com:8080/tmf-api/partyManagement/v4/party/456",
        "id": "456",
        "name": "Jean Pontus",
        "role": "user",
        "@referredType": "Individual"
      }
    ],
    "@type": "Service",
    "@schemaLocation": "https://mycsp.com:8080/tmf-api/schema/Service/vCPE.schema.json"
}
```

Delete service

DELETE /service/{id}

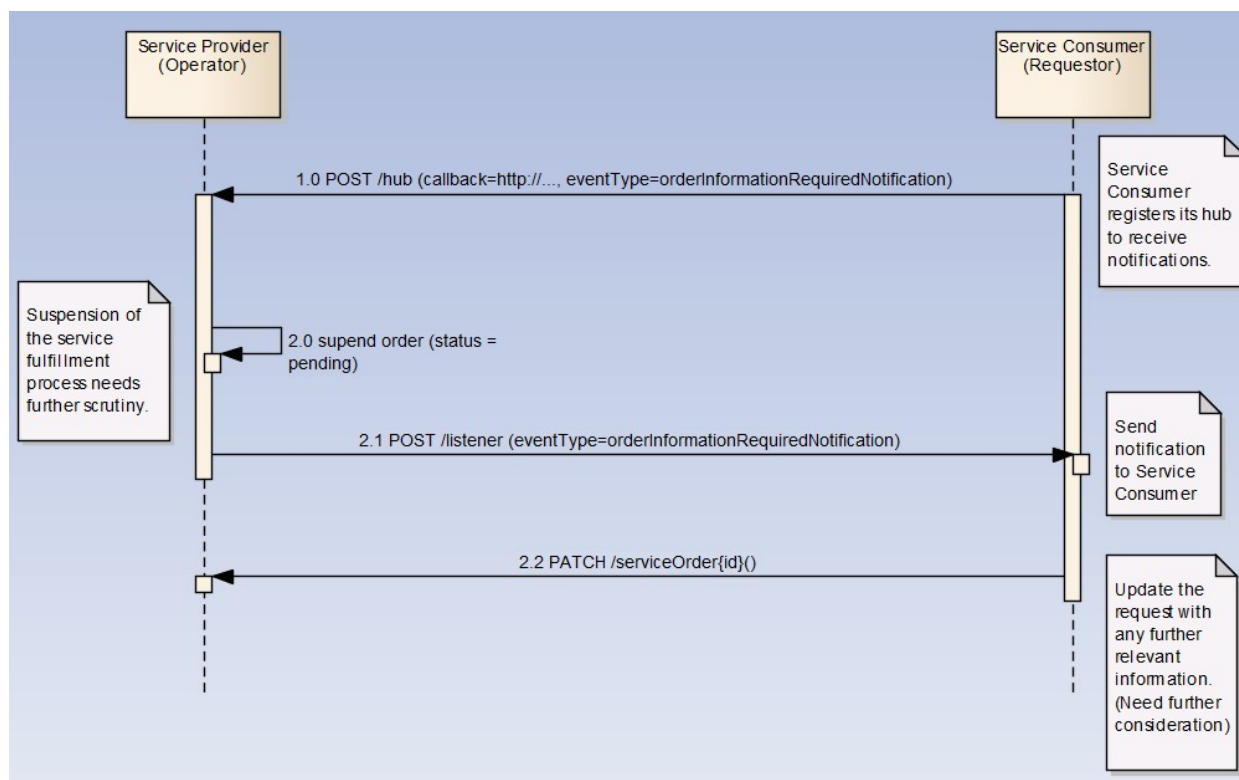**Description**

This operation deletes a service entity.

Usage Samples

Here's an example of a request for deleting a service.

Request
DELETE /tmf-api/serviceInventory/v4/service/5351

Response
204

API NOTIFICATIONS

For every single of operation on the entities use the following templates and provide sample REST notification POST calls.

It is assumed that the Pub/Sub uses the Register and UnRegister mechanisms described in the REST Guidelines part 1. Refer to the guidelines for more details.

Register listener

POST /hub

Description

Sets the communication endpoint address the service instance must use to deliver information about its health state, execution state, failures and metrics. Subsequent POST calls will be rejected by the service if it does not support multiple listeners. In this case DELETE /api/hub/{id} must be called before an endpoint can be created again.

Behavior

Returns HTTP/1.1 status code 204 if the request was successful.

Returns HTTP/1.1 status code 409 if request is not successful.

Usage Samples

Here's an example of a request for registering a listener.

Request
POST /api/hub Accept: application/json {"callback": "http://in.listener.com"}
Response
201 Content-Type: application/json Location: /api/hub/42 {"id":"42","callback":"http://in.listener.com","query":null}

Unregister listener

DELETE /hub/{id}

Description

Clears the communication endpoint address that was set by creating the Hub.

Behavior

Returns HTTP/1.1 status code 204 if the request was successful.

Returns HTTP/1.1 status code 404 if the resource is not found.

Usage Samples

Here's an example of a request for un-registering a listener.

Request
DELETE /api/hub/42 Accept: application/json
Response
204

Publish Event to listener

POST /client/listener

Description

Clears the communication endpoint address that was set by creating the Hub.

Provides to a registered listener the description of the event that was raised. The /client/listener url is the callback url passed when registering the listener.

Behavior

Returns HTTP/1.1 status code 201 if the service is able to set the configuration.

Usage Samples

Here's an example of a notification received by the listener. In this example "EVENT TYPE" should be replaced by one of the notification types supported by this API (see Notification resources Models section) and EVENT BODY refers to the data structure of the given notification type.

Request

POST /client/listener
Accept: application/json
 	
{
   "event": {
        EVENT BODY
     },
   "eventType": "EVENT_TYPE"
}

Response
201

For detailed examples on the general TM Forum notification mechanism, see the TMF REST Design Guidelines Part 1 document.

Acknowledgements

Release History

Release Number	Date	Release led by:	Description
Release 16.5 Version 1.0.0	15-Apr-2016	Pierre Gauthier TM Forum pgauthier@tmforum.org Nicoleta Stoica Vodafone Nicoleta.stoica@vodafone.com Mariano Belaunde Orange Labs	Generated using the API data model
Release 18.0 Version 2.0.0	31-Jan-2018	Pierre Gauthier TM Forum pgauthier@tmforum.org Ludovic Robert Orange ludovic.robert@orange.com	Alignment with TMF open API guidelines 3.0 Review with SID Team (Cécile Ludwichowski)
Release 18.5 Version 3.0.0	13-Jan-2018	Jonathan Goldberg Amdocs Jonathan.Goldberg@amdocs.com	Schema alignment for NaaS APIs in release 18.5
Release 20.0 Version 4.0	05-May-2020	Pierre Gauthier TM Forum pgauthier@tmforum.org Ludovic Robert Orange ludovic.robert@orange.com	Alignment with TMF open API guidelines 4.0

Contributors to Document

Pierre Gauthier	TM Forum
Nicoleta Stoica	Vodafone (formerly)
Mariano Belaunde	Orange
Ludovic Robert	Orange
Jonathan Goldberg	Amdocs