

TM Forum Specification

Document Management API User Guide

TMF667
November 2021

Latest Update: TM Forum November 2021	Member Evaluation
Version 4.0.0	IPR Mode: RAND

NOTICE

Copyright © TM Forum 2019. All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to TM FORUM, except as needed for the purpose of developing any document or deliverable produced by a TM FORUM Collaboration Project Team (in which case the rules applicable to copyrights, as set forth in the [TM FORUM IPR Policy](#), must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by TM FORUM or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and TM FORUM DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Direct inquiries to the TM Forum office:

4 Century Drive, Suite 100
Parsippany, NJ 07054, USA
Tel No. +1 973 944 5100
Fax No. +1 973 998 7196
TM Forum Web Page: www.tmforum.org

Table of Contents

NOTICE	Error! Bookmark not defined.
Table of Contents.....	2
List of Tables	Error! Bookmark not defined.
Introduction	4
SAMPLE USE CASES.....	6
Support of polymorphism and extension patterns	7
RESOURCE MODEL.....	8
Managed Entity and Task Resource Models.....	8
FIRST resource	Error! Bookmark not defined.
Notification Resource Models	30
First Notification	Error! Bookmark not defined.
API OPERATIONS.....	35
VERB url.....	Error! Bookmark not defined.
API NOTIFICATIONS.....	54
Register listener	54
Unregister listener	56
Publish Event to listener	56
Acknowledgements	58
Release History	58
Contributors to Document.....	58

List of Tables

N/A

Introduction

The following document is the user guide of the REST API for Document and DocumentSpecification management. It includes the model definition as well as all available operations.

The Document API manages the meta-data related to a **Document**, such as the related parties (like author, reviewer, publisher), the creation date, version, and lifecycle status. The body of the Document itself is either held or referred to by the associated **Attachment**. Binary content might be encoded into the `Attachment.content` attribute, or (if by reference) referred to by the `Attachment.url` attribute – such as a `file:` or `https:` reference. Attachment is not a managed resource since the binary content is assumed to be resident in a Content Management System and not created by an API – only retrieved from or referred to.

Each Document type (such as PDF file, Word document, Image or Video) might be described by its own **DocumentSpecification**. This resource manages the collection of characteristics that detail the contained media type. For example, the DocumentSpecification for a “book” might have characteristics such as *number of pages*, *language*, an *ISBN*, *publisher* and *dimensions*.

The DocumentSpecification follows the EntitySpecification pattern, so you can describe both the nature of the characteristics themselves (such as their *name*, *cardinality*, *mandatory/optional*) as well as the nature of the *value* of those characteristics (whether the value of that characteristic is an *integer*, *string* or *boolean*, say – or conform to a particular regular-expression or sit within a linear *range*).

For example:

- A DocumentSpecification for “video” might have characteristics such as a “playlength” in integer seconds, a “width” and “height” in integer pixels, and “hasAudio” as a boolean.
- A Document – (managing the meta-data of the video) might add the video’s author (as a role played by a RelatedParty), it’s creation date, a lifecycle status to say that it is “published”, and reference the “video” DocumentSpecification for details of the further characteristics that can be discovered, relevant to this particular Document type.
- The associated Attachment might represent a link to an online YouTube video in the “url” attribute.

SAMPLE USE CASES

Reader will find example of use cases using Usage API in “Open Digital Business Scenarios and Use Cases” document.

Support of polymorphism and extension patterns

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

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

The `@type` attribute provides a way to represent the actual class type of an entity. For example, within a list of `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.

