ITUPublications
Recommendations

International Telecommunication Union
Standardization Sector

Recommendation

ITU-T Y.4703 (03/2024)

SERIES Y: Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities

Internet of things and smart cities and communities – Management, control and performance

---

Internet of things service management application programming interface Representational State Transfer specification

ITU-T Y-SERIES RECOMMENDATIONS

Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities

GLOBAL INFORMATION INFRASTRUCTURE	Y.100-Y.999
INTERNET PROTOCOL ASPECTS	Y.1000-Y.1999
NEXT GENERATION NETWORKS	Y.2000-Y.2999
FUTURE NETWORKS	Y.3000-Y.3499
CLOUD COMPUTING	Y.3500-Y.3599
BIG DATA	Y.3600-Y.3799
QUANTUM KEY DISTRIBUTION NETWORKS	Y.3800-Y.3999
INTERNET OF THINGS AND SMART CITIES AND COMMUNITIES	Y.4000-Y.4999
General	Y.4000-Y.4049
Definitions and terminologies	Y.4050-Y.4099
Requirements and use cases	Y.4100-Y.4249
Infrastructure, connectivity and networks	Y.4250-Y.4399
Frameworks, architectures and protocols	Y.4400-Y.4549
Services, applications, computation and data processing	Y.4550-Y.4699
Management, control and performance	**Y.4700-Y.4799**
Identification and security	Y.4800-Y.4899
Evaluation and assessment	Y.4900-Y.4999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T Y.4703

Internet of things service management application programming interface Representational State Transfer specification

Summary

Recommendation ITU-T Y.4703 specifies the Representational State Transfer (REST) Internet of things (IoT) service management applications programming interface (API) user guide. It includes the model definition as well as all available operations.

History*

Edition	Recommendation	Approval	Study Group	Unique ID
1.0	ITU-T Y.4703	2024-03-15	20	11.1002/1000/15692

Keywords

API, IoT, REST, service management, TM Forum.

* To access the Recommendation, type the URL https://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/. Implementers should also be aware that the organization that originated the technically equivalent document listed in the Bibliography may have received notices of intellectual property required for the implementation of this Recommendation.

Table of Contents

Recommendation ITU-T Y.4703

Internet of things service management application programming interface Representational State Transfer specification

1 Scope

This Recommendation presents a strong framework for the management of Internet of things (IoT) services leveraging existing globally adopted and publicly available TM Forum (TMF) Open APIs (application programming interfaces). The business case for IoT requires that IoT platforms are highly scalable and interoperable supporting many different types of services. This Recommendation demonstrates how an organizations existing technology state can be reused to manage IoT services through their end to end lifecycle.

NOTE – The Recommendation is technically aligned with TMF 914 IoT Service Management API Component Suite v4.0.0 [b-TMF 914].

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[IETF RFC 7386] IETF RFC 7386 (2014), *JSON Merge Patch*.

[IETF RFC 5789] IETF RFC 5789 (2010), *PATCH Method for HTTP*.

3 Definitions

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

6LoWPAN	IPv6 over Low-Power Wireless Personal Area Networks
API	Application Programming Interface
CoAP	Constrained Application Protocol
ID	Identifier
IoT	Internet of Things
MSISDN	Mobile Station International Subscriber Directory Number
NGSI	Next Generation Service Interface
OSS	Operations Support System
RFSS	Resource Facing Service Specification
URI	Uniform Resource Identifier

5 Conventions

5.1 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 are described below. Polymorphism in collections occurs when entities inherit from base entities, for instance a BillingAccount and SettlementAccount inheriting properties from the abstract Account entity.

The @type attribute provides a way to represent the actual class type of an entity. For example, within a list of Account instances some may be instances of BillingAccount where other could be instances of SettlementAccount. 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 AccountRef 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 BillingAccountRef or SettlementAccountRef, 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.

6 Resource Model

6.1 Managed Entity and Task Resource Models

6.1.1 IoT Service Resource

Resource model

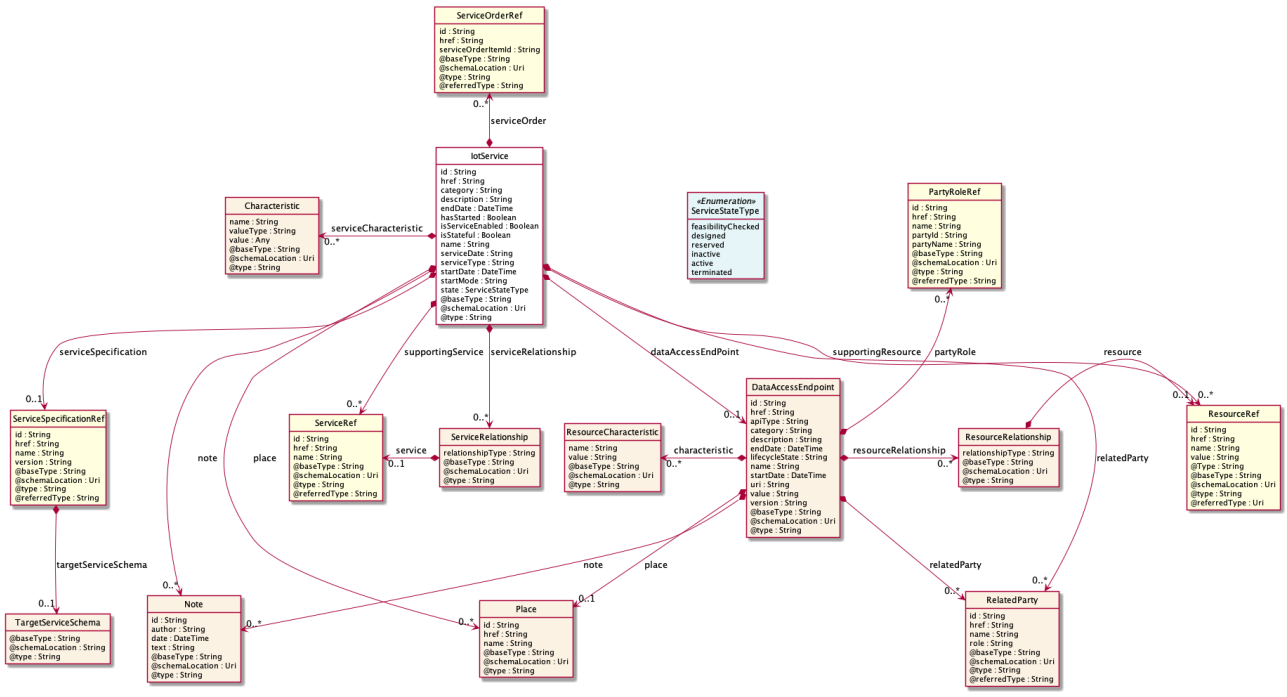

Figure 1 – IoT service resource model schematic

Field descriptions

IotService fields

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.
isServiceEnabled	A Boolean. If FALSE, this particular Service has NOT been enabled for use.
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).
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.
category	A string. Is it a customer facing or resource facing service?
href	A string. Reference of the service.
id	A string. Unique identifier of the service.
name	A string. Name of the service.
serviceType	A string. Business type of the service.
dataAccessEndPoint	A data access endpoint (DataAccessEndpoint). This is the endpoint exposed by the IoT Device to authorized users. A list of data access endpoints (DataAccessEndpoint[*]). Each endpoint is exposed by the IoT device to authorized users.
note	A list of notes (Note [*]). A list of notes made on this service.
serviceOrder	A list of service order references (ServiceOrderRef [*]). A list of service orders related to this service.
place	A list of places (Place [*]). A list of places (Place [*]). Used to define a place useful for the service (for example a delivery geographical place).
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 [*]).
serviceRelationship	A list of service relationships (ServiceRelationship [*]). A list of service relationships (ServiceRelationship [*]). Describes links with other service(s) in the inventory (useful for describing relies-on, relies-from between CFS for example).
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: [feasibilityChecked], [designed].
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 references (ServiceRef [*]). 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.

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.

DataAccessEndpoint sub-resource

This is the endpoint exposed by the IoT Device to authorized users.

category	A string. Category of the concrete resource, such as: Gold, Silver for Mobile Station International Subscriber Directory Number (MSISDN) concrete resource.
description	A string. Free-text description of the resource.
endDate	A date time (DateTime). The date till the resource is effective.
href	A string. The uniform resource identifier (URI) for the object itself.
id	A string. Identifier of an instance of the resource. Required to be unique within the resource type. Used in URIs as the identifier for specific instances of a type.
lifecycleState	A string. The life cycle state of the resource.
name	A string. A string used to give a name to the resource.
startDate	A date time (DateTime). A date time (DateTime). The date from which the resource is effective.
value	A string. The value of the logical resource, such as: 0044746712345 for an MSISDN.
version	A string. A field that identifies the specific version of an instance of a resource.
apiType	A string.
uri	A string. URI for using the data access API.
characteristic	A list of resource characteristics (ResourceCharacteristic [*]).
note	A list of notes (Note [*]). Extra information about a given entity.
partyRole	A list of party role references (PartyRoleRef [*]). A party role represents the part played by a party in a given context.
place	A place (Place). Place reference. Place defines the places where the products are sold or delivered.

relatedParty	A list of related parties (RelatedParty [*]). Related Entity reference. A related party defines party or party role linked to a specific entity.
resourceRelationship	A list of resource relationships (ResourceRelationship [*]). Describes links between resources.

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.

Place sub-resource

Place reference. Place defines the places where the products are sold or delivered.

href	A string. Unique reference of the place.
id	A string. Unique identifier of the place.
name	A string. A user-friendly name for the place, such as [Paris Store], [London Store], [Main Home].

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.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.
@referredType	A string. The actual type of the target instance when needed for disambiguation.
role	A string. Role played by the related party.

ResourceCharacteristic sub-resource

name	A string. name of the characteristic.
value	A string. value of the characteristic.

ResourceRelationship sub-resource

Describes links between resources.

relationshipType	A string. Semantic of the relationship.
resource	A resource reference (ResourceRef). A reference to the resource.

ServiceRelationship sub-resource

Describes links with services of the same category (useful for bundled services).

relationshipType	A string. The type of relationship (e.g., depends on, enables).
service	A service reference (ServiceRef). The service being referred to.

TargetServiceSchema sub-resource

The reference object to the schema and type of target service which is described by service specification.

@schemaLocation	A string. This field provides a link to the schema describing the target service.
@type	A string. Class type of the target service.

PartyRoleRef relationship

Party role reference. A party role represents the part played by a party in a given context.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Reference of the product.
id	A string. Unique identifier of the product.
name	A string. The name of the referred party role.
partyId	A string. The identifier of the engaged party that is linked to the PartyRole object.
partyName	A string. The name of the engaged party that is linked to the PartyRole object.

ResourceRef relationship

Information about a resource that holds realizes the product that is linked to the bucket balance.

@Type	A string. Indicates the type of resource.
href	A string. Reference to the party.
id	A string. Unique identifier of the related party.
name	A string. Name of the resource.
value	A string. The resource value that can be used to identify a resource with a public key (e.g.,: a tel nr, an msisdn).

ServiceOrderRef relationship

Service Order reference. Useful to understand the which was the Service order through which the service was instantiated in the service inventory.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Reference of the Service Order.
id	A string. Unique identifier of the Service Order.
serviceOrderItemId	A string. Unique identifier of the Service Order Item within a service order, not populated if this is a reference to a service order.

ServiceRef relationship

Service reference, for when Service is used by other entities.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.

name	A string. Name of the related entity.

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.
href	A string. Reference of the serviceSpecification.
id	A string. Unique identifier of the service specification.
name	A string. Name of the requiredServiceSpecification.
targetServiceSchema	A target service schema (TargetServiceSchema). A target service schema reference (TargetServiceSchemaRef). The reference object to the schema and type of target service which is described by service specification.
version	A string. Service specification version.

Json representation sample

The following is an example json representation of an 'IotService' resource object.


```
{
   "description": "This IoT service provides the access to data generated by IoT sensors.",
"@type": "IoTDataService",
"@schemaLocation" : "https://www.mandint.org/IoTDataService.json",
   "endDate": "2019-05-13T00:00",
   "hasStarted": true,
   "isServiceEnabled": true,
   "isStateful": true,
   "serviceDate": "2019-05-13T00:00",
   "startDate": "2019-05-13T00:00",
   "startMode": "1",
   "category": "IoT Data Provider",
   "href": "https://host:port/tmf-api/iotService/v1/iotService/8520",
   "id": "8520",
   "name": "IoT Data Service",
   "serviceType": "IoTDataService",
   "dataAccessEndPoint": [
{
"category": "Gold",
"description": "This is a temperature sensor using CoAP and 6LoWPAN.",
"endDate": "2019-05-13T00:00",
"href": "https://host:port/tmf-api/iotDevice/v1/iotDevice/temp_3",
"id": "3",
"lifecycleState": "InService",
"name": "temp_3",
"startDate": "2019-05-03T00:00",
"value": "0041227744222",
"version": "1.0",
"apiType": "NGSI",
```

```
"uri": "https://host:port/tmf-api/iotDevice/v1/iotDevice/temp_3/dataAccessEndpoint/3"
},
{
"category": "Gold",
"description": "This is a temperature sensor using CoAP and 6LoWPAN.",
"endDate": "2019-05-13T00:00",
"href": "https://host:port/tmf-api/iotDevice/v1/iotDevice/temp_4",
"id": "4",
"lifecycleState": "InService",
"name": "temp_4",
"startDate": "2019-05-03T00:00",
"value": "0041227744222",
"version": "1.0",
"apiType": "MQTT",
"uri": "https://host:port/tmf-api/iotDevice/v1/iotDevice/temp_4/dataAccessEndpoint/4"

}
],
"note": [
      {
"author": "Cedric Crettaz",
"date": "2019-05-13T00:00",
"id": "txt002",
"text": "This is an IoT service for temperature sensors."
}
   ],
   "serviceOrder": [
      {
"@referredType": "IoTDataService",
"href": " https://www.mandint.org/IoTDataServiceOrder",
"id": "85",
"serviceOrderItemId": "8520"
}
   ],
   "place": [
      {
"href": "Chemin du Champ-Baron 3",
"id": "1209",
"name": "Geneva Office"
}
   ],
   "relatedParty": [
      {
```

```
"@referredType": "temperatureSensor",
"href": "https://www.mandint.org/temperatureSensor",
"id": "CoapTempSensor",
"name": "Mandat International",
"role": "vendor"
}
   ],
   "serviceCharacteristic": [
      {
"name": "proxyIP",
"value": "2001:620:607:5110::31",
"valueType": "string"
}
   ],
   "serviceRelationship": [
      {
"relationshipType": "depends on",
"service":  {
"@referredType": "5GSliceNaaS",
"href": " https://host:port/tmf-api/iotService/v1/iotService/8525",
"id": "8525",
"name": "5G Slice NaaS"
}
}
   ],
   "serviceSpecification": {
"@referredType": "IoTDataService",
"href": "https://www.mandint.org/serviceSpecification/IOTDataServiceSpecification",
"id": "501",
"name": "serviceSpec501",
"targetServiceSchema": {
"@schemaLocation": "
https://www.mandint.org/serviceSpecification/IOTDataServiceSpecificationSchema.json",
"@type": "IoTDataService"
},
"version": "1.0.0"
},
   "state": "active",
   "supportingResource": [
{
"@Type": "IoTAgent",
"href": "https://www.mandint.org/IotDeviceMangementAPI/IoTAgent/building1",
"id": "iotAgent_1",
```

```
"name": "building_1",
"value": "0041227744222"
},
{
"@Type": "IoTAgent",
"href": "https://www.mandint.org/IotDeviceMangementAPI/IoTAgent/building2",
"id": "iotAgent_2",
"name": "building_2",
"value": "0041227744111"
}
  ],
  "supportingService": [
    {
"@referredType": "5GSliceNaaS",
"href": " https://host:port/tmf-api/iotService/v1/iotService/8525",
"id": "8525",
"name": "5G Slice NaaS"
}
  ]
}
```


6.1.2 Service Catalog resource

The root entity for service catalog management.

A service catalog is a group of service specifications made available through service candidates that an organization provides to the consumers (internal consumers like its employees or B2B customers or B2C customers).

A service catalog typically includes name, description and time period that is valid for. It will have a list of ServiceCandidate catalog items. A ServiceCandidate is an entity that makes a ServiceSpecification available to a catalog.

A ServiceCandidate and its associated ServiceSpecification may be "published" -made visible- in any number of ServiceCatalogs, or in none.

Resource model

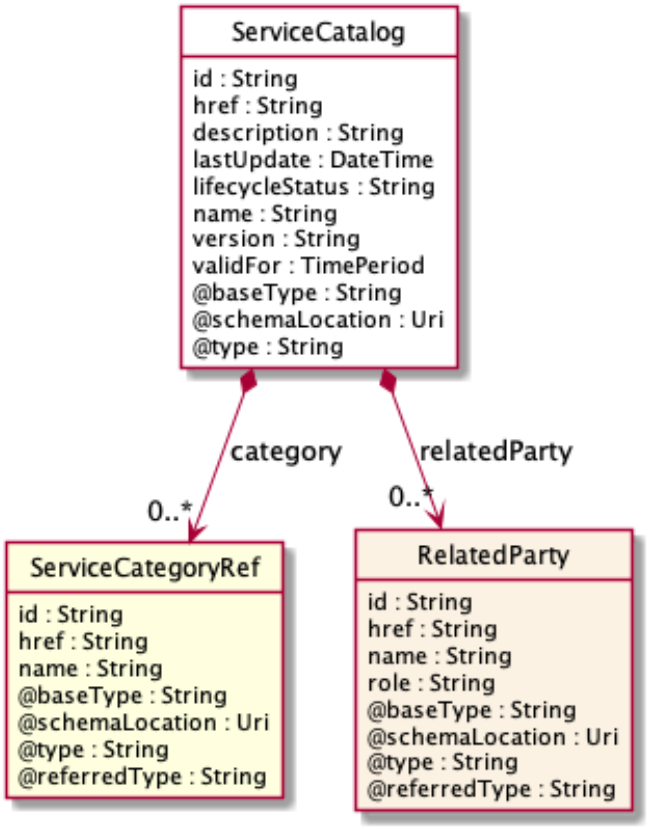

Figure 2 – Service Catalog Resource Model, Schematic

Field descriptions

ServiceCatalog fields

category	A list of service category references (ServiceCategoryRef [*]). List of service categories associated with this catalog.
description	A string. Description of this catalog.
href	A string. Unique reference of the service catalog.
id	A string. Unique identifier of the ServiceCatalog.
lastUpdate	A date time (DateTime). Date and time of the last update.
lifecycleStatus	A string. Used to indicate the current lifecycle status.
name	A string. Name of the service catalog.
relatedParty	A list of related parties (RelatedParty [*]). List of parties or party roles related to this category.
validFor	A time period. The period for which the service catalog is valid.
version	A string. ServiceCatalog version.

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.
href	A string. Reference of the related entity.

id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.
@referredType	A string. The actual type of the target instance when needed for disambiguation.
role	A string. Role played by the related party.

ServiceCategoryRef relationship

The service category resource is used to group service candidates in logical containers. Categories can contain other categories.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Hypertext Reference of the category.
id	A string. Unique identifier of category.
name	A string. Name of the category.

Json representation sample

The following is an example representation of a 'ServiceCatalog' resource object.

```
{
  "category": [
    {}
  ],
  "description": "This  service catalog ...",
  "href": "https:/host:port/tmf-api/serviceCatalog/v1/serviceCatalog/3035",
  "id": "3035",
  "lastUpdate": "2019-05-13T00:00",
  "lifecycleStatus": "a string ...",
  "name": "a string ...",
  "relatedParty": [
    {}
  ],
  "validFor": {},
  "version": "a string ..."
}
```

6.1.3 Service Category resource

The service category resource is used to group service candidates in logical containers. Categories can contain other categories.

Resource model

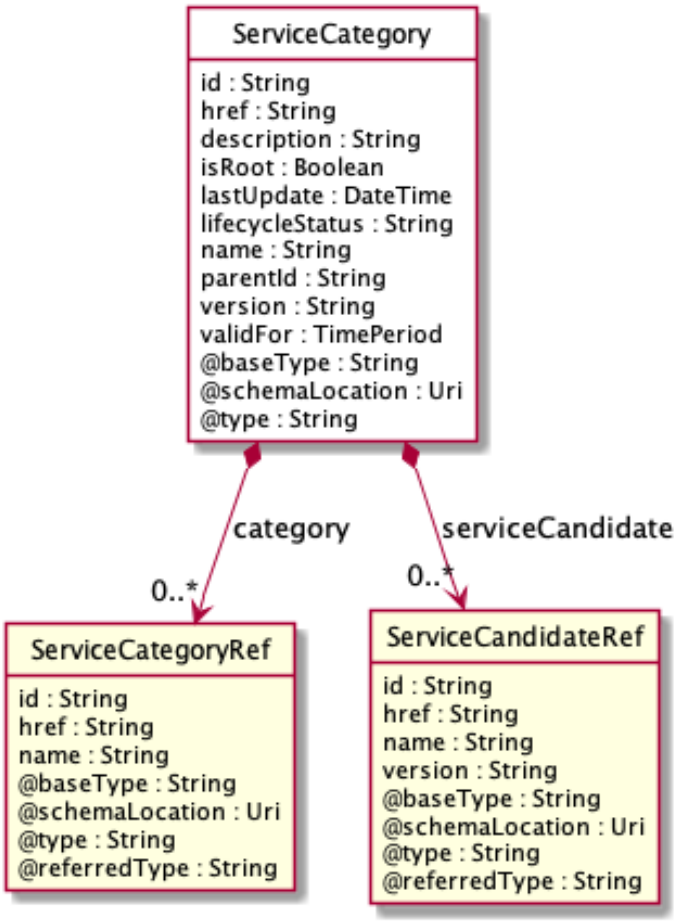

Figure 3 – Service Category Resource Model, Schematic

Field descriptions

ServiceCategory fields

category	A list of service category references (ServiceCategoryRef [*]). List of child categories in the tree for in this category.
description	A string. Description of the category.
href	A string. Reference of the category.
id	A string. Unique identifier of the category.
isRoot	A Boolean. If true, this Boolean indicates that the category is a root of categories.
lastUpdate	A date time (DateTime). Date and time of the last update.
lifecycleStatus	A string. Used to indicate the current lifecycle status.
name	A string. Name of the category.
parentId	A string. Unique identifier of the parent category.
serviceCandidate	A list of service candidate references (ServiceCandidateRef [*]). List of service candidates associated with this category.
validFor	A time period. The period for which the category is valid.
version	A string. ServiceCategory version.

ServiceCandidateRef relationship

ServiceCandidate reference. ServiceCandidate is an entity that makes a ServiceSpecification available to a catalog.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Unique reference of the service candidate.
id	A string. Unique identifier of the service candidate.
name	A string. Name of the service candidate.
version	A string. Version of the service candidate.

ServiceCategoryRef relationship

The service category resource is used to group service candidates in logical containers. Categories can contain other categories.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Hypertext Reference of the category.
id	A string. Unique identifier of category.
name	A string. Name of the category.

Json representation sample

The following is an example json representation of a 'ServiceCategory' resource object.

```
{
  "category": [
    {}
  ],
  "description": "This  service category ...",
  "href": "https:/host:port/tmf-api/serviceCategory/v1/serviceCategory/6889",
  "id": "6889",
  "isRoot": false,
  "lastUpdate": "2019-05-13T00:00",
  "lifecycleStatus": "a string ...",
  "name": "a string ...",
  "parentId": "203",
  "serviceCandidate": [
    {}
  ],
  "validFor": {},
  "version": "a string ..."
}
```

6.1.4 Service Candidate resource

ServiceCandidate is an entity that makes a service specification available to a catalog. AServiceCandidate and its associated service specification may be published -made visible- in any number of service catalogs, or in none. One service specification can be composed of other service specifications.

Resource model

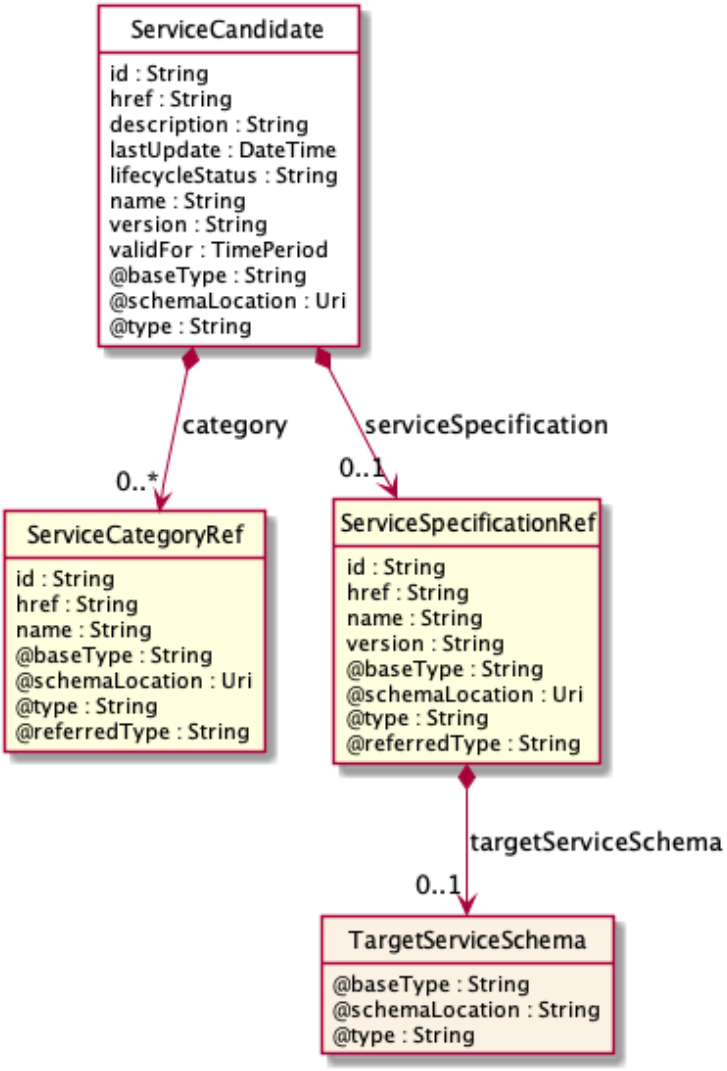

Figure 4 – Service Candidate Resource Model, Schematic

Field descriptions

ServiceCandidate fields

category	A list of service category references (ServiceCategoryRef [*]). List of categories for this candidate.
description	A string. Description of this REST resource.
href	A string. Hyperlink reference to this REST resource.
id	A string. Unique identifier of this REST resource.
lastUpdate	A date time (DateTime). Date and time of the last update of this REST resource.
lifecycleStatus	A string. Used to indicate the current lifecycle status of the service candidate.
name	A string. Name given to this REST resource.
serviceSpecification	A service specification reference (ServiceSpecificationRef). The service specification implied by this candidate.

validFor	A period of time. The period for which this REST resource is valid.
version	A string. the version of service candidate.

TargetServiceSchema sub-resource

The reference object to the schema and type of target service which is described by service specification.

@schemaLocation	A string. This field provides a link to the schema describing the target service.
@type	A string. Class type of the target service.

ServiceCategoryRef relationship

The service category resource is used to group service candidates in logical containers. Categories can contain other categories.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Hypertext Reference of the category.
id	A string. Unique identifier of category.
name	A string. Name of the category.

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.
href	A string. Reference of the serviceSpecification.
id	A string. Unique identifier of the service specification.
name	A string. Name of the requiredServiceSpecification.
targetServiceSchema	A target service schema (TargetServiceSchema). A target service schema reference (TargetServiceSchemaRef). The reference object to the schema and type of target service which is described by service specification.
version	A string. Service specification version.

Json representation sample

The following is an example json representation of a 'ServiceCandidate' resource object.


```
{
  "category": [
    {}
  ],
  "description": "This  service candidate ...",
  "href": "https:/host:port/tmf-api/serviceCandidate/v1/serviceCandidate/9315",
  "id": "9315",
  "lastUpdate": "2019-05-13T00:00",
  "lifecycleStatus": "a string ...",
  "name": "a string ...",
  "serviceSpecification": {},
  "validFor": {},
```

```
    "version": "a string ..."
}
```

6.1.5 Service Qualification resource

ServiceQualification is used to perform a technical eligibility. It allows to retrieve a list of services that are technically available in the context of the interaction (place, party, service characteristics, ...).

Resource model

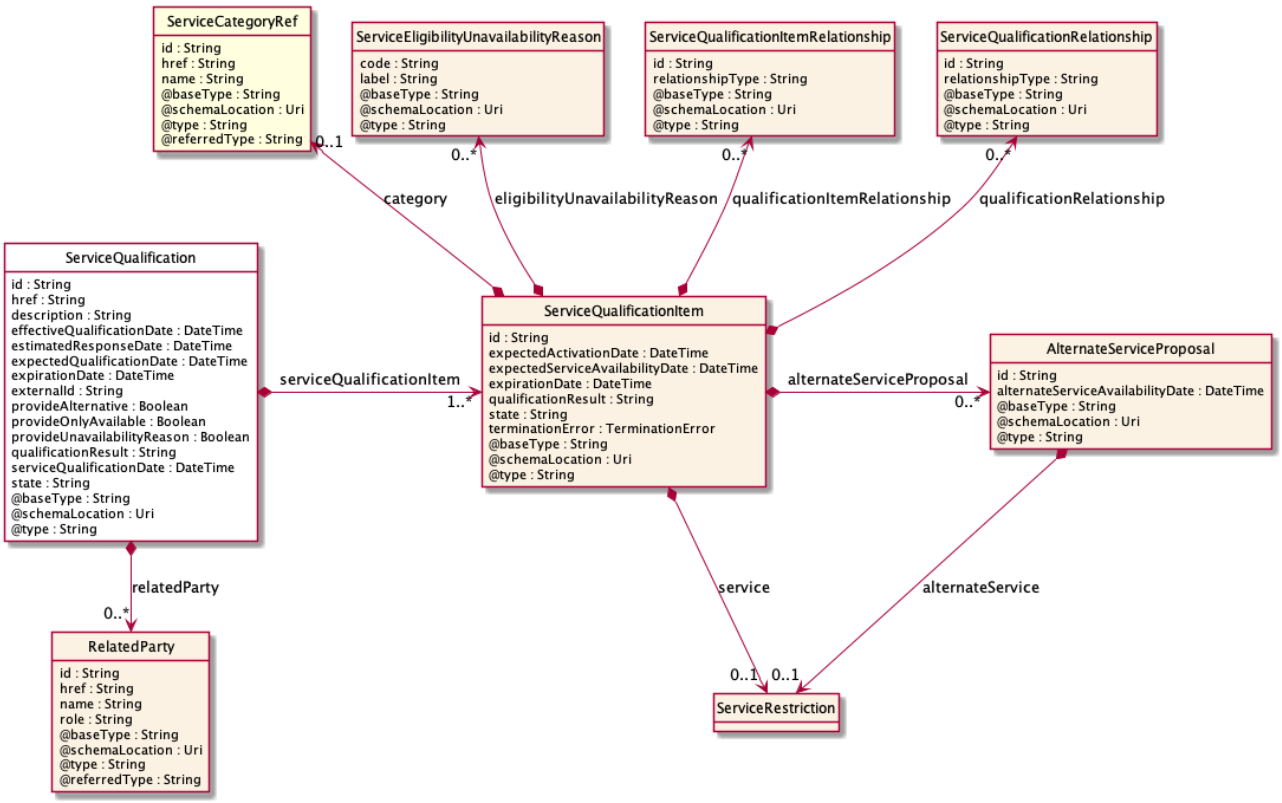

Figure 5 – Service Qualification Resource Model, Schematic

Field descriptions

ServiceQualification fields

description	A string. Description of the serviceQualification.
effectiveQualificationDate	A date time (DateTime). Effective date to serviceQualification completion.
estimatedResponseDate	A date time (DateTime). Date when the requester expects to provide an answer for the qualification request.
expectedQualificationDate	A date time (DateTime). A date (DateTime). Deadline date when the requester expected a qualification answer.
expirationDate	A date time (DateTime). Date when the qualification response expires.
externalId	A string. Identifier provided by the requester.
href	A string. Hyperlink to access the serviceQualification.
id	A string. Unique identifier of the serviceQualification resource.

provideAlternative	A Boolean. When the value is TRUE means that alternative solutions should be provided.
provideOnlyAvailable	A Boolean. When the value is TRUE means that only available service must be listed in the response.
provideUnavailabilityReason	A Boolean. When the value is TRUE means that unavailability reasons are expected for non-available service.
qualificationResult	A string. Qualification results for this serviceQualification. It could be: qualified (all qualification item are qualified), alternate (At least one item alternate and no item with unqualified), unqualified (At least one item unqualified).
relatedParty	A list of related parties (RelatedParty [*]). A list of related party references, defines party or party role linked to this request.
serviceQualificationDate	A date time (DateTime). Date when the serviceQualification was submitted.
serviceQualificationItem	A list of service qualification items (ServiceQualificationItem [1..*]).
state	A string. State of the serviceQualification (acknowledged, inProgress, terminatedWithError, done).

AlternateServiceProposal sub-resource

Alternate service proposal is used when the requested service is not available with characteristic and date asked for. An alternate proposal could be a distinct serviceSpecification close to requested one or same as requested but with a different activation date.

alternateService	A service restriction (ServiceRestriction).
alternateServiceAvailabilityDate	A date time (DateTime). Alternate availability date in case seller is not able to meet requested expected availability date for the service.
id	A string. Identifier of an alternate service proposal.

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.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.
@referredType	A string. The actual type of the target instance when needed for disambiguation.
role	A string. Role played by the related party.

ServiceEligibilityUnavailabilityReason sub-resource

Reason for eligibility result if the ServiceQualification result is no (meaning the Service is not available).

code	A string. Unavailability reason code.
label	A string. Unavailability reason label.

ServiceQualificationItem sub-resource

A ServiceQualificationItem relates to a specific service being checked in a qualification operation.

alternateServiceProposal	A list of alternate service proposals (AlternateServiceProposal [*]). Alternate service proposal is used when the requested service is not available with characteristic and date asked for. An alternate proposal could be a distinct service Spec close to requested one or same as requested but with a different availability date.
category	A service category reference (ServiceCategoryRef). The category resource is used to group product offerings, service and resource candidates in logical containers. Categories can contain other categories and/or product offerings, resource or service candidates.
eligibilityUnavailabilityReason	A list of service eligibility unavailability reasons (ServiceEligibilityUnavailabilityReason [*]). A list of eligibility unavailability reasons (EligibilityUnavailabilityReason [*]). Reason for eligibility result if the serviceQualification result is no (meaning the service is not available).
expectedActivationDate	A date time (DateTime). The date when the service is expected to be activated.
expectedServiceAvailabilityDate	A date time (DateTime). Date when the requester looks for service availability.
expirationDate	A date time (DateTime). Date when the qualification item response expires.
id	A string. Id of the Service Qualification Item.
qualificationItemRelationship	A list of service qualification item relationships (ServiceQualificationItemRelationship [*]). A list of qualification item relationships used to describe relationship between serviceQualification item from the same serviceQualification.
qualificationRelationship	A list of service qualification relationships (ServiceQualificationRelationship [*]). Structure used to describe relationship between serviceQualification item from the same serviceQualification.
qualificationResult	A string. Qualification result for serviceQualification item. It could be: - qualified (request service are available), - unqualified (requested not available and not alternate available), - alternate (requested not available but proposal available).

service	A service restriction (ServiceRestriction). Configure the service characteristics (only configurable characteristics and necessary only if a non-default value is selected) and/or identify the service that needs to be modified/deleted.
state	A string. State of the serviceQualification item (acknowledged, inProgress, terminatedWithError, done).
terminationError	A list of termination errors (TerminationError [*]). If qualificationItem has not been done properly this lists the error(s) that caused termination of the qualification.

ServiceQualificationItemRelationship sub-resource

Structure used to describe relationship between serviceQualification items from the same serviceQualification.

id	A string. Id of the serviceQualificationItem (from the same serviceQualificationItem).
relationshipType	*A string. Type of relationship (ex: reliesOn, connectedTo, etc...).*

ServiceQualificationRelationship sub-resource

id	A string. The id of the target qualification pointed to by this relationship.
relationshipType	A string. The type of relationship.

TerminationError sub-resource

This indicates an error that caused a qualificationItem to be terminated.

id	A string. Unique identifier of the termination error.
value	A string. Text to describe the termination error - for example: Unable to proceed to qualification because incomplete information provided.

ServiceCategoryRef relationship

The service category resource is used to group service candidates in logical containers. Categories can contain other categories.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Hypertext Reference of the category.
id	A string. Unique identifier of category.
name	A string. Name of the category.

Json representation sample

The following is an example json representation of a 'ServiceQualification' resource object.


```
{
  "description": "This  service qualification ...",
  "effectiveQualificationDate": "2019-05-13T00:00",
  "estimatedResponseDate": "2019-05-13T00:00",
  "expectedQualificationDate": "2019-05-13T00:00",
  "expirationDate": "2019-05-13T00:00",
  "externalId": "506",
```

```
    "href": "https:/host:port/tmf-api/serviceQualification/v1/serviceQualification/7775",
    "id": "7775",
    "provideAlternative": true,
    "provideOnlyAvailable": true,
    "provideUnavailabilityReason": true,
    "qualificationResult": "a string ...",
    "relatedParty": [
      {}
    ],
    "serviceQualificationDate": "2019-05-13T00:00",
    "serviceQualificationItem": [
      {}
    ],
    "state": "a string ..."
  }
```


6.1.6 Service Problem resource

The problem information for Middle B which is abstracted in the service layer from the issued event information by First B.

Resource model

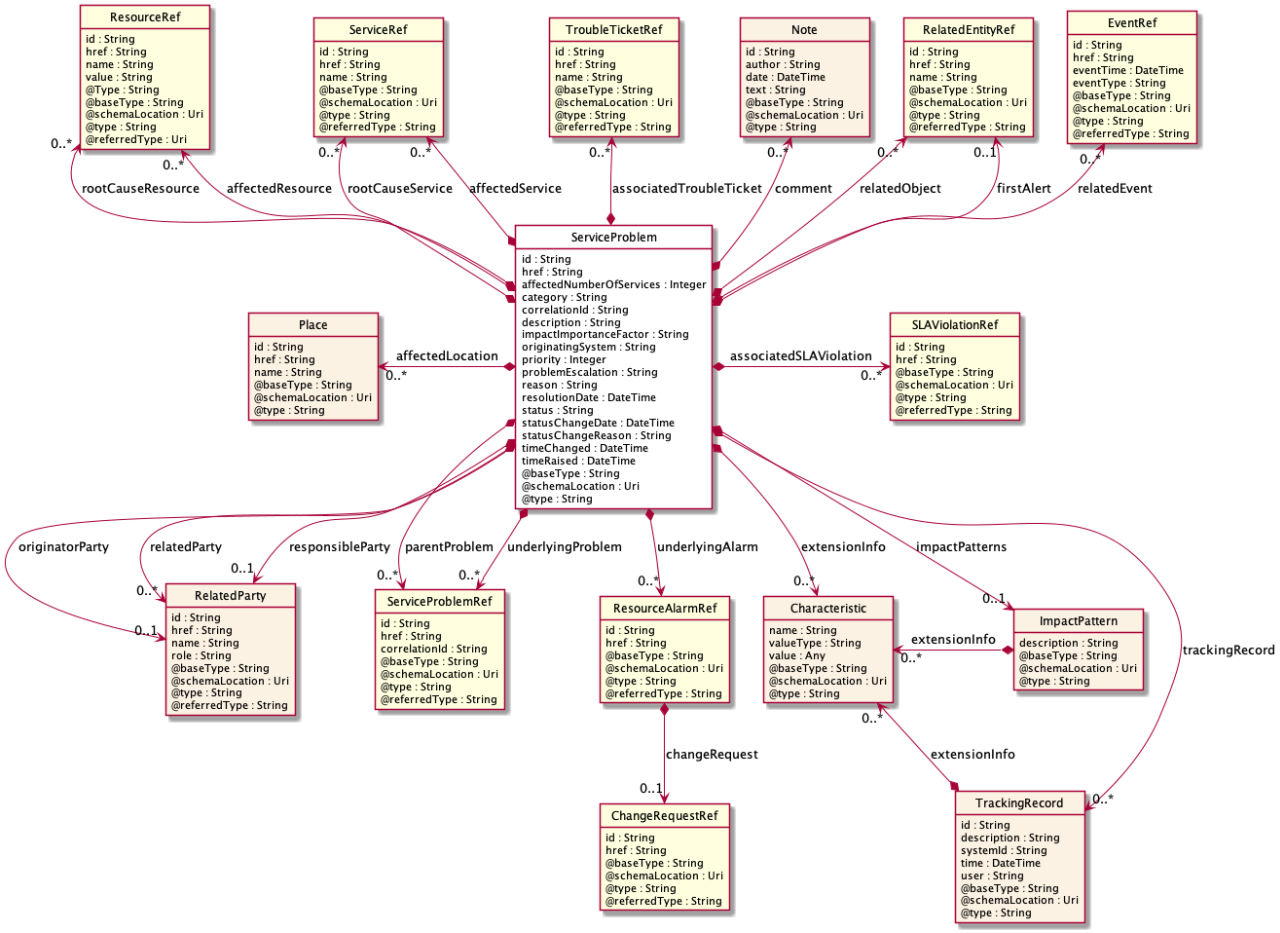

Figure 6 – Service Problem Resource Model, Schematic

Field descriptions

ServiceProblem fields

affectedLocation	A list of places (Place [*]). A list of the locations affected by the problem. At least one of affectedResource, affectedService or affectedLocation should be present.
affectedNumberOfServices	An integer. Number of affected services.
affectedResource	A list of resource references (ResourceRef [*]). A list of the resources affected by the problem. At least one of affectedResource, affectedService or affectedLocation should be present.
affectedService	A list of service references (ServiceRef [*]). List of affected services. At least one of affectedResource, affectedService or affectedLocation should be present.
associatedSLAViolation	A list of s l a violation references (SLAViolationRef [*]). A List of SLA violations associated with this problem.
associatedTroubleTicket	A list of trouble ticket references (TroubleTicketRef [*]). A list of trouble tickets associated with this problem.
category	A string. Classifier for the problem. Settable. For example, this is used for distinguishing the category of problem originator in [role].[category] format. Example: serviceProvider.declarer, supplier.originated, system.originated.
comment	A list of notes (Note [*]). A list of comments or notes made on the problem.
correlationId	A string. Additional identifier coming from an external system.
description	A string. Free form text describing the Service Problem.
extensionInfo	A list of characteristics (Characteristic [*]). A generic list of any type of elements. Used for vendor Extensions or loose element encapsulation from other namespaces.
firstAlert	A related entity reference (RelatedEntityRef). Indicates what first alerted the system to the problem. It is not the root cause of the Service Problem. Examples: Threshold crossing alert.
href	A string. Reference to the Service Problem.
id	A string. Identifier of the service problem.
impactImportanceFactor	A string. Impact Importance is characterized by an Impact Importance Factor: overall importance of the impact of all the affected services, e.g., 0 (zero impact) to 100 (worst impact). The Impact Importance is a calculated field which is set by the operations support system (OSS) determining the impact.
impactPatterns	An impact pattern (ImpactPattern). Define the patterns of impact (optional) – e.g., other service characteristics – Used when defining impact through another pattern than the predefined attributes.