RESOURCE MODEL

Managed Entity and Task Resource Models

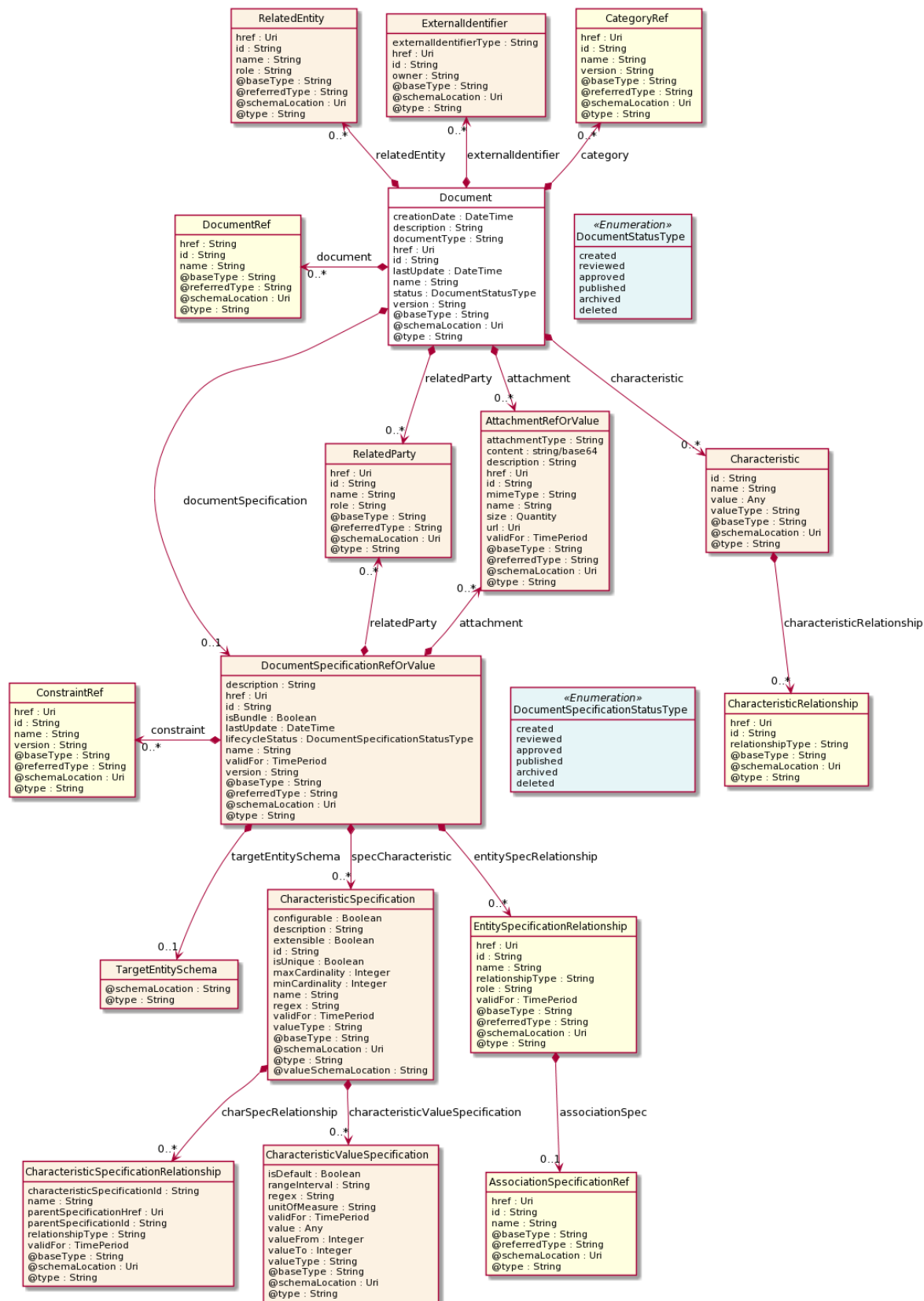
Document resource

A Document describes the meta-data of a Document - such as the name, creationDate and lifecycle status. The (typically binary) body of this document (such as a Word.doc, PDF, Video clip, or Image) will be held in the associated Attachment(s) either by Ref or Value. If by value - the binary content is held in the Attachment.content. If by reference, the Attachment.url might point to a (file:) or remote (http:) pointer to the Document media.

A Document may be associated with a DocumentSpecification, which will detail the characteristics of that type of Document (an Image may have a width, height and format; a Video may have a length and format).

A Document has a collection of RelatedParty's, for roles such as author, reviewer, publisher; and a lifecycle status to take the document through a simple set of production stages.