originatingSystem	A string. Indicates where the problem was generated.
originatorParty	A related party (RelatedParty). Individual or organization that created the problem.
parentProblem	A list of service problem references (ServiceProblemRef [*]). The parent problem to which this problem is attached.
priority	An integer. An indication varying from 1 (highest) to 10 (lowest) of how important it is for the service provider to correct the Service Problem.
problemEscalation	A string. Indicates if this service problem has been escalated or not. Possible values are 0 to 10. A value of zero means no escalation. The meanings of values 1-10 are to be determined by the user of the interface, but they show increasing levels of escalation.
reason	A string. Free text or optionally structured text. It can be Unknown.
relatedEvent	A list of event references (EventRef [*]). List of events associated to this problem.
relatedObject	A list of related entity references (RelatedEntityRef [*]). List of objects associated to this problem.
relatedParty	A list of related parties (RelatedParty [*]). List of parties or party roles playing a role within the service problem.
resolutionDate	A date time (DateTime). Time the problem was resolved.
responsibleParty	A related party (RelatedParty). Individual or organization responsible for handling this problem.
rootCauseResource	A list of resource references (ResourceRef [*]). Resource(s) that are associated to the underlying service problems that are the Root Cause of this one if any (used only if applicable).
rootCauseService	A list of service references (ServiceRef [*]). Service(s) that are associated to the underlying service problems that are the Root Cause of this one if any (used only if applicable).
status	A string. The current status of the service problem. Possible values are Submitted, Rejected, Acknowledged, In Progress [Held, Pending], Resolved, Closed, and Cancelled.
statusChangeDate	A date time (DateTime). Time the problem was last status changed.
statusChangeReason	A string. The reason of state change.
timeChanged	A date time (DateTime). Time the problem was last changed.
timeRaised	A date time (DateTime). Time the problem was raised.
trackingRecord	A list of tracking records (TrackingRecord [*]). List of tracking records that allow the tracking of modifications on the problem. The tracking records should not be embedded in the problem to allow retrieving the problem without the tracking records.

underlyingAlarm	A list of resource alarm references (ResourceAlarmRef [*]). A list of alarms underlying this problem.
underlyingProblem	A list of service problem references (ServiceProblemRef [*]). A list of underlying problems. Relevant only if this problem is derived from other problems.

Characteristic sub-resource

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

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.

ImpactPattern sub-resource

Define the patterns of impact (optional), such as other service characteristics- Used when defining impact through another pattern than the pre-defined attributes.

description	A string. Basic description of the impact pattern.
extensionInfo	A list of characteristics (Characteristic [*]). A generic list of any type of elements. Used for extensions or loose element encapsulation from other namespaces.

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.

Place sub-resource

Place reference. Place defines the places where the products are sold or delivered.

href	A string. Unique reference of the place.
id	A string. Unique identifier of the place.
name	A string. A user-friendly name for the place, such as [Paris Store], [London Store], [Main Home].

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.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.
@referredType	A string. The actual type of the target instance when needed for disambiguation.

role A string. Role played by the related party.

TrackingRecord sub-resource

Tracking records allow the tracking of modifications on the problem. The tracking records should not be embedded in the problem to allow retrieving the problem without the tracking records.

description A string. Describes the action being done, such as: ack, clear.

extensionInfo A list of characteristics (Characteristic [*]). A generic list of any type of elements. Used for vendor Extensions or loose element encapsulation from other namespaces.

id A string. Identifier of the TrackingRecord.

systemId A string. Describes the system Id from which the action was done.

time A date time (DateTime). Describes the time at which the action was done.

user A string. Describes the user doing the action.

ChangeRequestRef relationship

Reference to a Change Request.

@referredType A string. The actual type of the target instance when needed for disambiguation.

href A string. The reference link to the change request.

id A string. The identifier of the change request.

EventRef relationship

Events linked with service problem.

@referredType A string. The actual type of the target instance when needed for disambiguation.

eventTime A date time (DateTime). Time the event occurred.

eventType A string. Type of the event.

href A string. event reference.

id A string. ID of the event.

RelatedEntityRef relationship

A reference to an entity, where the type of the entity is not known in advance.

@referredType A string. The actual type of the target instance when needed for disambiguation.

href A string. The hyperlink to access an entity.

id A string. The identifier of an entity.

name A string. The name of an entity.

ResourceAlarmRef relationship

A set of alarm ids identifying the alarms that are underlying this problem.

@referredType A string. The actual type of the target instance when needed for disambiguation.

changeRequest	A change request reference (ChangeRequestRef). Reference to a Change Request.
href	A string. Reference of the alarm.
id	A string. Unique identifier of the alarm.

ResourceRef relationship

Information about a resource that holds realizes the product that is linked to the bucket balance.

@Type	A string. Indicates the type of resource.
href	A string. Reference to the party.
id	A string. Unique identifier of the related party.
name	A string. Name of the resource.
value	A string. The resource value that can be used to identify a resource with a public key (e.g.,: a tel nr, an msisdn).

SLAViolationRef relationship

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string.
id	A string.

ServiceProblemRef relationship

@referredType	A string. The actual type of the target instance when needed for disambiguation.
correlationId	A string. Additional identifier coming from an external system.
href	A string. Reference of the Problem.
id	A string. Unique identifier of the Problem.

ServiceRef relationship

Service reference, for when Service is used by other entities.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.

TroubleTicketRef relationship

TroubleTicket reference, for when a Trouble Ticket is used by other entities.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.

Json representation sample

The following is an example json representation of a 'ServiceProblem' resource object.


```
{
  "affectedLocation": [
    {}
  ],
  "affectedNumberOfServices": 69,
  "affectedResource": [
    {}
  ],
  "affectedService": [
    {}
  ],
  "associatedSLAViolation": [
    {}
  ],
  "associatedTroubleTicket": [
    {}
  ],
  "category": "a string ...",
  "comment": [
    {}
  ],
  "correlationId": "305",
  "description": "This  service problem ...",
  "extensionInfo": [
    {}
  ],
  "firstAlert": {},
  "href": "https:/host:port/tmf-api/serviceProblem/v1/serviceProblem/8561",
  "id": "8561",
  "impactImportanceFactor": "a string ...",
  "impactPatterns": {},
  "originatingSystem": "a string ...",
  "originatorParty": {},
  "parentProblem": [
    {}
  ],
  "priority": 51,
  "problemEscalation": "a string ...",
  "reason": "a string ...",
  "relatedEvent": [
    {}
  ],
  "relatedObject": [
    {}
  ],
  "relatedParty": [
    {}
  ],
  "resolutionDate": "2019-05-13T00:00",
  "responsibleParty": {},
  "rootCauseResource": [
    {}
  ],
  "rootCauseService": [
    {}
```

```
    ],
    "status": "a string ...",
    "statusChangeDate": "2019-05-13T00:00",
    "statusChangeReason": "a string ...",
    "timeChanged": "2019-05-13T00:00",
    "timeRaised": "2019-05-13T00:00",
    "trackingRecord": [
      {}
    ],
    "underlyingAlarm": [
      {}
    ],
    "underlyingProblem": [
      {}
    ]
  }
```

6.1.7 Iot Service Specification resource

Resource model

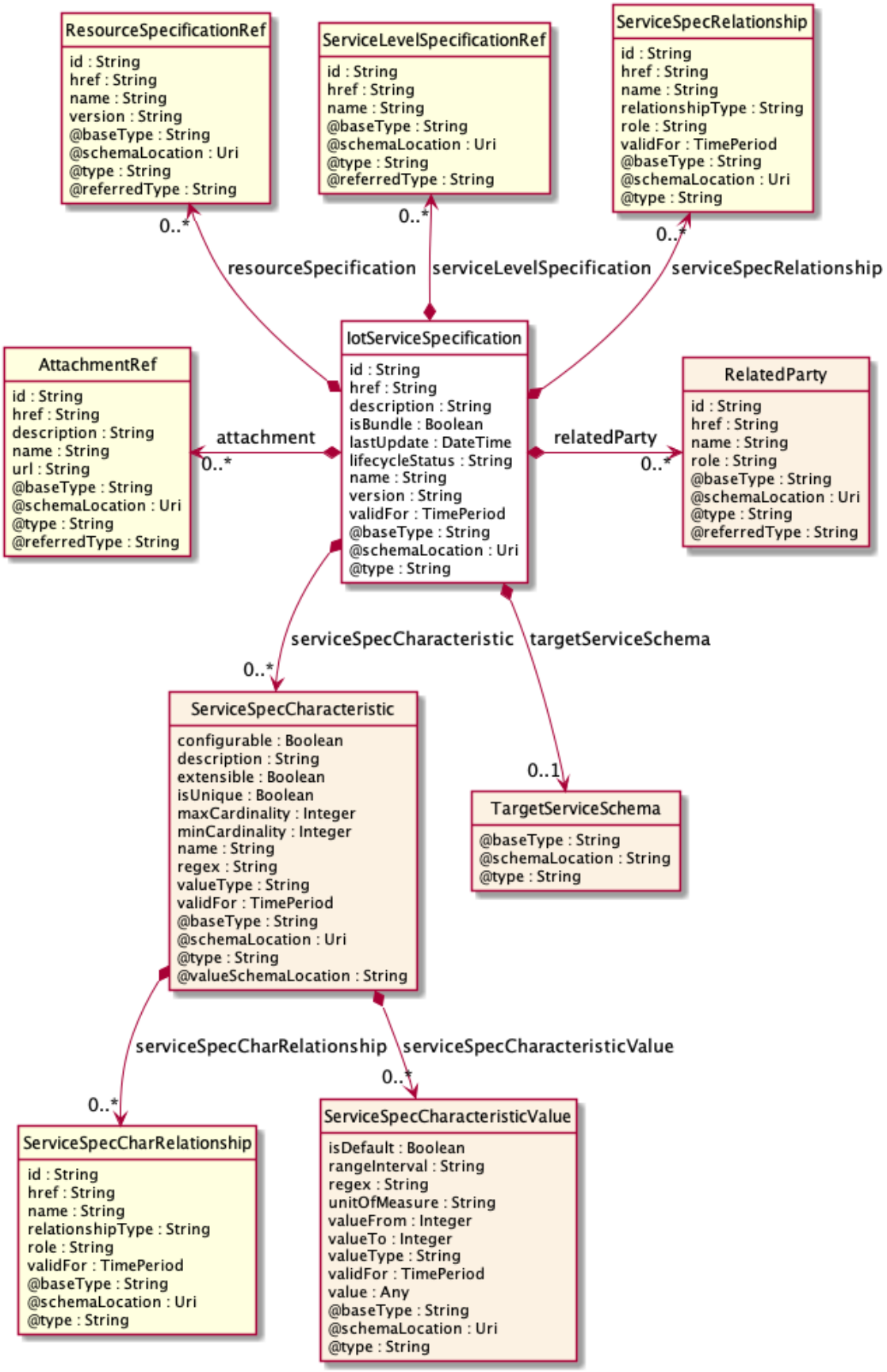

Figure 7 – IoT Service Specification Resource Model, Schematic

Field descriptions

IotServiceSpecification fields

description	A string. A narrative that explains in detail what the service specification is.
href	A string. Reference of the service specification.
id	A string. Unique identifier of the service specification.
isBundle	A Boolean. isBundle determines whether a ServiceSpecification represents a single ServiceSpecification (false), or a bundle of ServiceSpecification (true).
lastUpdate	A date time (DateTime). Date and time of the last update of the service specification.
lifecycleStatus	A string. Used to indicate the current lifecycle status of the service specification.
name	A string. Name of the service specification.
version	A string. Service specification version.
attachment	A list of attachment references (AttachmentRef [*]). A list of attachments (Attachment [*]). Complements the description of the specification through video, pictures...
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.
resourceSpecification	A list of resource specification references (ResourceSpecificationRef [*]). A list of resource specification references (ResourceSpecificationRef [*]). The ResourceSpecification is required for a service specification with type ResourceFacingServiceSpecification (RFSS).
serviceLevelSpecification	A list of service level specification references (ServiceLevelSpecificationRef [*]). A list of service level specifications related to this service specification, and which will need to be satisfiable for corresponding service instances; e.g., Gold, Platinum.
serviceSpecCharacteristic	A list of service spec characteristics (ServiceSpecCharacteristic [*]). A list of service spec characteristics (ServiceSpecCharacteristic [*]). This class represents the key features of this service specification.
serviceSpecRelationship	A list of service spec relationships (ServiceSpecRelationship [*]). A list of service specifications related to this specification, e.g., migration, substitution, dependency or exclusivity relationship.
targetServiceSchema	A target service schema (TargetServiceSchema). A target service schema reference (TargetServiceSchemaRef). The reference object to the schema and type of target service which is described by service specification.
validFor	A period of time. The period for which the service specification is valid.

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.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.
@referredType	A string. The actual type of the target instance when needed for disambiguation.
role	A string. Role played by the related party.

ServiceSpecCharRelationship sub-resource

An aggregation, migration, substitution, dependency or exclusivity relationship between/among serviceSpecCharacteristics.

href	A string. Hyperlink reference to the target specification.
id	A string. Unique identifier of the target specification.
name	A string. Name of the target characteristic.
relationshipType	A string. Type of relationship such as aggregation, migration, substitution, dependency, exclusivity.
role	A string. The association role for this service specification.
validFor	A time period. The period for which the serviceSpecCharRelationship is valid.

ServiceSpecCharacteristic sub-resource

This class represents the key features of this service specification. For example, bandwidth is a characteristic of many different types of services; if bandwidth is a relevant characteristic (e.g., from the point-of-view of a Customer obtaining this Service via a Product) then bandwidth would be a ServiceSpecCharacteristic for that particular Service.

@valueSchemaLocation	A string. This (optional) field provides a link to the schema describing the value type.
configurable	A Boolean. If true, the Boolean indicates that the serviceSpecCharacteristic is configurable.
description	A string. A narrative that explains in detail what the serviceSpecCharacteristic is.
extensible	A Boolean. An indicator that specifies that the values for the characteristic can be extended by adding new values when instantiating a characteristic for an Entity.
isUnique	A Boolean. An indicator that specifies if a value is unique for the specification. Possible values are: "unique while value is in effect" and "unique whether value is in effect or not".

maxCardinality	An integer. The maximum number of instances a CharacteristicValue can take on. For example, zero to five phone numbers in a group calling plan, where five is the value for the maxCardinality.
minCardinality	An integer. The minimum number of instances a CharacteristicValue can take on. For example, zero to five phone numbers in a group calling plan, where zero is the value for the minCardinality.
name	A string. A word, term, or phrase by which this characteristic specification is known and distinguished from other characteristic specifications.
regex	A string. A rule or principle represented in regular expression used to derive the value of a characteristic value.
serviceSpecCharRelationship	A list of service spec char relationships (ServiceSpecCharRelationship [*]). A list of service spec char relationships (ServiceSpecCharRelationship [*]). An aggregation, migration, substitution, dependency or exclusivity relationship between/among Specification Characteristics.
serviceSpecCharacteristicValue	A list of service spec characteristic values (ServiceSpecCharacteristicValue [*]). A list of service spec characteristic values (ServiceSpecCharacteristicValue [*]). A ServiceSpecCharacteristicValue object is used to define a set of attributes, each of which can be assigned to a corresponding set of attributes in a ServiceSpecCharacteristic object. The values of the attributes in the ServiceSpecCharacteristicValue object describe the values of the attributes that a corresponding ServiceSpecCharacteristic object can take on.
validFor	A time period. The period for which the serviceSpecCharacteristic is valid.
valueType	A string. A kind of value that the characteristic can take on, such as numeric, text and so forth.

ServiceSpecCharacteristicValue sub-resource

A ServiceSpecCharacteristicValue object is used to define a set of attributes, each of which can be assigned to a corresponding set of attributes in a ServiceSpecCharacteristic object. The values of the attributes in the ServiceSpecCharacteristicValue object describe the values of the attributes that a corresponding ServiceSpecCharacteristic object can take on.

isDefault	A Boolean. Indicates if the value is the default value for a characteristic.
rangeInterval	A string. An indicator that specifies the inclusion or exclusion of the valueFrom and valueTo attributes. If applicable, possible values are "open", "closed", "closedBottom" and "closedTop".
regex	A string. A regular expression constraint for given value.

unitOfMeasure	A string. A length, surface, volume, dry measure, liquid measure, money, weight, time, and the like. In general, a determinate quantity or magnitude of the kind designated, taken as a standard of comparison for others of the same kind, in assigning to them numerical values, as 1 foot, 1 yard, 1 mile, 1 square foot.
validFor	A time period. The period of time for which a value is applicable.
value	An any (Any). A discrete value that the characteristic can take on, or the actual value of the characteristic.
valueFrom	An integer. The low range value that a characteristic can take on.
valueTo	An integer. The upper range value that a characteristic can take on.
valueType	A string. A kind of value that the characteristic can take on, such as numeric, text, and so forth.

ServiceSpecRelationship sub-resource

A migration, substitution, dependency or exclusivity relationship between/among service specifications.

href	A string. Reference of the target serviceSpecification.
id	A string. Unique identifier of the target serviceSpecification.
name	A string. The name given to the target service specification instance.
relationshipType	A string. Type of relationship such as migration, substitution, dependency, exclusivity.
role	A string. The association role for this service specification.
validFor	A period of time. The period for which the serviceSpecRelationship is valid.

TargetServiceSchema sub-resource

The reference object to the schema and type of target service which is described by service specification.

@schemaLocation	A string. This field provides a link to the schema describing the target service.
@type	A string. Class type of the target service.

AttachmentRef relationship

Attachment reference. An attachment complements the description of an element (for instance a product) through video, pictures.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. URL serving as reference for the attachment resource.
id	A string. Unique-Identifier for this attachment.
name	A string. Name of the related entity.
description	A string. A narrative text describing the content of the attachment.
href	A string. URL serving as reference for the attachment resource.
url	A string. Link to the attachment media/content.

ResourceSpecificationRef relationship

Resource Specification reference: The ResourceSpecification is required to realize a ProductSpecification.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Reference of the resource specification.
id	A string. Unique identifier of the resource specification.
name	A string. Name of the requiredResourceSpecification.
version	A string. Resource specification version.

ServiceLevelSpecificationRef relationship

A Service Level Specification represents a pre-defined or negotiated set of Service Level Objectives. In addition, certain consequences are associated with not meeting the Service Level Objectives. Service Level Agreements are expressed in terms of Service Level Specifications.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. The hyperlink to access a service level specification.
id	A string. The identifier to a service level specification.
name	A string. The name of Service Level Specification.

Json representation sample

The following is an example json representation of an 'IotServiceSpecification' resource object.


```
{
   "description": "This  IoT service specification describes the access to data generated by IoT sensors.",
   "href": "https://host:port/tmf-api/iotServiceSpecification/v1/iotServiceSpecification/4530",
   "id": "4530",
   "isBundle": true,
   "lastUpdate": "2019-05-13T00:00",
   "lifecycleStatus": "InService",
   "name": "IoT Data Service Specification",
   "version": "1.0.0",
   "attachment": [
      {}
   ],
   "relatedParty": [
      {
"@referredType": "temperatureSensor",
"href": "https://www.mandint.org/temperatureSensor",
"id": "CoapTempSensor",
"name": "Mandat International",
"role": "vendor"
}
   ],
   "resourceSpecification": [
      {
"@referredType": "IoTDataService",
"href": "https://www.mandint.org/serviceSpecification/IOTDataServiceSpecification",
```

```
"id": "501",
"name": "serviceSpec501",
"version": "1.0.0"
}
  ],
  "serviceLevelSpecification": [
    {
"@referredType": "IoTDataService",
"href": "https://www.mandint.org/serviceLevelSpecification/IoTDataServiceLevelSpecification",
"id": "601",
"name": "serviceLevelSpec602"
}
  ],
  "serviceSpecCharacteristic": [
    {
"isDefault": true,
"rangeInterval": "open",
"regex": "",
"unitOfMeasure": "degrees Celsius",
"validFor": {
"startPeriod": "2019-08-21T11:45:00+02:00",
"endPeriod": "2019-12-21T11:45:00+2:00"
},
"value": 20.0,
"valueFrom": -273.15,
"valueTo": 10000,
"valueType": "numeric"
}
  ],
  "serviceSpecRelationship": [
    {
"href": "https://www.mandint.org/serviceSpecification/IOTDataServiceSpecification",
"id": "501",
"name": "serviceSpec501",
"relationshipType": "substitution",
"role": "vendor",
"validFor": {
"startPeriod": "2019-08-21T11:45:00+02:00",
"endPeriod": "2019-12-21T11:45:00+2:00"
}
}
  ],
  "targetServiceSchema": {
```

```
"@schemaLocation": "
https://www.mandint.org/serviceSpecification/IOTDataServiceSpecificationSchema.json",

"@type": "IoTDataService"

},
   "validFor": {

"startPeriod": "2019-08-21T11:45:00+02:00",

"endPeriod": "2019-12-21T11:45:00+2:00"

}
}
```

6.1.8 Service Test resource

A service test is an entity that exists for a controlled test invocation on a service. The service test is executed according to a schedule and contains service test configuration parameters that are to be applied at execution time, and service test measures that result.

Resource model

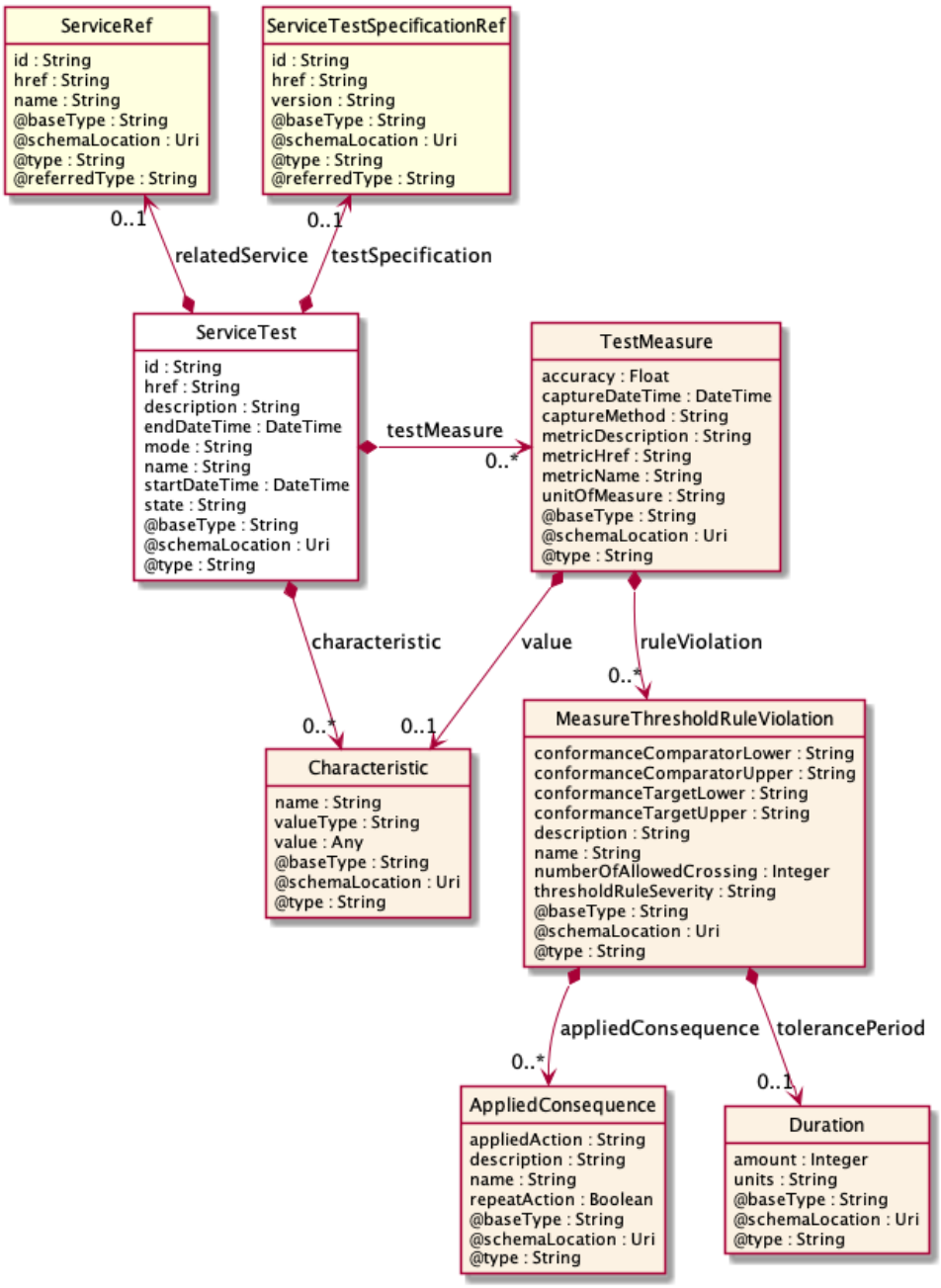

Figure 8 – Service Test Resource Model, Schematic

Field descriptions

ServiceTest fields

characteristic	A list of characteristics (Characteristic [*]). List of characteristics with values that define the test run.
description	A string. Description of the service test.
endDateTime	A date time (DateTime). The end date and time of the service test.
href	A string. Hyperlink to access the service test.
id	A string. The identifier of the service test.

mode	A string. An indication of whether the service test is running in "PROACTIVE" or "ONDEMAND" mode.
name	A string. The name of the service test.
relatedService	A service reference (ServiceRef). The actual service being tested.
startDateTime	A date time (DateTime). The start date and time of the service test.
state	A string. The actual state the service test is in.
testMeasure	A list of test measures (TestMeasure [*]). The results of the test in terms of the measured metrics.
testSpecification	A service test specification reference (ServiceTestSpecificationRef). The specification for this test.

AppliedConsequence sub-resource

An Applied Consequence defines the action (prescribed action or notification) to take when a MeasureThresholdRuleViolation occurs.

appliedAction	A string. The action for a violated threshold. This could be a hyperlink to the action.
description	A string. A narrative that explains in detail what the consequence is.
name	A string. A word, term, or phrase by which Consequence is known and distinguished from other MetricDefMeasureConsequences.
repeatAction	A Boolean. An indicator used to specify that a consequence should cease being applied if a value is in the same range as the previous value or continue being applied if a value is in the same range as the previous value. If the repeatAction is True, if the consequence is always applied as soon as the MetricMeasure value is in the range of values and if the repeatAction is False, the consequence is applied only if the previous MetricMeasure value was not in the same range.

Characteristic sub-resource

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

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.

Duration sub-resource

A time interval in a given unit of time.

amount	An integer. Time interval (number of seconds, minutes, hours, etc.).
units	A string. Unit of time (seconds, minutes, hours, etc.).

MeasureThresholdRuleViolation sub-resource

A measureThresholdRuleViolation is a violation of a rule that defines the in the MericDefMeasureThresholdRule.

appliedConsequence	A list of applied consequences (AppliedConsequence [*]). An Applied Consequence defines the action (prescribed action or notification) to take when a MeasureThresholdRuleViolation occurs.
conformanceComparatorLower	A string. An operator that when applied on a value specifies whether a threshold is crossed or ceased to be crossed. This operator is used to compare with the conformanceTargetLower.
conformanceComparatorUpper	A string. An operator that when applied on a value specifies whether a threshold is crossed or ceased to be crossed. This operator is used to compare with the conformanceTargetUpper.
conformanceTargetLower	A string. A value used to determine if the threshold is crossed or ceases to be crossed. It represents the lower limit. The value should be less than the conformanceTargetUpper. The conformance comparators should also be logically defined so as to not lead to a logically impossible condition.
conformanceTargetUpper	A string. A value used to determine if the threshold is crossed or ceases to be crossed. It represents the Upper limit. The value should be greater than the conformanceTargetLower. The conformance comparators should also be logically defined so as to not lead to a logically impossible condition.
description	A string. Description for the MetricDefMeasure ThresholdRule.
name	A string. Name for the MetricDefMeasure ThresholdRule.
numberOfAllowedCrossing	An integer. The number of allowed crossing occurrences in reference to the tolerancePeriod without a consequence being initiated.
thresholdRuleSeverity	A string. A threshold can be generated in different severity levels. A crossing for each level may require a different condition and possibly trigger a different consequence.
tolerancePeriod	A duration (Duration). An interval of time of allowed crossing occurrences before a consequence being initiated.

TestMeasure sub-resource

A TestMeasure specifies a measure of a specific aspect of a product, service, or resource test, such as lost packets or connectivity status.

accuracy	A float. The number of digits of accuracy captured for associated Metrics.
captureDateTime	A date time (DateTime). The date and time that the metric was captured.

captureMethod	A string. The method used to capture the Metrics (This may be replaced by a set of entities similar to the Performance Monitoring Ref).
metricDescription	A string. Brief description of the metric.
metricHref	A string. Hyperlink to access a metric for detail information.
metricName	A string. The name of the metric.
ruleViolation	A list of measure threshold rule violations (MeasureThreshold RuleViolation [*]). A list of rules that were violated in this test measure.
unitOfMeasure	A string. The unit of measure for the metric values, such as meters, cubic yards, kilograms [ISO 1000].
value	A characteristic (Characteristic). The value of Metric in the test. This also could be a document to record all values captured during the service test.

ServiceRef relationship

Service reference, for when Service is used by other entities.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.

ServiceTestSpecificationRef relationship

The service test specification used by the service test.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
href	A string. Hyperlink to access a service test specification.
id	A string. Identifier of a service test specification.
version	A string. Version of a service test specification.

Json representation sample

The following is an example json representation of a 'ServiceTest' resource object.


```
{
  "characteristic": [
    {}
  ],
  "description": "This  service test ...",
  "endDateTime": "2019-05-13T00:00",
  "href": "https:/host:port/tmf-api/serviceTest/v1/serviceTest/1983",
  "id": "1983",
  "mode": "a string ...",
  "name": "a string ...",
  "relatedService": {},
  "startDateTime": "2019-05-13T00:00",
  "state": "a string ...",
  "testMeasure": [
    {}
```

```
    ],
    "testSpecification": {}
}
```

6.1.9 Service Test Specification resource

The service test specification describes the service test in terms of parameters to be configured and measures to be taken.

Resource model

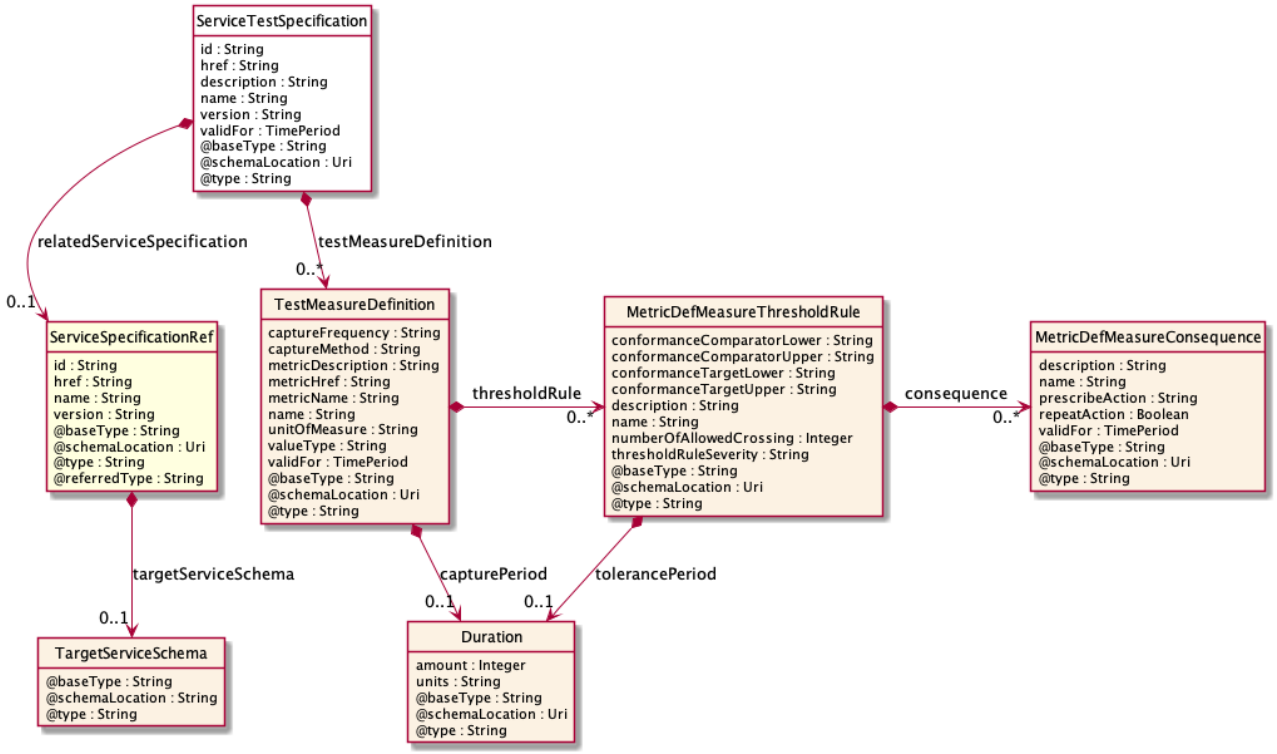

Figure 9 – Service Test Specification Resource Model, Schematic

Field descriptions

ServiceTestSpecification fields

description	A string. Description of a service test specification.
href	A string. Hyperlink to access a service test specification.
id	A string. Identifier of a service test specification.
name	A string. Name of a service test specification.
relatedServiceSpecification	A service specification reference (ServiceSpecificationRef). The related service specification.
testMeasureDefinition	A list of test measure definitions (TestMeasureDefinition [*]). A list of definitions for the measurements for the test defined by this specification.
validFor	A period of time. The period of time for which this specification is valid.
version	A string. Version of a service test specification.

Duration sub-resource

A time interval in a given unit of time.

amount	An integer. Time interval (number of seconds, minutes, hours, etc.).
units	A string. Unit of time (seconds, minutes, hours, etc.).

MetricDefMeasureConsequence sub-resource

A MetricDefMeasureConsequence defines the action (prescribed action or notification) to take when a MetricDefMeasureThresholdRule is crossed.

description	A string. A narrative that explains in detail what the consequence is.
name	A string. A word, term, or phrase by which a MetricDefMeasure Consequence is known and distinguished from other MetricDefMeasureConsequences.
prescribeAction	A string. Recommended remedy for a violated threshold. This could be the hyperlink to the action.
repeatAction	A Boolean. An indicator used to specify that a consequence should cease being applied if a value is in the same range as the previous value or continue being applied if a value is in the same range as the previous value. If the repeatAction is True, if the consequence is always applied as soon as the MetricMeasure value is in the range of values and if the repeatAction is False, the consequence is applied only if the previous MetricMeasure value was not in the same range.
validFor	A period of time. A valid duration of a thing.

MetricDefMeasureThresholdRule sub-resource

A MetricDefMeasureThresholdRule is a rule that defines the condition (raise or clear) to achieve to apply consequences when a threshold is crossed or ceased to be crossed. It also defines the severity of the raise or clear of the threshold.

conformanceComparatorLower	A string. An operator that when applied on a value specifies whether a threshold is crossed or ceased to be crossed. This operator is used to Service Test Management API REST Specification compare with the conformanceTargetLower.
conformanceComparatorUpper	A string. An operator that when applied on a value specifies whether a threshold is crossed or ceased to be crossed. This operator is used to compare with the conformanceTargetUpper.
conformanceTargetLower	A string. A value used to determine if the threshold is crossed or ceases to be crossed. It represents the lower limit. The value should be less than the conformance TargetUpper. The conformance comparators should also be logically defined so as to not lead to a logically impossible condition.

conformanceTargetUpper	A string. A value used to determine if the threshold is crossed or ceases to be crossed. It represents the Upper limit. The value should be greater than the conformanceTargetLower. The conformance comparators should also be logically defined so as to not lead to a logically impossible condition.
consequence	A list of metric def measure consequences (MetricDefMeasureConsequence [*]). A list of consequences (actions, notifications) that will arise if the threshold is crossed.
description	A string. Description for the MetricDefMeasure ThresholdRule.
name	A string. Name for the MetricDefMeasure ThresholdRule.
numberOfAllowedCrossing	An integer. The number of allowed crossing occurrences in reference to the tolerancePeriod without a consequence being initiated.
thresholdRuleSeverity	A string. A threshold can be generated in different severity levels. A crossing for each level may require a different condition and possibly trigger a different consequence.
tolerancePeriod	A duration (Duration). An interval of time of allowed crossing occurrences before a consequence being initiated.

TargetServiceSchema sub-resource

The reference object to the schema and type of target service which is described by service specification.

@schemaLocation	A string. This field provides a link to the schema describing the target service.
@type	A string. Class type of the target service.

TestMeasureDefinition sub-resource

A TestMeasureDefinition specifies a measure of a specific aspect of a product, service, or resource test, such as lost packets or connectivity status.

captureFrequency	A string. The frequency of capture for the metric. Note: This may be replaced by a set of entities similar to the Performance Monitoring.
captureMethod	A string. The method used to capture the Metric. Note: This may be replaced by a set of entities similar to the Performance Monitoring.
capturePeriod	A duration (Duration). A period of the capture.
metricDescription	A string. Brief description of the metric.
metricHref	A string. Hyperlink to access a metric for detail information.
metricName	A string. The name of a metric that in the test measure.
name	A string. The name of the TestMeasureDefinition.

thresholdRule	A list of metrics def measure threshold rules (MetricDefMeasure ThresholdRule [*]). The rule(s) associated with the measure threshold.
unitOfMeasure	A string. Name of a service test specification.
validFor	A time period. The time period for which this definition is valid.
valueType	A string. A kind of value that the Metric value can take on, such as numeric, text, and so forth.

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.
href	A string. Reference of the serviceSpecification.
id	A string. Unique identifier of the service specification.
name	A string. Name of the requiredServiceSpecification.
targetServiceSchema	A target service schema (TargetServiceSchema). A target service schema reference (TargetServiceSchemaRef). The reference object to the schema and type of target service which is described by service specification.
version	A string. Service specification version.

Json representation sample

The following is an example json representation of a 'ServiceTestSpecification' resource object.

```
{
  "description": "This service test specification ...",
  "href": "https:/host:port/tmf-api/serviceTestSpecification/v1/serviceTestSpecification/8023",
  "id": "8023",
  "name": "a string ...",
  "relatedServiceSpecification": {},
  "testMeasureDefinition": [
    {}
  ],
  "validFor": {},
  "version": "a string ..."
}
```

6.1.10 Usage Consumption Report Request resource

An UsageConsumptionReportRequest allows to manage the calculation request of a usage consumption report.

Resource model

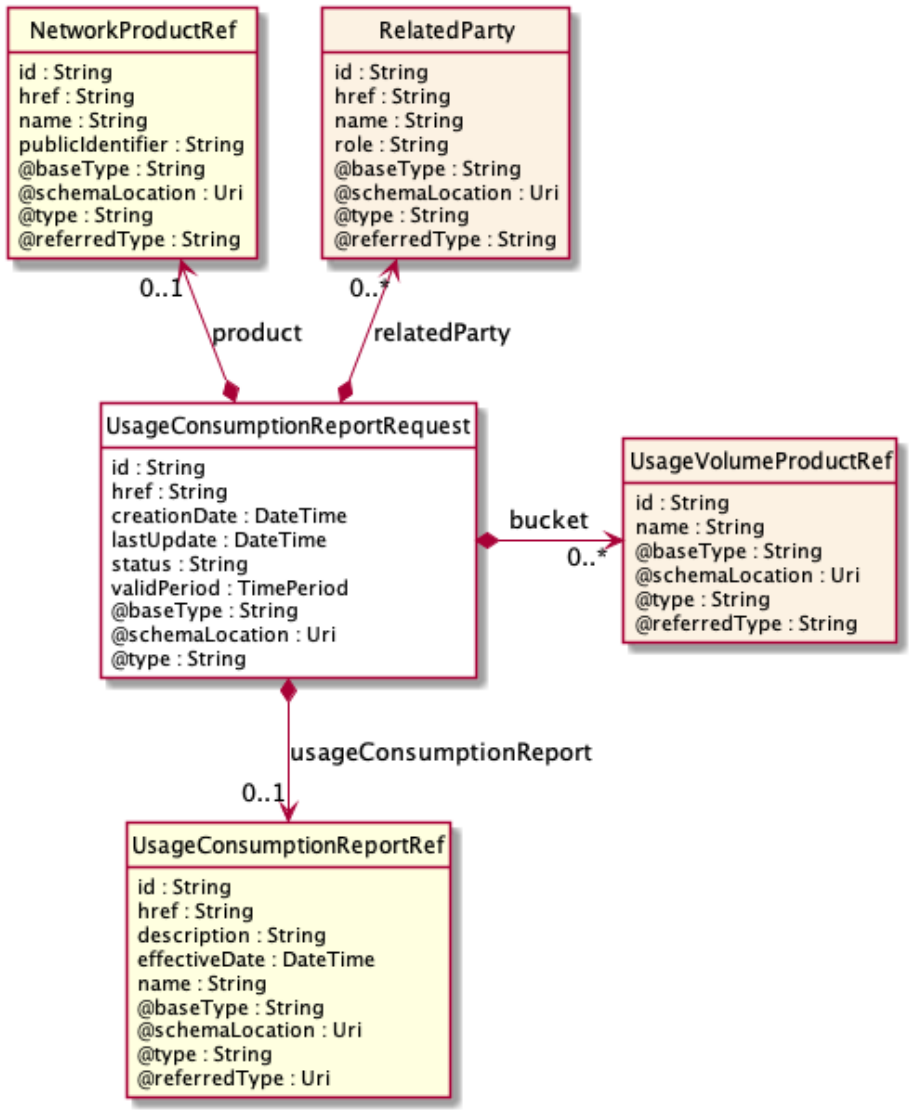

Figure 10 – Usage Consumption Report Request Resource Model, Schematic

Field descriptions

UsageConsumptionReportRequest fields

bucket	A list of usage volume product references (UsageVolumeProductRef [*]). Reference of the buckets for which the usage consumption report is requested.
creationDate	A date time (DateTime). Date and time of the request creation.
href	A string. Reference of the usage consumption report request.
id	A string. Unique identifier of the usage consumption report request given by the server.
lastUpdate	A date time (DateTime). Date when the status was last changed.
product	A network product reference (NetworkProductRef). Network product reference for which the usage consumption report is requested.

relatedParty	A list of related parties (RelatedParty [*]). Reference and role of the related parties for which the usage consumption report is requested.
status	A string. Status of the usage consumption report request (InProgress or done).
usageConsumptionReport	A usage consumption report reference (UsageConsumption ReportRef). References of the usage consumption report (given when it has been calculated and the status of the request is done).
validPeriod	A period of time. Validity period.

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.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.
@referredType	A string. The actual type of the target instance when needed for disambiguation.
role	A string. Role played by the related party.

NetworkProductRef relationship

Reference of a product.

@referredType	A string. Generic attribute indicating the name of the class type of the referred resource entity.
href	A string. Reference to the network product.
id	A string. Unique identifier of the network product.
name	A string. Network product name.
publicIdentifier	A string. Public number associated to the product (msisdn number for mobile line for example).

UsageConsumptionReportRef relationship

Reference of a usage consumption report.

description	A string. Free short text describing the usage consumption report content.
effectiveDate	A date time (DateTime). Date and time when the usage consumption report was calculated and generated.
href	A string. Hyperlink to access the usage consumption report.
id	A string. Unique identifier of the usage consumption report given by the server.
name	A string. Usage consumption report name.

UsageVolumeProductRef relationship

Reference of a bucket.

@referredType	A string. Generic attribute indicating the name of the class type of the referred resource entity.
id	A string. Unique identifier of the bucket.
name	A string. Bucket name.

Json representation sample

The following is an example json representation of a 'UsageConsumptionReportRequest' resource object.


```
{
   "bucket": [
      {}
   ],
   "creationDate": "2019-05-13T00:00",
   "href": "https:/host:port/tmf-
api/usageConsumptionReportRequest/v1/usageConsumptionReportRequest/4925",
   "id": "4925",
   "lastUpdate": "2019-05-13T00:00",
   "product": {},
   "relatedParty": [
      {}
   ],
   "status": "a string ...",
   "usageConsumptionReport": {},
   "validPeriod": {}
}
```


6.1.11 User resource

Generic User structure used to define commonalities between sub concepts of PartyUser and Financial User.

Resource model

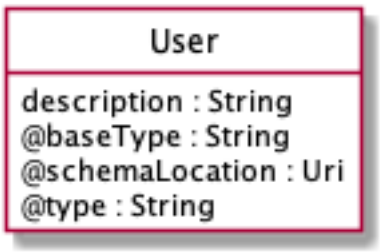

Figure 11 – User Resource Model, Schematic

Field descriptions

User fields

description	A string.

Json representation sample

The following is an example json representation of a 'User' resource object.

```
{
   "description": "This user ..."
}
```

6.1.12 Usage Consumption Report resource

A usage consumption report enables to know at a given point the balances and the consumption counters related to various buckets (SMS, Voice, Data for example). It could be calculated for a device identified by a public key (msisdn number for a mobile device for example or PSTN or VOIP number for a fix device), for a subscribed offer or option or for a user.

Resource model

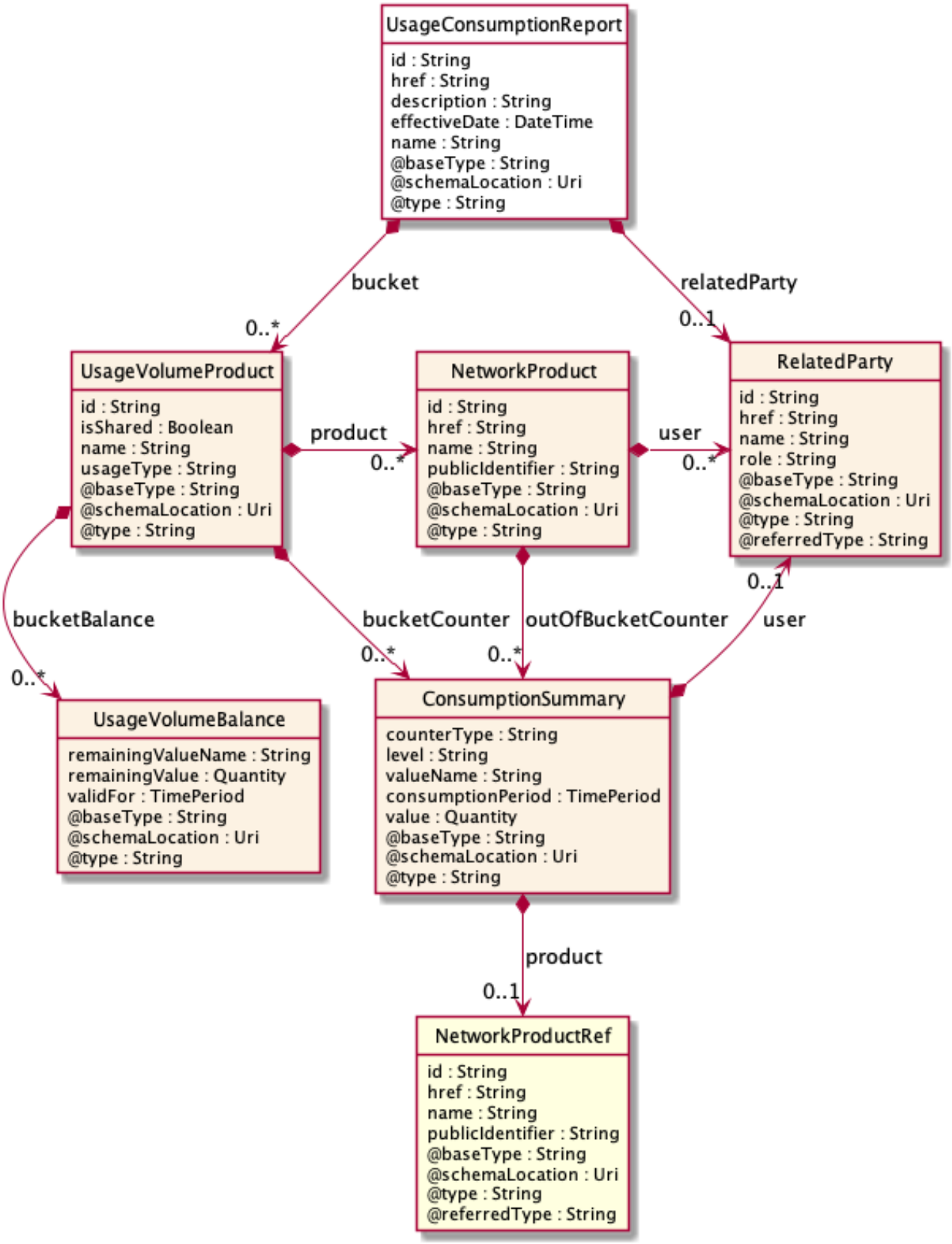

Figure 12 – Usage Consumption Report Resource Model, Schematic

Field descriptions

UsageConsumptionReport fields

bucket	A list of usage volume products (UsageVolumeProduct [*]). Bucket(s) included in the offer or option subscribed.
description	A string. Free short text describing the usage consumption report content.
effectiveDate	A date time (DateTime). Date and time when the usage consumption report was calculated and generated.
href	A string. Hyperlink to access the usage consumption report. The report is calculated at the time of the request by the server. Generally, this report is not recorded by the server. If it is, a unique identifier of the usage consumption report is given by the server.
id	A string. The report is calculated at the time of the request by the server. Generally, this report is not recorded by the server. If it is, a unique identifier of the usage consumption report is given by the server.
name	A string. Usage consumption report name.
relatedParty	A related party (RelatedParty). Related Entity reference. A related party defines party or party role linked to a specific entity.

ConsumptionSummary sub-resource

The consumption counters (called ConsumptionSummary in the SID model) detail for example the different kind of consumption done on the bucket.

consumptionPeriod	A period of time. Consumption counter period between a start date time and an end date time. For prepaid bucket, the period of counters is between the start date of the bucket and the effective date of the usage consumption report generation. For postpaid bucket, the period of counters is between the last bill date and the effective date of the usage consumption report generation.
counterType	A string. Type of the consumption counter. We For example, a counter of the used value for a bucket (counterType=used for example) or the value of the consumption done out of the bucket(s) (counterType=outOfBucket for example).
level	A string. Counter level. The counter can be given globally for the bucket or detailed by user or by network product for example in case of shared bucket.
product	A network product reference (NetworkProductRef). Public identifier associated to the product for which the consumption counter is detailed. If the level of the counter is global (i.e., concerning all the network products), the list of them is not specified. It can be found through the network products listed in the bucket sub-resource.
user	A related party (RelatedParty). Reference of the user for which the consumption counter is detailed. If the level of the counter is global (i.e., concerning all the users), the list of them is not specified. It can be found through the users listed in the network products of the bucket sub-resource.

value	A quantity (Quantity). Numeric value of the bucket counter in a given unit.
valueName	A string. Value of the counter in a formatted string used for display needs for example.

NetworkProduct sub-resource

An instantiated network product (specialization of a product) subscribed by a customer.

href	A string. Reference to the network product.
id	A string. Unique identifier of the network product.
name	A string. Network product name.
outOfBucketCounter	A list of consumption summaries (ConsumptionSummary [*]). Counters detailing usage consumption out of the buckets.
publicIdentifier	A string. Public number associated to the network product (msisdn number for mobile line for example).
user	A list of related parties (RelatedParty [*]). References of the users of the network product.

Quantity sub-resource

An amount in a given unit.

amount	A float. Numeric value in a given unit.
units	A string. Unit.

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.
href	A string. Reference of the related entity.
id	A string. Unique identifier of a related entity.
name	A string. Name of the related entity.
@referredType	A string. The actual type of the target instance when needed for disambiguation.
role	A string. Role played by the related party.

UsageVolumeBalance sub-resource

The balance (called UsageVolumeBalance in the SID model) defines the remaining allowed product usage quantity in terms of volume, time, currency or events. It corresponds to the initial allowed usage quantity minus the usage consumed on the bucket.

remainingValue	A quantity (Quantity). Numeric remaining value for the bucket given in the unit (for example 1.9). This numeric value could be used for calculation for example.
remainingValueName	A string. Remaining value in a formatted string for the bucket given in the balance unit (for example 1.9 Gb). This formatted string could be used for display needs for example.

validFor — A time period. Balance period between a start date time and an end date time. For prepaid bucket, the period of the balance is between the effective date of the usage consumption report generation and the end date of the bucket. For postpaid bucket, the period of the balance is between the effective date of the usage consumption report generation and the next bill date.

UsageVolumeProduct sub-resource

A bucket (called UsageVolumeProduct in the SID model) represents a quantity of usage, as 2 hours national calls or 50 sms for example. It could be either a quantity or an amount in a currency (i.e., It could represent a fixed number of SMS, MMS, minutes of calls, quantity of data, number of events as well as a specific amount in a given currency). It requires one or more network products from which usages will debit the bucket.

bucketBalance — A list of usage volume balances (UsageVolumeBalance [*]). The balance (called UsageVolumeBalance in the SID model) defines the remaining allowed product usage quantity in terms of volume, time, currency or events. It corresponds to the initial allowed usage quantity minus the usage consumed on the bucket.

bucketCounter — A list of consumption summaries (ConsumptionSummary [*]). The consumption counters (called ConsumptionSummary in the SID model) detail for example the different kind of consumption done on the bucket.

id — A string. Unique identifier of the bucket.

isShared — A Boolean. True if the bucket is shared between several devices or users.

name — A string. Bucket name.

product — A list of network products (NetworkProduct [*]). An instantiated network product (specialization of a product) subscribed by a customer.

usageType — A string. Type of usage concerned by the bucket, such as voice, sms, data.

NetworkProductRef relationship

Reference of a product.

@referredType — A string. Generic attribute indicating the name of the class type of the referred resource entity.

href — A string. Reference to the network product.

id — A string. Unique identifier of the network product.

name — A string. Network product name.

publicIdentifier — A string. Public number associated to the product (msisdn number for mobile line for example).

Json representation sample

The following is an example json representation of a 'UsageConsumptionReport' resource object

```
{
  "bucket": [
    {}
  ],
  "description": "This  usage consumption report ...",
```

```
    "effectiveDate": "2019-05-13T00:00",
    "href": "https:/host:port/tmf-api/usageConsumptionReport/v1/usageConsumptionReport/8486",
    "id": "8486",
    "name": "a string ...",
    "relatedParty": {}
}
```


6.1.13 Import Job resource

Represents a task used to import resources from a file.

Resource model

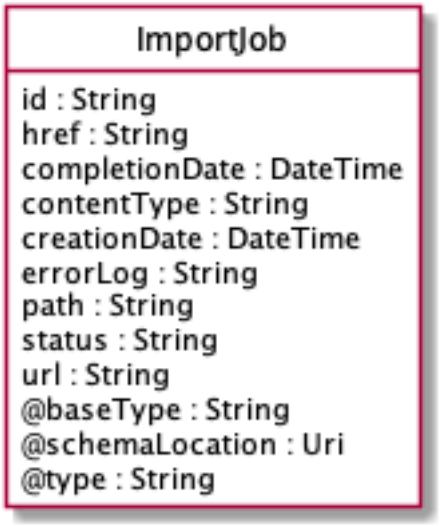

Figure 13 – Import Job Resource Model, Schematic

Field descriptions

ImportJob fields

completionDate	A date time (DateTime). Date at which the job was completed.
contentType	A string. Indicates the format of the imported data.
creationDate	A date time (DateTime). Date at which the job was created.
errorLog	A string. Reason for failure if status is failed.
href	A string. Reference of the import job.
id	A string. Identifier of the import job.
path	A string. URL of the root resource where the content of the file specified by the import job must be applied.
status	A string. Status of the import job (not started, running, succeeded, failed).
url	A string. URL of the file containing the data to be imported.

Json representation sample

The following is an example json representation of an 'ImportJob' resource object.


```
{
    "completionDate": "2019-05-13T00:00",
    "contentType": "a string ...",
```

```
    "creationDate": "2019-05-13T00:00",
    "errorLog": "a string ...",
    "href": "https:/host:port/tmf-api/importJob/v1/importJob/4640",
    "id": "4640",
    "path": "a string ...",
    "status": "a string ...",
    "url": "a string ..."
  }
```


6.1.14 Export Job resource

Represents a task used to export resources to a file.

Resource model

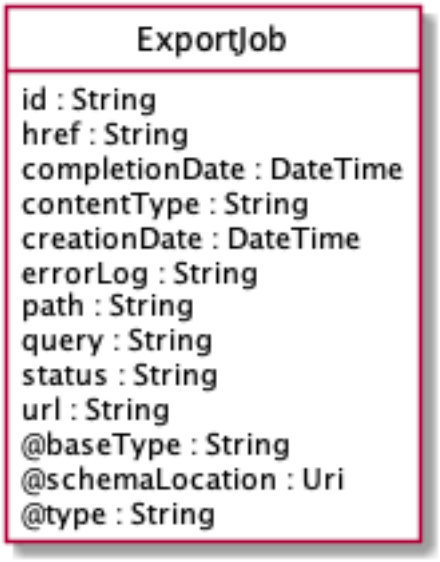

Figure 14 – Export Job Resource Model, Schematic

Field descriptions

ExportJob fields

completionDate	A date time (DateTime). Data at which the job was completed.
contentType	A string. The format of the exported data.
creationDate	A date time (DateTime). Date at which the job was created.
errorLog	A string. Reason for failure.
href	A string. Reference of the export job.
id	A string. Identifier of the export job.
path	A string. URL of the root resource acting as the source for streaming content to the file specified by the export job.
query	A string. Used to scope the exported data.
status	A string. Status of the export job (not started, running, succeeded, failed).
url	A string. URL of the file containing the data to be exported.

Json representation sample

The following is an example json representation of an 'ExportJob' resource object


```
{
  "completionDate": "2019-05-13T00:00",
  "contentType": "a string ...",
  "creationDate": "2019-05-13T00:00",
  "errorLog": "a string ...",
  "href": "https:/host:port/tmf-api/exportJob/v1/exportJob/1721",
  "id": "1721",
  "path": "a string ...",
  "query": "a string ...",
  "status": "a string ...",
  "url": "a string ..."
}
```


6.2 Notification Resource Models

35 notifications are defined for this API.

Notifications related to IotService:
- IotServiceCreateEvent
- IotServiceAttributeValueChangeEvent
- IotServiceStateChangeEvent
- IotServiceBatchEvent
- IotServiceDeleteEvent

Notifications related to ServiceCatalog:
- ServiceCatalogCreateEvent
- ServiceCatalogChangeEvent
- ServiceCatalogBatchEvent
- ServiceCatalogDeleteEvent

Notifications related to ServiceCategory:
- ServiceCategoryCreateEvent
- ServiceCategoryChangeEvent
- ServiceCategoryDeleteEvent

Notifications related to ServiceCandidate:
- ServiceCandidateCreateEvent
- ServiceCandidateChangeEvent
- ServiceCandidateDeleteEvent

Notifications related to ServiceQualification:
- ServiceQualificationCreateEvent
- ServiceQualificationChangeEvent
- ServiceQualificationDeleteEvent

Notifications related to ServiceProblem:
- ServiceProblemCreateEvent
- ServiceProblemStateChangeEvent
- ServiceProblemAttributeValueChangeEvent
- ServiceProblemInformationRequiredEvent

Notifications related to IotServiceSpecification:
- IotServiceSpecificationCreateEvent
- IotServiceSpecificationChangeEvent
- IotServiceSpecificationDeleteEvent

Notifications related to ServiceTest:
- ServiceTestCreateEvent
- ServiceTestAttributeValueChangeEvent
- ServiceTestDeleteEvent

Notifications related to ServiceTestSpecification:
- ServiceTestSpecificationCreateEvent
- ServiceTestSpecificationAttributeValueChangeEvent
- ServiceTestSpecificationDeleteEvent

Notifications related to UsageConsumptionReportRequest:
- UsageConsumptionReportRequestStateChangeEvent

Notifications related to User:
- UserCreateEvent
- UserChangeEvent
- UserDeleteEvent

The notification structure for all notifications in this API follow the pattern depicted by the figure below.

A notification resource (depicted by "SpecificNotification" placeholder) is a sub class of a generic Notification structure containing an id of the event occurrence (eventId), an event timestamp (eventTime), and the name of the notification resource (eventType).

This notification structure owns an event structure ("SpecificEvent" placeholder) linked to the resource concerned by the notification using the resource name as access field ("resourceName" placeholder).

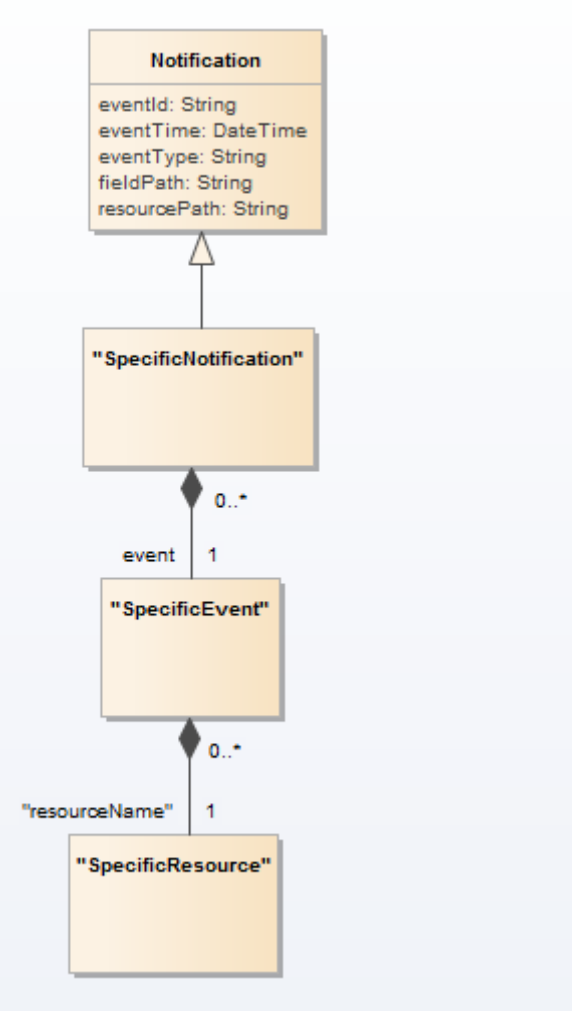

Figure 15 – Notification Resource Model, Schematic

6.2.1 Iot Service Create Event Notification

Notification IotServiceCreateEvent case for resource IotService

Json representation sample

The following is an example json representation of an 'IotServiceIotServiceCreateEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"IotServiceIotServiceCreateEventNotification",
    "event": {
      "iotService" :
         {-- SEE IotService RESOURCE SAMPLE --}
   }
}
```

6.2.2 Iot Service Iot Service Attribute Value Change Event Notification

Notification IotServiceAttributeValueChangeEvent case for resource IotService.

Json representation sample

The following is an example json representation of an 'IotServiceIotServiceAttribute ValueChangeEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"IotServiceIotServiceAttributeValueChangeEventNotification",
    "event": {
      "iotService" :
         {-- SEE IotService RESOURCE SAMPLE --}
   }
}
```

6.2.3 Iot Service Iot Service State Change Event Notification

Notification IotServiceStateChangeEvent case for resource IotService

Json representation sample

The following is an example json representation of an example of an 'IotServiceIotServiceStateChangeEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"IotServiceIotServiceStateChangeEventNotification",
    "event": {
      "iotService" :
         {-- SEE IotService RESOURCE SAMPLE --}
   }
}
```

6.2.4 Iot Service Iot Service Batch Event Notification

Notification IotServiceBatchEvent case for resource IotService

Json representation sample

The following is an example json representation of an 'IotServiceIotServiceBatchEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"IotServiceIotServiceBatchEventNotification",
    "event": {
      "iotService" :
         {-- SEE IotService RESOURCE SAMPLE --}
   }
}
```

6.2.5 Iot Service Iot Service Delete Event Notification

Notification IotServiceDeleteEvent case for resource IotService

Json representation sample

The following is an example json representation of an 'IotServiceIotServiceDeleteEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"IotServiceIotServiceDeleteEventNotification",
    "event": {
      "iotService" :
         {-- SEE IotService RESOURCE SAMPLE --}
   }
}
```

6.2.6 Service Catalog Service Catalog Create Event Notification

Notification ServiceCatalogCreateEvent case for resource ServiceCatalog

Json representation sample

The following is an example json representation of a 'ServiceCatalogServiceCatalogCreate EventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCatalogServiceCatalogCreateEventNotification",
    "event": {
      "serviceCatalog" :
         {-- SEE ServiceCatalog RESOURCE SAMPLE --}
   }
}
```

6.2.7 Service Catalog Service Catalog Change Event Notification

Notification ServiceCatalogChangeEvent case for resource ServiceCatalog

Json representation sample

The following is an example json representation of a 'ServiceCatalogService CatalogChangeEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCatalogServiceCatalogChangeEventNotification",
    "event": {
      "serviceCatalog" :
         {-- SEE ServiceCatalog RESOURCE SAMPLE --}
   }
}
```

6.2.8 Service Catalog Service Catalog Batch Event Notification

Notification ServiceCatalogBatchEvent case for resource ServiceCatalog

Json representation sample

The following is an example json representation of a 'ServiceCatalogService CatalogBatchEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCatalogServiceCatalogBatchEventNotification",
    "event": {
      "serviceCatalog" :
         {-- SEE ServiceCatalog RESOURCE SAMPLE --}
   }
}
```

6.2.9 Service Catalog Service Catalog Delete Event Notification

Notification ServiceCatalogDeleteEvent case for resource ServiceCatalog

Json representation sample

The following is an example json representation of a 'ServiceCatalogService CatalogDeleteEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCatalogServiceCatalogDeleteEventNotification",
    "event": {
      "serviceCatalog" :
         {-- SEE ServiceCatalog RESOURCE SAMPLE --}
   }
}
```

6.2.10 Service Category Service Category Create Event Notification

Notification ServiceCategoryCreateEvent case for resource ServiceCategory

Json representation sample

The following is an example json representation of a 'ServiceCategoryService CategoryCreateEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCategoryServiceCategoryCreateEventNotification",
    "event": {
      "serviceCategory" :
         {-- SEE ServiceCategory RESOURCE SAMPLE --}
   }
}
```

6.2.11 Service Category Service Category Change Event Notification

Notification ServiceCategoryChangeEvent case for resource ServiceCategory

Json representation sample

The following is an example json representation of a 'ServiceCategoryService CategoryChangeEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCategoryServiceCategoryChangeEventNotification",
    "event": {
      "serviceCategory" :
         {-- SEE ServiceCategory RESOURCE SAMPLE --}
   }
}
```

6.2.12 Service Category Service Category Delete Event Notification

Notification ServiceCategoryDeleteEvent case for resource ServiceCategory

Json representation sample

The following is an example json representation of a 'ServiceCategoryServiceCategory DeleteEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCategoryServiceCategoryDeleteEventNotification",
    "event": {
      "serviceCategory" :
         {-- SEE ServiceCategory RESOURCE SAMPLE --}
   }
}
```

6.2.13 Service Candidate Service Candidate Create Event Notification

Notification ServiceCandidateCreateEvent case for resource ServiceCandidate

Json representation sample

The following is an example json representation of a 'ServiceCandidateService CandidateCreateEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCandidateServiceCandidateCreateEventNotification",
    "event": {
      "serviceCandidate" :
         {-- SEE ServiceCandidate RESOURCE SAMPLE --}
   }
}
```

6.2.14 Service Candidate Service Candidate Change Event Notification

Notification ServiceCandidateChangeEvent case for resource ServiceCandidate

Json representation sample

The following is an example json representation of a 'ServiceCandidateService CandidateChangeEventNotification' notification object.

```
{
  "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCandidateServiceCandidateChangeEventNotification",
    "event": {
      "serviceCandidate" :
        {-- SEE ServiceCandidate RESOURCE SAMPLE --}
   }
}
```

6.2.15 Service Candidate Service Candidate Delete Event Notification

Notification ServiceCandidateDeleteEvent case for resource ServiceCandidate

Json representation sample

The following is an example json representation of a 'ServiceCandidateService CandidateDeleteEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceCandidateServiceCandidateDeleteEventNotification",
    "event": {
      "serviceCandidate" :
        {-- SEE ServiceCandidate RESOURCE SAMPLE --}
   }
}
```

6.2.16 Service Qualification Service Qualification Create Event Notification

Notification ServiceQualificationCreateEvent case for resource ServiceQualification

Json representation sample

The following is an example representation of a 'ServiceQualificationService QualificationCreateEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceQualificationServiceQualificationCreateEventNotification",
    "event": {
      "serviceQualification" :
        {-- SEE ServiceQualification RESOURCE SAMPLE --}
   }
}
```

6.2.17 Service Qualification Service Qualification Change Event Notification

Notification ServiceQualificationChangeEvent case for resource ServiceQualification

Json representation sample

The following is an example representation of a 'ServiceQualificationServiceQualification ChangeEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceQualificationServiceQualificationChangeEventNotification",
    "event": {
      "serviceQualification" :
         {-- SEE ServiceQualification RESOURCE SAMPLE --}
   }
}
```

6.2.18 Service Qualification Service Qualification Delete Event Notification

Notification ServiceQualificationDeleteEvent case for resource ServiceQualification

Json representation sample

The following is an example representation of a 'ServiceQualificationService QualificationDeleteEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceQualificationServiceQualificationDeleteEventNotification",
    "event": {
      "serviceQualification" :
         {-- SEE ServiceQualification RESOURCE SAMPLE --}
   }
}
```

6.2.19 Service Problem Service Problem Create Event Notification

Notification ServiceProblemCreateEvent case for resource ServiceProblem

Json representation sample

The following is an example representation of a 'ServiceProblemServiceProblem CreateEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceProblemServiceProblemCreateEventNotification",
    "event": {
      "serviceProblem" :
         {-- SEE ServiceProblem RESOURCE SAMPLE --}
   }
}
```

6.2.20 Service Problem Service Problem State Change Event Notification

Notification ServiceProblemStateChangeEvent case for resource ServiceProblem

Json representation sample

The following is an example json representation of a 'ServiceProblemService ProblemStateChangeEventNotification' notification object.

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"ServiceProblemServiceProblemStateChangeEventNotification",
   "event": {
     "serviceProblem" :
       {-- SEE ServiceProblem RESOURCE SAMPLE --}
  }
}
```

6.2.21 Service Problem Service Problem Attribute Value Change Event Notification

Notification ServiceProblemAttributeValueChangeEvent case for resource ServiceProblem

Json representation sample

The following is an example json representation of a 'ServiceProblemService ProblemAttributeValueChangeEventNotification' notification object.

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"ServiceProblemServiceProblemAttributeValueChangeEventNotification",
   "event": {
     "serviceProblem" :
       {-- SEE ServiceProblem RESOURCE SAMPLE --}
  }
}
```

6.2.22 Service Problem Service Problem Information Required Event Notification

Notification ServiceProblemInformationRequiredEvent case for resource ServiceProblem

Json representation sample

The following is an example json representation of a 'ServiceProblemService ProblemInformationRequiredEventNotification' notification object.

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"ServiceProblemServiceProblemInformationRequiredEventNotification",
   "event": {
     "serviceProblem" :
       {-- SEE ServiceProblem RESOURCE SAMPLE --}
  }
}
```

6.2.23 Iot Service Specification Iot Service Specification Create Event Notification

Notification IotServiceSpecificationCreateEvent case for resource IotServiceSpecification

Json representation sample

The following is an example json representation of an 'IotServiceSpecification IotServiceSpecificationCreateEventNotification' notification object.

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"IotServiceSpecificationIotServiceSpecificationCreateEventNotification",
   "event": {
     "iotServiceSpecification" :
        {-- SEE IotServiceSpecification RESOURCE SAMPLE --}
  }
}
```

6.2.24 Iot Service Specification Iot Service Specification Change Event Notification

Notification IotServiceSpecificationChangeEvent case for resource IotServiceSpecification

Json representation sample

The following is an example json representation of an 'IotServiceSpecification IotServiceSpecificationChangeEventNotification' notification object.

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"IotServiceSpecificationIotServiceSpecificationChangeEventNotification",
   "event": {
     "iotServiceSpecification" :
        {-- SEE IotServiceSpecification RESOURCE SAMPLE --}
  }
}
```

6.2.25 Iot Service Specification Iot Service Specification Delete Event Notification

Notification IotServiceSpecificationDeleteEvent case for resource IotServiceSpecification

Json representation sample

The following is an example representation of an 'IotServiceSpecification IotServiceSpecificationDeleteEventNotification' notification object.

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"IotServiceSpecificationIotServiceSpecificationDeleteEventNotification",
   "event": {
     "iotServiceSpecification" :
        {-- SEE IotServiceSpecification RESOURCE SAMPLE --}
  }
}
```

6.2.26 Service Test Service Test Create Event Notification

Notification ServiceTestCreateEvent case for resource ServiceTest

Json representation sample

The following is an example representation of a 'ServiceTestServiceTestCreateEventNotification' notification object.

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"ServiceTestServiceTestCreateEventNotification",
   "event": {
     "serviceTest" :
        {-- SEE ServiceTest RESOURCE SAMPLE --}
  }
}
```

6.2.27 Service Test Service Test Attribute Value Change Event Notification

Notification ServiceTestAttributeValueChangeEvent case for resource ServiceTest

Json representation sample

The following is an example representation of a 'ServiceTestService TestAttributeValueChangeEventNotification' notification object.

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"ServiceTestServiceTestAttributeValueChangeEventNotification",
   "event": {
     "serviceTest" :
        {-- SEE ServiceTest RESOURCE SAMPLE --}
  }
}
```

6.2.28 Service Test Service Test Delete Event Notification

Notification ServiceTestDeleteEvent case for resource ServiceTest

Json representation sample

The following is an example representation of a 'ServiceTestServiceTestDeleteEventNotification' notification object.

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"ServiceTestServiceTestDeleteEventNotification",
   "event": {
     "serviceTest" :
        {-- SEE ServiceTest RESOURCE SAMPLE --}
  }
}
```

6.2.29 Service Test Specification Service Test Specification Create Event Notification

Notification ServiceTestSpecificationCreateEvent case for resource ServiceTestSpecification

Json representation sample

The following is an example representation of a 'ServiceTestSpecification ServiceTestSpecificationCreateEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceTestSpecificationServiceTestSpecificationCreateEventNotification",
    "event": {
      "serviceTestSpecification" :
         {-- SEE ServiceTestSpecification RESOURCE SAMPLE --}
   }
}
```

6.2.30 Service Test Specification Service Test Specification Attribute Value Change Event Notification

Notification ServiceTestSpecificationAttributeValueChangeEvent case for resource ServiceTestSpecification

Json representation sample

The following is an example representation of a 'ServiceTest SpecificationServiceTestSpecificationAttributeValueChangeEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceTestSpecificationServiceTestSpecificationAttributeValueChangeEventNotification",
    "event": {
      "serviceTestSpecification" :
         {-- SEE ServiceTestSpecification RESOURCE SAMPLE --}
   }
}
```

6.2.31 Service Test Specification Service Test Specification Delete Event Notification

Notification ServiceTestSpecificationDeleteEvent case for resource ServiceTestSpecification

Json representation sample

The following is an example representation of a 'ServiceTestSpecification ServiceTestSpecificationDeleteEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"ServiceTestSpecificationServiceTestSpecificationDeleteEventNotification",
    "event": {
      "serviceTestSpecification" :
         {-- SEE ServiceTestSpecification RESOURCE SAMPLE --}
   }
}
```

6.2.32 Usage Consumption Report Request Usage Consumption Report Request State Change Event Notification

Notification UsageConsumptionReportRequestStateChangeEvent case for resource UsageConsumptionReportRequest

Json representation sample

The following is an example representation of a 'UsageConsumptionReport RequestUsageConsumptionReportRequestStateChangeEventNotification' notification object.

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

"eventType":"UsageConsumptionReportRequestUsageConsumptionReportRequestStateChangeEventNotifi
cation",
    "event": {
      "usageConsumptionReportRequest" :
         {-- SEE UsageConsumptionReportRequest RESOURCE SAMPLE --}
   }
}
```

6.2.33 User User Create Event Notification

Notification UserCreateEvent case for resource User

Json representation sample

The following is an example representation of a 'UserUserCreateEventNotification' notification object

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"UserUserCreateEventNotification",
    "event": {
      "user" :
         {-- SEE User RESOURCE SAMPLE --}
   }
}
```

6.2.34 User User Change Event Notification

Notification UserChangeEvent case for resource User

Json representation sample

The following is an example representation of a 'UserUserChangeEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"UserUserChangeEventNotification",
    "event": {
      "user" :
         {-- SEE User RESOURCE SAMPLE --}
   }
}
```

6.2.35 User User Delete Event Notification

Notification UserDeleteEvent case for resource User

Json representation sample

The following is an example representation of a 'UserUserDeleteEventNotification' notification object.

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"UserUserDeleteEventNotification",
    "event": {
      "user" :
         {-- SEE User RESOURCE SAMPLE --}
   }
}
```

7 API Operations

Remember the following Uniform Contract:

Table 1 – API Operations

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
Complete Update of an Entity	PUT Resource	PUT must be used to completely update a resource identified by its resource URI
Remove an Entity	DELETE Resource	DELETE must be used to remove a resource
Execute an Action on an Entity	POST on TASK Resource	POST must be used to execute Task Resources
Other Request Methods	POST on TASK Resource	GET and POST must not be used to tunnel other request methods.

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

Notifications are also described in a subsequent section.

7.1 Operations on Iot Service

7.1.1 List iot services

GET/iotService?fields=...&{filtering}

Description

This operation list iot 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 IotService resources.

Request

```
GET {apiRoot}/iotService
Accept: application/json
```

Response

```
200

[
{
   "description": "This  iot service ...",
   "endDate": "2019-05-13T00:00",
   "hasStarted": true,
   "isServiceEnabled": true,
   "isStateful": true,
   "serviceDate": "a string ...",
   "startDate": "2019-05-13T00:00",
   "startMode": "a string ...",
   "category": "a string ...",
   "href": "https:/host:port/tmf-api/iotService/v1/iotService/8520",
   "id": "8520",
   "name": "a string ...",
   "serviceType": "a string ...",
   "dataAccessEndPoint": {},
   "note": [
      {}
   ],
   "serviceOrder": [
      {}
   ],
   "place": [
      {}
   ],
   "relatedParty": [
      {}
   ],
   "serviceCharacteristic": [
      {}
   ],
   "serviceRelationship": [
      {}
   ],
```

```
    "serviceSpecification": {},
    "state": {},
    "supportingResource": [
      {}
    ],
    "supportingService": [
      {}
    ]
  }
  ]
```


7.1.2 Retrieve iot service

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

Description

This operation retrieves an iot 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 an IotService resource.

Request

```
GET {apiRoot}/iotService/8520
Accept: application/json
```

Response


```
200

{
   "description": "This  iot service ...",
   "endDate": "2019-05-13T00:00",
   "hasStarted": true,
   "isServiceEnabled": true,
   "isStateful": true,
   "serviceDate": "a string ...",
   "startDate": "2019-05-13T00:00",
   "startMode": "a string ...",
   "category": "a string ...",
   "href": "https:/host:port/tmf-api/iotService/v1/iotService/8520",
   "id": "8520",
   "name": "a string ...",
   "serviceType": "a string ...",
   "dataAccessEndPoint": {},
   "note": [
      {}
   ],
   "serviceOrder": [
      {}
   ],
   "place": [
```

```
            {}
        ],
        "relatedParty": [
            {}
        ],
        "serviceCharacteristic": [
            {}
        ],
        "serviceRelationship": [
            {}
        ],
        "serviceSpecification": {},
        "state": {},
        "supportingResource": [
            {}
        ],
        "supportingService": [
            {}
        ]
    }
```


7.1.3 Create iot service

POST/iotService

Description

This operation creates an iot service entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating an IotService, 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
description	
endDate	
hasStarted	
isServiceEnabled	
isStateful	
serviceDate	
startDate	
startMode	
category	
name	
serviceType	

Non-Mandatory Attributes	Rule
dataAccessEndPoint	
note	
serviceOrder	
place	
relatedParty	
serviceCharacteristic	
serviceRelationship	
serviceSpecification	
supportingResource	
supportingService	

Usage Samples

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

Request

```
POST {apiRoot}/iotService
Content-Type: application/json

{
   "state": {}
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/iotService/v1/iotService/8520",
   "id": "8520",
   "state": {}
}
```

7.1.4 Patch iot service

PATCH/iotService/{id}

Description

This operation allows partial updates of an iot service entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory [IETF RFC 7386], support of json/patch (http://tools.ietf.org/html/rfc5789) is optional [IETF RFC 5789].

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
description	
endDate	
hasStarted	
isServiceEnabled	
isStateful	
serviceDate	
startDate	
startMode	
category	
name	
serviceType	
dataAccessEndPoint	
note	
serviceOrder	
place	
relatedParty	
serviceCharacteristic	
serviceSpecification	
state	
supportingResource	
supportingService	

Non-Patchable Attributes	Rule
id	
href	
serviceRelationship	

Usage Samples

Here's an example of a request for patching an IotService resource.

Request
PATCH {apiRoot}/iotService/8520 Content-Type: application/merge-patch+json

```
{
   "name": "new name"
}
```


Response

200


```
{
   "description": "This  iot service ...",
   "endDate": "2019-05-13T00:00",
   "hasStarted": true,
   "isServiceEnabled": true,
   "isStateful": true,
   "serviceDate": "a string ...",
   "startDate": "2019-05-13T00:00",
   "startMode": "a string ...",
   "category": "a string ...",
   "href": "https:/host:port/tmf-api/iotService/v1/iotService/8520",
   "id": "8520",
   "name": "new name",
   "serviceType": "a string ...",
   "dataAccessEndPoint": {},
   "note": [
      {}
   ],
   "serviceOrder": [
      {}
   ],
   "place": [
      {}
   ],
   "relatedParty": [
      {}
   ],
   "serviceCharacteristic": [
      {}
   ],
   "serviceRelationship": [
      {}
   ],
   "serviceSpecification": {},
   "state": {},
   "supportingResource": [
      {}
   ],
   "supportingService": [
      {}
   ]
}
```


7.1.5 Delete iot service

DELETE/iotService/{id}

Description

This operation deletes an iot service entity.

Usage Samples

Here's an example of a request for deleting an IotService resource.

Request
DELETE {apiRoot}/iotService/42
Response
204

7.2 Operations on Service Catalog

7.2.1 List service catalogs

GET/serviceCatalog?fields=...&{filtering}

Description

This operation list service catalog 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 ServiceCatalog resources.

Request
GET {apiRoot}/serviceCatalog Accept: application/json
Response
200 [{ "category": [{}], "description": "This service catalog ...", "href": "https:/host:port/tmf-api/serviceCatalog/v1/serviceCatalog/3035", "id": "3035", "lastUpdate": "2019-05-13T00:00", "lifecycleStatus": "a string ...", "name": "a string ...", "relatedParty": [{}], "validFor": {},

```
    "version": "a string ..."
}
]
```

7.2.2 Retrieve service catalog

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

Description

This operation retrieves a service catalog 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 ServiceCatalog resource.

Request

```
GET {apiRoot}/serviceCatalog/3035
Accept: application/json
```

Response

```
200

{
   "category": [
      {}
   ],
   "description": "This  service catalog ...",
   "href": "https:/host:port/tmf-api/serviceCatalog/v1/serviceCatalog/3035",
   "id": "3035",
   "lastUpdate": "2019-05-13T00:00",
   "lifecycleStatus": "a string ...",
   "name": "a string ...",
   "relatedParty": [
      {}
   ],
   "validFor": {},
   "version": "a string ..."
}
```

7.2.3 Create service catalog

POST/serviceCatalog

Description

This operation creates a service catalog entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating a ServiceCatalog, 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
name	

Non-Mandatory Attributes	Rule
category	
description	
lastUpdate	
lifecycleStatus	
relatedParty	
validFor	
version	

Usage Samples

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

Request

```
POST {apiRoot}/serviceCatalog
Content-Type: application/json

{
   "name": "a string ..."
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/serviceCatalog/v1/serviceCatalog/3035",
   "id": "3035",
   "name": "a string ..."
}
```

7.2.4 Patch service catalog

PATCH/serviceCatalog/{id}

Description

This operation allows partial updates of a service catalog entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory [IETF RFC 7386], support of json/patch (http://tools.ietf.org/html/rfc5789) is optional [IETF RFC 5786].

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	
lifecycleStatus	
name	
relatedParty	
validFor	
version	

Non-Patchable Attributes	Rule
id	
href	
lastUpdate	

Usage Samples

Here's an example of a request for patching a ServiceCatalog resource.

Request

```
PATCH {apiRoot}/serviceCatalog/3035
Content-Type: application/merge-patch+json

{
   "name": "new name"
}
```

Response

200

```
{
   "category": [
      {}
   ],
   "description": "This  service catalog ...",
   "href": "https:/host:port/tmf-api/serviceCatalog/v1/serviceCatalog/3035",
   "id": "3035",
   "lastUpdate": "2019-05-13T00:00",
   "lifecycleStatus": "a string ...",
   "name": "new name",
   "relatedParty": [
      {}
   ],
   "validFor": {},
   "version": "a string ..."
}
```


7.2.5 Delete service catalog

DELETE/serviceCatalog/{id}

Description

This operation deletes a service catalog entity.

Usage Samples

Here's an example of a request for deleting a ServiceCatalog resource.

Request
DELETE {apiRoot}/serviceCatalog/42
Response
204

7.3 Operations on Service Category

7.3.1 List service categories

GET/serviceCategory?fields=...&{filtering}

Description

This operation list service category 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 ServiceCategory resources.

Request

```
GET {apiRoot}/serviceCategory
Accept: application/json
```

Response

```
200

[
{
   "category": [
      {}
   ],
   "description": "This  service category ...",
   "href": "https:/host:port/tmf-api/serviceCategory/v1/serviceCategory/6889",
   "id": "6889",
   "isRoot": false,
   "lastUpdate": "2019-05-13T00:00",
   "lifecycleStatus": "a string ...",
   "name": "a string ...",
   "parentId": "203",
   "serviceCandidate": [
      {}
   ],
   "validFor": {},
   "version": "a string ..."
}
]
```

7.3.2 Retrieve service category

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

Description

This operation retrieves a service category 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 ServiceCategory resource.

Request

```
GET {apiRoot}/serviceCategory/6889
Accept: application/json
```

Response

```
200

{
   "category": [
      {}
   ],
   "description": "This  service category ...",
   "href": "https:/host:port/tmf-api/serviceCategory/v1/serviceCategory/6889",
   "id": "6889",
   "isRoot": false,
   "lastUpdate": "2019-05-13T00:00",
   "lifecycleStatus": "a string ...",
   "name": "a string ...",
   "parentId": "203",
   "serviceCandidate": [
      {}
   ],
   "validFor": {},
   "version": "a string ..."
}
```

7.3.3 Create service category

POST/serviceCategory

Description

This operation creates a service category entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating a ServiceCategory, 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
name	

Non-Mandatory Attributes	Rule
category	
description	
isRoot	
lastUpdate	
lifecycleStatus	
parentId	
serviceCandidate	
validFor	
version	

Usage Samples

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

Request

```
POST {apiRoot}/serviceCategory
Content-Type: application/json

{
   "name": "a string ..."
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/serviceCategory/v1/serviceCategory/6889",
   "id": "6889",
   "name": "a string ..."
}
```

7.3.4 Patch service category

PATCH/serviceCategory/{id}

Description

This operation allows partial updates of a service category entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory [IETF RFC 7386], support of json/patch (http://tools.ietf.org/html/rfc5789) is optional [IETF RFC 5789].

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	
isRoot	
lifecycleStatus	
name	
parentId	

Patchable Attributes	Rule
serviceCandidate	
validFor	
version	

Non-Patchable Attributes	Rule
id	
href	
lastUpdate	

Usage Samples

Here's an example of a request for patching a ServiceCategory resource.

Request

```
PATCH {apiRoot}/serviceCategory/6889
Content-Type: application/merge-patch+json

{
   "name": "new name"
}
```

Response

```
200

{
   "category": [
      {}
   ],
   "description": "This  service category ...",
   "href": "https:/host:port/tmf-api/serviceCategory/v1/serviceCategory/6889",
   "id": "6889",
   "isRoot": false,
   "lastUpdate": "2019-05-13T00:00",
   "lifecycleStatus": "a string ...",
   "name": "new name",
   "parentId": "203",
   "serviceCandidate": [
      {}
   ],
   "validFor": {},
   "version": "a string ..."
}
```

7.3.5 Delete service category

DELETE/serviceCategory/{id}

Description

This operation deletes a service category entity.

Usage Samples

Here's an example of a request for deleting a ServiceCategory resource.

Request
DELETE {apiRoot}/serviceCategory/42
Response
204

7.4 Operations on Service Candidate

7.4.1 List service candidates

GET/serviceCandidate?fields=...&{filtering}

Description

This operation list service candidate 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 ServiceCandidate resources.

Request
GET {apiRoot}/serviceCandidate Accept: application/json
Response
200 [{ "category": [{}], "description": "This service candidate ...", "href": "https:/host:port/tmf-api/serviceCandidate/v1/serviceCandidate/9315", "id": "9315",

```
    "lastUpdate": "2019-05-13T00:00",
    "lifecycleStatus": "a string ...",
    "name": "a string ...",
    "serviceSpecification": {},
    "validFor": {},
    "version": "a string ..."
}
]
```

7.4.2 Retrieve service candidate

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

Description

This operation retrieves a service candidate 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 ServiceCandidate resource.

Request

```
GET {apiRoot}/serviceCandidate/9315
Accept: application/json
```

Response

```
200

{
   "category": [
      {}
   ],
   "description": "This  service candidate ...",
   "href": "https:/host:port/tmf-api/serviceCandidate/v1/serviceCandidate/9315",
   "id": "9315",
   "lastUpdate": "2019-05-13T00:00",
   "lifecycleStatus": "a string ...",
   "name": "a string ...",
   "serviceSpecification": {},
   "validFor": {},
   "version": "a string ..."
}
```

7.4.3 Create service candidate

POST/serviceCandidate

Description

This operation creates a service candidate entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating a ServiceCandidate, 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
name	

Non-Mandatory Attributes	Rule
category	
description	
lastUpdate	
lifecycleStatus	
serviceSpecification	
validFor	
version	

Usage Samples

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

Request

```
POST {apiRoot}/serviceCandidate
Content-Type: application/json

{
   "name": "a string ..."
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/serviceCandidate/v1/serviceCandidate/9315",
   "id": "9315",
   "name": "a string ..."
}
```

7.4.4 Patch service candidate

PATCH/serviceCandidate/{id}

Description

This operation allows partial updates of a service candidate entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory [IETF RFC 7386], support of json/patch (http://tools.ietf.org/html/rfc5789) is optional [IETF RFC 5789].

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	
lifecycleStatus	
name	
serviceSpecification	
validFor	
version	

Non-Patchable Attributes	Rule
id	
href	
lastUpdate	

Usage Samples

Here's an example of a request for patching a ServiceCandidate resource.

Request

```
PATCH {apiRoot}/serviceCandidate/9315
Content-Type: application/merge-patch+json

{
   "name": "new name"
}
```

Response

200


```
{
   "category": [
      {}
   ],
   "description": "This  service candidate ...",
   "href": "https:/host:port/tmf-api/serviceCandidate/v1/serviceCandidate/9315",
   "id": "9315",
   "lastUpdate": "2019-05-13T00:00",
   "lifecycleStatus": "a string ...",
   "name": "new name",
   "serviceSpecification": {},
   "validFor": {},
   "version": "a string ..."
}
```


7.4.5 Delete service candidate

DELETE/serviceCandidate/{id}

Description

This operation deletes a service candidate entity.

Usage Samples

Here's an example of a request for deleting a ServiceCandidate resource.

Request
DELETE {apiRoot}/serviceCandidate/42
Response
204

7.5 Operations on Service Qualification

7.5.1 List service qualifications

GET/serviceQualification?fields=...&{filtering}

Description

This operation list service qualification 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 ServiceQualification resources.

Request

```
GET {apiRoot}/serviceQualification
Accept: application/json
```

Response

```
200

[
{
   "description": "This  service qualification ...",
   "effectiveQualificationDate": "2019-05-13T00:00",
   "estimatedResponseDate": "2019-05-13T00:00",
   "expectedQualificationDate": "2019-05-13T00:00",
   "expirationDate": "2019-05-13T00:00",
   "externalId": "506",
   "href": "https:/host:port/tmf-api/serviceQualification/v1/serviceQualification/7775",
   "id": "7775",
   "provideAlternative": true,
   "provideOnlyAvailable": true,
   "provideUnavailabilityReason": true,
   "qualificationResult": "a string ...",
   "relatedParty": [
      {}
   ],
   "serviceQualificationDate": "2019-05-13T00:00",
   "serviceQualificationItem": [
      {}
   ],
   "state": "a string ..."
}
]
```

7.5.2 Retrieve service qualification

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

Description

This operation retrieves a service qualification 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 ServiceQualification resource.

Request

```
GET {apiRoot}/serviceQualification/7775
Accept: application/json
```

Response

200


```
{
   "description": "This  service qualification ...",
   "effectiveQualificationDate": "2019-05-13T00:00",
   "estimatedResponseDate": "2019-05-13T00:00",
   "expectedQualificationDate": "2019-05-13T00:00",
   "expirationDate": "2019-05-13T00:00",
   "externalId": "506",
   "href": "https:/host:port/tmf-api/serviceQualification/v1/serviceQualification/7775",
   "id": "7775",
   "provideAlternative": true,
   "provideOnlyAvailable": true,
   "provideUnavailabilityReason": true,
   "qualificationResult": "a string ...",
   "relatedParty": [
      {}
   ],
   "serviceQualificationDate": "2019-05-13T00:00",
   "serviceQualificationItem": [
      {}
   ],
   "state": "a string ..."
}
```


7.5.3 Create service qualification

POST/serviceQualification

Description

This operation creates a service qualification entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating a ServiceQualification, 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
serviceQualificationItem	

Non-Mandatory Attributes	Rule
description	
effectiveQualificationDate	
estimatedResponseDate	
expectedQualificationDate	
expirationDate	
externalId	
provideAlternative	

Non-Mandatory Attributes	Rule
provideOnlyAvailable	
provideUnavailabilityReason	
qualificationResult	
relatedParty	
serviceQualificationDate	
state	

Usage Samples

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

Request

```
POST {apiRoot}/serviceQualification
Content-Type: application/json

{
   "serviceQualificationItem": [
      {}
   ]
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/serviceQualification/v1/serviceQualification/7775",
   "id": "7775",
   "serviceQualificationItem": [
      {}
   ]
}
```

7.5.4 Patch service qualification

PATCH/serviceQualification/{id}

Description

This operation allows partial updates of a service qualification entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory [IETF RFC 7386], support of json/patch (http://tools.ietf.org/html/rfc5789) is optional [IETF RFC 5789].

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
description	
effectiveQualificationDate	
estimatedResponseDate	
expectedQualificationDate	
expirationDate	
externalId	
provideAlternative	
provideOnlyAvailable	
provideUnavailabilityReason	
qualificationResult	
relatedParty	
serviceQualificationDate	
serviceQualificationItem	
state	

Non-Patchable Attributes	Rule
id	
href	

Usage Samples

Here's an example of a request for patching a ServiceQualification resource.

Request

```
PATCH {apiRoot}/serviceQualification/7775
Content-Type: application/merge-patch+json

{
   "name": "new name"
}
```

Response

```
200

{
   "description": "This  service qualification ...",
   "effectiveQualificationDate": "2019-05-13T00:00",
   "estimatedResponseDate": "2019-05-13T00:00",
   "expectedQualificationDate": "2019-05-13T00:00",
   "expirationDate": "2019-05-13T00:00",
```

```
    "externalId": "506",
    "href": "https:/host:port/tmf-api/serviceQualification/v1/serviceQualification/7775",
    "id": "7775",
    "provideAlternative": true,
    "provideOnlyAvailable": true,
    "provideUnavailabilityReason": true,
    "qualificationResult": "a string ...",
    "relatedParty": [
      {}
    ],
    "serviceQualificationDate": "2019-05-13T00:00",
    "serviceQualificationItem": [
      {}
    ],
    "state": "a string ...",
    "name": "new name"
}
```


7.5.5 Delete service qualification

DELETE/serviceQualification/{id}

Description

This operation deletes a service qualification entity.

Usage Samples

Here's an example of a request for deleting a ServiceQualification resource.

Request
DELETE {apiRoot}/serviceQualification/42
Response
204

7.6 Operations on Service Problem

7.6.1 List service problems

GET/serviceProblem?fields=...&{filtering}

Description

This operation list service problem 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 ServiceProblem resources.

Request

GET {apiRoot}/serviceProblem
Accept: application/json

Response

200


```
[
{
   "affectedLocation": [
      {}
   ],
   "affectedNumberOfServices": 69,
   "affectedResource": [
      {}
   ],
   "affectedService": [
      {}
   ],
   "associatedSLAViolation": [
      {}
   ],
   "associatedTroubleTicket": [
      {}
   ],
   "category": "a string ...",
   "comment": [
      {}
   ],
   "correlationId": "305",
   "description": "This  service problem ...",
   "extensionInfo": [
      {}
   ],
   "firstAlert": {},
   "href": "https:/host:port/tmf-api/serviceProblem/v1/serviceProblem/8561",
   "id": "8561",
   "impactImportanceFactor": "a string ...",
   "impactPatterns": {},
   "originatingSystem": "a string ...",
   "originatorParty": {},
   "parentProblem": [
      {}
   ],
   "priority": 51,
   "problemEscalation": "a string ...",
   "reason": "a string ...",
   "relatedEvent": [
      {}
   ],
   "relatedObject": [
      {}
   ],
   "relatedParty": [
      {}
   ],
```

```
    "resolutionDate": "2019-05-13T00:00",
    "responsibleParty": {},
    "rootCauseResource": [
      {}
    ],
    "rootCauseService": [
      {}
    ],
    "status": "a string ...",
    "statusChangeDate": "2019-05-13T00:00",
    "statusChangeReason": "a string ...",
    "timeChanged": "2019-05-13T00:00",
    "timeRaised": "2019-05-13T00:00",
    "trackingRecord": [
      {}
    ],
    "underlyingAlarm": [
      {}
    ],
    "underlyingProblem": [
      {}
    ]
  }
]
```


7.6.2 Retrieve service problem

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

Description

This operation retrieves a service problem 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 ServiceProblem resource.

Request

```
GET {apiRoot}/serviceProblem/8561
Accept: application/json
```

Response


```
200

{
  "affectedLocation": [
    {}
  ],
  "affectedNumberOfServices": 69,
  "affectedResource": [
    {}
  ],
  "affectedService": [
```

```
        {}
    ],
    "associatedSLAViolation": [
        {}
    ],
    "associatedTroubleTicket": [
        {}
    ],
    "category": "a string ...",
    "comment": [
        {}
    ],
    "correlationId": "305",
    "description": "This  service problem ...",
    "extensionInfo": [
        {}
    ],
    "firstAlert": {},
    "href": "https:/host:port/tmf-api/serviceProblem/v1/serviceProblem/8561",
    "id": "8561",
    "impactImportanceFactor": "a string ...",
    "impactPatterns": {},
    "originatingSystem": "a string ...",
    "originatorParty": {},
    "parentProblem": [
        {}
    ],
    "priority": 51,
    "problemEscalation": "a string ...",
    "reason": "a string ...",
    "relatedEvent": [
        {}
    ],
    "relatedObject": [
        {}
    ],
    "relatedParty": [
        {}
    ],
    "resolutionDate": "2019-05-13T00:00",
    "responsibleParty": {},
    "rootCauseResource": [
        {}
    ],
    "rootCauseService": [
        {}
    ],
    "status": "a string ...",
    "statusChangeDate": "2019-05-13T00:00",
    "statusChangeReason": "a string ...",
    "timeChanged": "2019-05-13T00:00",
    "timeRaised": "2019-05-13T00:00",
    "trackingRecord": [
        {}
    ],
    "underlyingAlarm": [
        {}
    ],
    "underlyingProblem": [
        {}
    ]
}
```

7.6.3 Create service problem

POST/serviceProblem

Description

This operation creates a service problem entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating a ServiceProblem, 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
category	
priority	
description	
reason	
originatorParty	

Non-Mandatory Attributes	Rule
affectedLocation	
affectedNumberOfServices	
affectedResource	
affectedService	
associatedSLAViolation	
associatedTroubleTicket	
comment	
correlationId	
extensionInfo	
firstAlert	
impactImportanceFactor	
impactPatterns	
originatingSystem	
parentProblem	
problemEscalation	
relatedEvent	
relatedObject	
relatedParty	
resolutionDate	
responsibleParty	

Non-Mandatory Attributes	Rule
rootCauseResource	
rootCauseService	
status	
statusChangeDate	
statusChangeReason	
timeChanged	
timeRaised	
trackingRecord	
underlyingAlarm	
underlyingProblem	

Usage Samples

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

Request

```
POST {apiRoot}/serviceProblem
Content-Type: application/json

{
   "category": "a string ...",
   "description": "This service problem ...",
   "originatorParty": {},
   "priority": 51,
   "reason": "a string ..."
}
```

Response

```
201

{
   "category": "a string ...",
   "description": "This service problem ...",
   "href": "https:/host:port/tmf-api/serviceProblem/v1/serviceProblem/8561",
   "id": "8561",
   "originatorParty": {},
   "priority": 51,
   "reason": "a string ..."
}
```

7.6.4 Patch service problem

PATCH/serviceProblem/{id}

Description

This operation allows partial updates of a service problem entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory [IETF RFC 7386], support of json/patch (http://tools.ietf.org/html/rfc5789) is optional [IETF RFC 5789].

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
affectedLocation	
affectedNumberOfServices	
affectedResource	
affectedService	
associatedSLAViolation	
associatedTroubleTicket	
category	
comment	
description	
extensionInfo	
impactImportanceFactor	
impactPatterns	
originatorParty	
parentProblem	
priority	
problemEscalation	
reason	
relatedEvent	
relatedObject	
relatedParty	
resolutionDate	
responsibleParty	
rootCauseResource	
rootCauseService	

Patchable Attributes	Rule
status	
statusChangeDate	
statusChangeReason	
timeChanged	
underlyingAlarm	
underlyingProblem	

Non-Patchable Attributes	Rule
correlationId	
firstAlert	
href	
id	
originatingSystem	
timeRaised	
trackingRecord	

Usage Samples

Here's an example of a request for patching a ServiceProblem resource.

Request

```
PATCH {apiRoot}/serviceProblem/8561
Content-Type: application/merge-patch+json

{
   "name": "new name"
}
```

Response

```
200

{
   "affectedLocation": [
      {}
   ],
   "affectedNumberOfServices": 69,
   "affectedResource": [
      {}
   ],
   "affectedService": [
      {}
   ],
   "associatedSLAViolation": [
```

```
        {}
    ],
    "associatedTroubleTicket": [
        {}
    ],
    "category": "a string ...",
    "comment": [
        {}
    ],
    "correlationId": "305",
    "description": "This  service problem ...",
    "extensionInfo": [
        {}
    ],
    "firstAlert": {},
    "href": "https:/host:port/tmf-api/serviceProblem/v1/serviceProblem/8561",
    "id": "8561",
    "impactImportanceFactor": "a string ...",
    "impactPatterns": {},
    "originatingSystem": "a string ...",
    "originatorParty": {},
    "parentProblem": [
        {}
    ],
    "priority": 51,
    "problemEscalation": "a string ...",
    "reason": "a string ...",
    "relatedEvent": [
        {}
    ],
    "relatedObject": [
        {}
    ],
    "relatedParty": [
        {}
    ],
    "resolutionDate": "2019-05-13T00:00",
    "responsibleParty": {},
    "rootCauseResource": [
        {}
    ],
    "rootCauseService": [
        {}
    ],
    "status": "a string ...",
    "statusChangeDate": "2019-05-13T00:00",
    "statusChangeReason": "a string ...",
    "timeChanged": "2019-05-13T00:00",
    "timeRaised": "2019-05-13T00:00",
    "trackingRecord": [
        {}
    ],
    "underlyingAlarm": [
        {}
    ],
    "underlyingProblem": [
        {}
    ],
    "name": "new name"
}
```

7.6.5 Delete service problem

DELETE/serviceProblem/{id}

Description

This operation deletes a service problem entity.

Usage Samples

Here's an example of a request for deleting a ServiceProblem resource.

Request
DELETE {apiRoot}/serviceProblem/42
Response
204

7.7 Operations on Iot Service Specification

7.7.1 List iot service specifications

GET/iotServiceSpecification?fields=...&{filtering}

Description

This operation list iot service specification 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 IotServiceSpecification resources.

Request
GET {apiRoot}/iotServiceSpecification Accept: application/json
Response
200 [{ "description": "This iot service specification ...", "href": "https:/host:port/tmf-api/iotServiceSpecification/v1/iotServiceSpecification/4530", "id": "4530", "isBundle": true, "lastUpdate": "2019-05-13T00:00", "lifecycleStatus": "a string ...",

```
    "name": "a string ...",
    "version": "a string ...",
    "attachment": [
      {}
    ],
    "relatedParty": [
      {}
    ],
    "resourceSpecification": [
      {}
    ],
    "serviceLevelSpecification": [
      {}
    ],
    "serviceSpecCharacteristic": [
      {}
    ],
    "serviceSpecRelationship": [
      {}
    ],
    "targetServiceSchema": {},
    "validFor": {}
  }
]
```


7.7.2 Retrieve iot service specification

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

Description

This operation retrieves an iot service specification 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 an IotServiceSpecification resource.

Request

```
GET {apiRoot}/iotServiceSpecification/4530
Accept: application/json
```

Response

200


```
{
    "description": "This  iot service specification ...",
    "href": "https:/host:port/tmf-api/iotServiceSpecification/v1/iotServiceSpecification/4530",
    "id": "4530",
    "isBundle": true,
    "lastUpdate": "2019-05-13T00:00",
    "lifecycleStatus": "a string ...",
    "name": "a string ...",
```

```
    "version": "a string ...",
    "attachment": [
      {}
    ],
    "relatedParty": [
      {}
    ],
    "resourceSpecification": [
      {}
    ],
    "serviceLevelSpecification": [
      {}
    ],
    "serviceSpecCharacteristic": [
      {}
    ],
    "serviceSpecRelationship": [
      {}
    ],
    "targetServiceSchema": {},
    "validFor": {}
  }
```


7.7.3 Create iot service specification

POST/iotServiceSpecification

Description

This operation creates an iot service specification entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating an IotServiceSpecification, 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
name	

Non-Mandatory Attributes	Rule
description	
isBundle	
lastUpdate	
lifecycleStatus	
version	
attachment	
relatedParty	
resourceSpecification	
serviceLevelSpecification	
serviceSpecCharacteristic	
serviceSpecRelationship	

Non-Mandatory Attributes	Rule
targetServiceSchema	
validFor	

Usage Samples

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

Request

```
POST {apiRoot}/iotServiceSpecification
Content-Type: application/json

{
   "name": "a string ..."
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/iotServiceSpecification/v1/iotServiceSpecification/4530",
   "id": "4530",
   "name": "a string ..."
}
```

7.7.4 Patch iot service specification

PATCH/iotServiceSpecification/{id}

Description

This operation allows partial updates of an iot service specification entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory [IETF RFC 7386], support of json/patch (http://tools.ietf.org/html/rfc5789) is optional [IETF RFC 5789].

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
description	
isBundle	
lifecycleStatus	
name	

Patchable Attributes	Rule
version	
attachment	
relatedParty	
resourceSpecification	
serviceLevelSpecification	
serviceSpecCharacteristic	
serviceSpecRelationship	
targetServiceSchema	
validFor	

Non-Patchable Attributes	Rule
id	
href	
lastUpdate	

Usage Samples

Here's an example of a request for patching an IotServiceSpecification resource.

Request

```
PATCH {apiRoot}/iotServiceSpecification/4530
Content-Type: application/merge-patch+json

{
   "name": "new name"
}
```

Response

```
200

{
   "description": "This  iot service specification ...",
   "href": "https:/host:port/tmf-api/iotServiceSpecification/v1/iotServiceSpecification/4530",
   "id": "4530",
   "isBundle": true,
   "lastUpdate": "2019-05-13T00:00",
   "lifecycleStatus": "a string ...",
   "name": "new name",
   "version": "a string ...",
   "attachment": [
      {}
   ],
   "relatedParty": [
      {}
```

```
    ],
    "resourceSpecification": [
      {}
    ],
    "serviceLevelSpecification": [
      {}
    ],
    "serviceSpecCharacteristic": [
      {}
    ],
    "serviceSpecRelationship": [
      {}
    ],
    "targetServiceSchema": {},
    "validFor": {}
}
```


7.7.5 Delete iot service specification

DELETE/iotServiceSpecification/{id}

Description

This operation deletes an iot service specification entity.

Usage Samples

Here's an example of a request for deleting an IotServiceSpecification resource.

Request
DELETE {apiRoot}/iotServiceSpecification/42
Response
204

7.8 Operations on Service Test

7.8.1 List service tests

GET/serviceTest?fields=...&{filtering}

Description

This operation list service test 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 ServiceTest resources.

Request

```
GET {apiRoot}/serviceTest
Accept: application/json
```

Response

```
200

[
{
   "characteristic": [
      {}
   ],
   "description": "This  service test ...",
   "endDateTime": "2019-05-13T00:00",
   "href": "https:/host:port/tmf-api/serviceTest/v1/serviceTest/1983",
   "id": "1983",
   "mode": "a string ...",
   "name": "a string ...",
   "relatedService": {},
   "startDateTime": "2019-05-13T00:00",
   "state": "a string ...",
   "testMeasure": [
      {}
   ],
   "testSpecification": {}
}
]
```

7.8.2 Retrieve service test

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

Description

This operation retrieves a service test 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 ServiceTest resource.

Request

```
GET {apiRoot}/serviceTest/1983
Accept: application/json
```

Response

```
200

{
   "characteristic": [
      {}
   ],
   "description": "This  service test ...",
   "endDateTime": "2019-05-13T00:00",
   "href": "https:/host:port/tmf-api/serviceTest/v1/serviceTest/1983",
   "id": "1983",
   "mode": "a string ...",
   "name": "a string ...",
   "relatedService": {},
   "startDateTime": "2019-05-13T00:00",
   "state": "a string ...",
   "testMeasure": [
      {}
   ],
   "testSpecification": {}
}
```

7.8.3 Create service test

POST/serviceTest

Description

This operation creates a service test entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating a ServiceTest, 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
name	
relatedService	
testSpecification	

Non-Mandatory Attributes	Rule
characteristic	
description	
endDateTime	
mode	
startDateTime	
state	
testMeasure	

Usage Samples

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

Request

```
POST {apiRoot}/serviceTest
Content-Type: application/json

{
   "name": "a string ...",
   "relatedService": {},
   "testSpecification": {}
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/serviceTest/v1/serviceTest/1983",
   "id": "1983",
   "name": "a string ...",
   "relatedService": {},
   "testSpecification": {}
}
```

7.8.4 Patch service test

PATCH/serviceTest/{id}

Description

This operation allows partial updates of a service test entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory [IETF RFC 7386], support of json/patch (http://tools.ietf.org/html/rfc5789) is optional IETF [RFC 5789].

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
characteristic	
description	
endDateTime	
mode	
name	

Patchable Attributes	Rule
relatedService	
startDateTime	
state	
testMeasure	
testSpecification	

Non-Patchable Attributes	Rule
href	
id	

Usage Samples

Here's an example of a request for patching a ServiceTest resource.

Request

```
PATCH {apiRoot}/serviceTest/1983
Content-Type: application/merge-patch+json

{
   "name": "new name"
}
```

Response

```
200

{
   "characteristic": [
      {}
   ],
   "description": "This  service test ...",
   "endDateTime": "2019-05-13T00:00",
   "href": "https:/host:port/tmf-api/serviceTest/v1/serviceTest/1983",
   "id": "1983",
   "mode": "a string ...",
   "name": "new name",
   "relatedService": {},
   "startDateTime": "2019-05-13T00:00",
   "state": "a string ...",
   "testMeasure": [
      {}
   ],
   "testSpecification": {}
}
```

7.8.5 Delete service test

DELETE/serviceTest/{id}

Description

This operation deletes a service test entity.

Usage Samples

Here's an example of a request for deleting a ServiceTest resource.

Request
DELETE {apiRoot}/serviceTest/42
Response
204

7.9 Operations on Service Test Specification

7.9.1 List service test specifications

GET/serviceTestSpecification?fields=...&{filtering}

Description

This operation list service test specification 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 ServiceTestSpecification resources.

Request
GET {apiRoot}/serviceTestSpecification Accept: application/json
Response
200 [{ "description": "This service test specification ...", "href": "https:/host:port/tmf-api/serviceTestSpecification/v1/serviceTestSpecification/8023", "id": "8023", "name": "a string ...", "relatedServiceSpecification": {}, "testMeasureDefinition": [

```
        {}
    ],
    "validFor": {},
    "version": "a string ..."
}
]
```

7.9.2 Retrieve service test specification

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

Description

This operation retrieves a service test specification 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 ServiceTestSpecification resource.

Request

```
GET {apiRoot}/serviceTestSpecification/8023
Accept: application/json
```

Response

```
200

{
    "description": "This service test specification ...",
    "href": "https:/host:port/tmf-api/serviceTestSpecification/v1/serviceTestSpecification/8023",
    "id": "8023",
    "name": "a string ...",
    "relatedServiceSpecification": {},
    "testMeasureDefinition": [
        {}
    ],
    "validFor": {},
    "version": "a string ..."
}
```

7.9.3 Create service test specification

POST/serviceTestSpecification

Description

This operation creates a service test specification entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating a ServiceTestSpecification, 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
name	
relatedServiceSpecification	

Non-Mandatory Attributes	Rule
description	
testMeasureDefinition	
validFor	
version	

Usage Samples

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

Request

```
POST {apiRoot}/serviceTestSpecification
Content-Type: application/json

{
   "name": "a string ...",
   "relatedServiceSpecification": {}
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/serviceTestSpecification/v1/serviceTestSpecification/8023",
   "id": "8023",
   "name": "a string ...",
   "relatedServiceSpecification": {}
}
```

7.9.4 Patch service test specification

PATCH/serviceTestSpecification/{id}

Description

This operation allows partial updates of a service test specification entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory [IETF RFC 7386], support of json/patch (http://tools.ietf.org/html/rfc5789) is optional IETF [RFC 5789].

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
description	
name	
relatedServiceSpecification	
testMeasureDefinition	
version	

Non-Patchable Attributes	Rule
href	
id	
validFor	

Usage Samples

Here's an example of a request for patching a ServiceTestSpecification resource.

Request

```
PATCH {apiRoot}/serviceTestSpecification/8023
Content-Type: application/merge-patch+json

{
   "name": "new name"
}
```

Response

```
200

{
   "description": "This service test specification ...",
   "href": "https:/host:port/tmf-api/serviceTestSpecification/v1/serviceTestSpecification/8023",
   "id": "8023",
   "name": "new name",
   "relatedServiceSpecification": {},
   "testMeasureDefinition": [
      {}
   ],
   "validFor": {},
   "version": "a string ..."
}
```

7.9.5 Delete service test specification

DELETE/serviceTestSpecification/{id}

Description

This operation deletes a service test specification entity.

Usage Samples

Here's an example of a request for deleting a ServiceTestSpecification resource.

Request
DELETE {apiRoot}/serviceTestSpecification/42
Response
204

7.10 Operations on Usage Consumption Report Request

7.10.1 List usage consumption report requests

GET/usageConsumptionReportRequest?fields=...&{filtering}

Description

This operation list usage consumption report request 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 UsageConsumptionReportRequest resources.

Request
GET {apiRoot}/usageConsumptionReportRequest Accept: application/json
Response
200 [{ "bucket": [{}], "creationDate": "2019-05-13T00:00", "href": "https:/host:port/tmf-api/usageConsumptionReportRequest/v1/usageConsumptionReportRequest/4925",

```
    "id": "4925",
    "lastUpdate": "2019-05-13T00:00",
    "product": {},
    "relatedParty": [
      {}
    ],
    "status": "a string ...",
    "usageConsumptionReport": {},
    "validPeriod": {}
}
]
```


7.10.2 Retrieve usage consumption report request

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

Description

This operation retrieves a usage consumption report request 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 UsageConsumptionReportRequest resource.

Request

```
GET {apiRoot}/usageConsumptionReportRequest/4925
Accept: application/json
```

Response

```
200

{
   "bucket": [
      {}
   ],
   "creationDate": "2019-05-13T00:00",
   "href": "https:/host:port/tmf-
api/usageConsumptionReportRequest/v1/usageConsumptionReportRequest/4925",
   "id": "4925",
   "lastUpdate": "2019-05-13T00:00",
   "product": {},
   "relatedParty": [
      {}
   ],
   "status": "a string ...",
   "usageConsumptionReport": {},
   "validPeriod": {}
}
```

7.10.3 Create usage consumption report request

POST/usageConsumptionReportRequest

Description

This operation creates a usage consumption report request entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating a UsageConsumptionReportRequest, 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

Non-Mandatory Attributes	Rule
bucket	
creationDate	
lastUpdate	
product	
relatedParty	
status	
usageConsumptionReport	
validPeriod	

Usage Samples

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

Request

```
POST {apiRoot}/usageConsumptionReportRequest
Content-Type: application/json

{}
```

Response

```
201

{
   "href": "https:/host:port/tmf-
api/usageConsumptionReportRequest/v1/usageConsumptionReportRequest/4925",
   "id": "4925"
}
```

7.10.4 Delete usage consumption report request

DELETE/usageConsumptionReportRequest/{id}

Description

This operation deletes a usage consumption report request entity.

Usage Samples

Here's an example of a request for deleting a UsageConsumptionReportRequest resource.

Request
DELETE {apiRoot}/usageConsumptionReportRequest/42
Response
204

7.11 Operations on User

7.11.1 List users

GET/user?fields=...&{filtering}

Description

This operation list user 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 User resources.

Request
GET {apiRoot}/user Accept: application/json
Response

200
 	
[
{
"description": "This user ..."
}
]

7.11.2 Retrieve user

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

Description

This operation retrieves a user 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 User resource.

Request
GET {apiRoot}/user/42 Accept: application/json
Response

200
 	
{
"description": "This user ..."
}

7.11.3 Create user

POST/user

Description

This operation creates a user entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating a User, 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

Non-Mandatory Attributes	Rule
description	

Usage Samples

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

Request
POST {apiRoot}/user Content-Type: application/json {}
Response
201 {}

7.11.4 Delete user

DELETE/user/{id}

Description

This operation deletes a user entity.

Usage Samples

Here's an example of a request for deleting a User resource.

Request
DELETE {apiRoot}/user/42
Response
204

7.12 Operations on Usage Consumption Report

7.12.1 List usage consumption reports

GET/usageConsumptionReport?fields=...&{filtering}

Description

This operation list usage consumption report 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 UsageConsumptionReport resources.

Request

```
GET {apiRoot}/usageConsumptionReport
Accept: application/json
```

Response

```
200

[
{
   "bucket": [
      {}
   ],
   "description": "This  usage consumption report ...",
   "effectiveDate": "2019-05-13T00:00",
   "href": "https:/host:port/tmf-api/usageConsumptionReport/v1/usageConsumptionReport/8486",
   "id": "8486",
   "name": "a string ...",
   "relatedParty": {}
}
]
```

7.12.2 Retrieve usage consumption report

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

Description

This operation retrieves a usage consumption report 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 UsageConsumptionReport resource.

Request

```
GET {apiRoot}/usageConsumptionReport/8486
Accept: application/json
```

Response

```
200

{
   "bucket": [
      {}
   ],
   "description": "This  usage consumption report ...",
```

```
    "effectiveDate": "2019-05-13T00:00",
    "href": "https:/host:port/tmf-api/usageConsumptionReport/v1/usageConsumptionReport/8486",
    "id": "8486",
    "name": "a string ...",
    "relatedParty": {}
}
```

7.12.3 Delete usage consumption report

DELETE/usageConsumptionReport/{id}

Description

This operation deletes a usage consumption report entity.

Usage Samples

Here's an example of a request for deleting a UsageConsumptionReport resource.

Request
DELETE {apiRoot}/usageConsumptionReport/42
Response
204

7.13 Operations on Import Job

7.13.1 List import jobs

GET/importJob?fields=...&{filtering}

Description

This operation list import job 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 ImportJob resources.

Request
GET {apiRoot}/importJob Accept: application/json
Response
200

```
[
{
   "completionDate": "2019-05-13T00:00",
   "contentType": "a string ...",
   "creationDate": "2019-05-13T00:00",
   "errorLog": "a string ...",
   "href": "https:/host:port/tmf-api/importJob/v1/importJob/4640",
   "id": "4640",
   "path": "a string ...",
   "status": "a string ...",
   "url": "a string ..."
}
]
```

7.13.2 Retrieve import job

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

Description

This operation retrieves an import job 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 an ImportJob resource.

Request

```
GET {apiRoot}/importJob/4640
Accept: application/json
```

Response

```
200

{
   "completionDate": "2019-05-13T00:00",
   "contentType": "a string ...",
   "creationDate": "2019-05-13T00:00",
   "errorLog": "a string ...",
   "href": "https:/host:port/tmf-api/importJob/v1/importJob/4640",
   "id": "4640",
   "path": "a string ...",
   "status": "a string ...",
   "url": "a string ..."
}
```

7.13.3 Create import job

POST/importJob

Description

This operation creates an import job entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating an ImportJob, 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
url	

Non-Mandatory Attributes	Rule
completionDate	
contentType	
creationDate	
errorLog	
path	
status	

Usage Samples

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

Request

```
POST {apiRoot}/importJob
Content-Type: application/json

{
   "url": "a string ..."
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/importJob/v1/importJob/4640",
   "id": "4640",
   "url": "a string ..."
}
```

7.13.4 Delete import job

DELETE/importJob/{id}

Description

This operation deletes an import job entity.

Usage Samples

Here's an example of a request for deleting an ImportJob resource.

Request

```
DELETE {apiRoot}/importJob/42
```

Response

```
204
```

7.14 Operations on Export Job

7.14.1 List export jobs

GET/exportJob?fields=...&{filtering}

Description

This operation list export job 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 ExportJob resources.

Request

```
GET {apiRoot}/exportJob
Accept: application/json
```

Response

```
200

[
{
   "completionDate": "2019-05-13T00:00",
   "contentType": "a string ...",
   "creationDate": "2019-05-13T00:00",
   "errorLog": "a string ...",
   "href": "https:/host:port/tmf-api/exportJob/v1/exportJob/1721",
   "id": "1721",
   "path": "a string ...",
   "query": "a string ...",
   "status": "a string ...",
   "url": "a string ..."
}
]
```

7.14.2 Retrieve export job

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

Description

This operation retrieves an export job 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 an ExportJob resource.

Request

```
GET {apiRoot}/exportJob/1721
Accept: application/json
```

Response

```
200

{
   "completionDate": "2019-05-13T00:00",
   "contentType": "a string ...",
   "creationDate": "2019-05-13T00:00",
   "errorLog": "a string ...",
   "href": "https:/host:port/tmf-api/exportJob/v1/exportJob/1721",
   "id": "1721",
   "path": "a string ...",
   "query": "a string ...",
   "status": "a string ...",
   "url": "a string ..."
}
```

7.14.3 Create export job

POST/exportJob

Description

This operation creates an export job entity.

Mandatory and Non-Mandatory Attributes

The following tables provide the list of mandatory and non-mandatory attributes when creating an ExportJob, 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
url	

Non-Mandatory Attributes	Rule
completionDate	
contentType	
creationDate	
errorLog	
path	
query	
status	

Usage Samples

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

Request

```
POST {apiRoot}/exportJob
Content-Type: application/json

{
   "url": "a string ..."
}
```

Response

```
201

{
   "href": "https:/host:port/tmf-api/exportJob/v1/exportJob/1721",
   "id": "1721",
   "url": "a string ..."
}
```

7.14.4 Delete export job

DELETE/exportJob/{id}

Description

This operation deletes an export job entity.

Usage Samples

Here's an example of a request for deleting an ExportJob resource.

Request

```
DELETE {apiRoot}/exportJob/42
```

Response
204

8 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 reproduced below.

8.1 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}

8.2 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

8.3 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.

Bibliography

[b-TMF 914] TMForum 914 (2020), *IoT Service Management API Component Suite V4.0.0*.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	**Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities**
Series Z	Languages and general software aspects for telecommunication systems

Published in Switzerland
Geneva, 2024