Resource model



Field descriptions

Document fields

href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
attachment	A list of attachment ref or values (AttachmentRefOrValue [*]). An attachment by value or by reference. An attachment complements the description of an element, for example through a document, a video, a picture.
category	A list of category references (CategoryRef [*]). The category for grouping recommendations.
characteristic	A list of characteristics (Characteristic [*]). Describes a given characteristic of an object or entity through a name/value pair.
creationDate	A date time (DateTime). The date and time the document was created. A date and time formatted in compliance with the ISO8601 standard must be used.
description	A string. free-text description of the document.
document	A list of document references (DocumentRef [*]).
documentSpecification	A document specification ref or value (DocumentSpecificationRefOrValue). A DocumentSpecificationRefOrValue where you can select between a DocumentSpecification (by Value) or a DocumentSpecificationRef (by Reference).
documentType	A string. Name of the document type.
externalIdentifier	A list of external identifiers (ExternalIdentifier [*]). An identification of an entity that is owned by or originates in a software system different from the current system, for example a ProductOrder handed off from a commerce platform into an order handling system. The structure identifies the system itself, the nature of the entity within the system (e.g. class name) and the unique ID of the entity within the system. It is anticipated that multiple external IDs can be held for a single entity, e.g. if the entity passed through multiple systems on the way to the current system. In this case the consumer is expected to sequence the IDs in the array in reverse order of provenance, i.e. most recent system first in the list.
lastUpdate	A date time (DateTime). The date and time the document was last modified. A date and time formatted in compliance with the ISO8601 standard must be used.

name	A string. A string used to give a name to the document.
relatedEntity	A list of related entities (RelatedEntity [*]). A reference to an entity, where the type of the entity is not known in advance.
relatedParty	A list of related parties (RelatedParty [*]). Related Entity reference. A related party defines party or party role linked to a specific entity.
status	A document status type (DocumentStatusType). The life cycle state of the document.
version	A string. A particular form or variety of an artefact that is different from others or from the original. The form represents differences in properties that characterize an artefact, that are not enough to warrant creating a new artefact.

AttachmentRefOrValue sub-resource

An attachment by value or by reference. An attachment complements the description of an element, for example through a document, a video, a picture.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
description	A string. A narrative text describing the content of the attachment.
href	An uri (Uri). URI for this Attachment.
id	A string. Unique identifier for this particular attachment.
url	An uri (Uri). Remote reference to the content if web-addressable.
name	A string. The name of the attachment.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
attachmentType	A string. Attachment type such as video, picture.
content	A base 6 4 (Base64). The actual contents of the attachment object, if embedded, encoded as base64.
contentType	A string. Attachment mime type such as extension file for video, picture and document.
size	A quantity (Quantity). The size of the attachment.
validFor	A time period. The period of time for which the attachment is valid.

Characteristic sub-resource

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

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

CharacteristicRelationship sub-resource

Another Characteristic that is related to the current Characteristic;.

href	An uri (Uri). Hyperlink reference.
id	A string. Unique identifier of the characteristic.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
relationshipType	A string. The type of relationship.

CharacteristicSpecification sub-resource

This class defines a characteristic specification.

@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 target Characteristic is configurable.
description	A string. A narrative that explains the CharacteristicSpecification.

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 a resource.
id	A string. Unique ID for the characteristic.
isUnique	A boolean. Specifies if the value of this characteristic is unique across all entities instantiated from the specification that uses this characteristic. For example, consider a ProductSpecification for a set-top box, with a CharacteristicSpecification cardID. Each set-top box must have a different value for cardID, so this isUnique attribute would be set to true for the characteristic.
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.
valueType	A string. A kind of value that the characteristic can take on, such as numeric, text and so forth.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
charSpecRelationship	A list of characteristic specification relationships (CharacteristicSpecificationRelationship [*]). An aggregation, migration, substitution, dependency or exclusivity relationship between/among Specification Characteristics.
characteristicValueSpecification	A list of characteristic value specifications (CharacteristicValueSpecification [*]). A CharacteristicValueSpecification object is used to define a set of attributes, each of which can be assigned to a corresponding set of attributes in a CharacteristicSpecification object. The values of the attributes in the

CharacteristicValueSpecification object describe the values of the attributes that a corresponding Characteristic object can take on.

validFor A time period. The period of time for which a characteristic is applicable.

CharacteristicSpecificationRelationship sub-resource

An aggregation, migration, substitution, dependency or exclusivity relationship between/among Characteristic specifications. The specification characteristic is embedded within the specification whose ID and href are in this entity, and identified by its ID.

@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
characteristicSpecificationId	A string. Unique identifier of the characteristic within the specification.
name	A string. Name of the target characteristic within the specification.
parentSpecificationHref	An uri (Uri). Hyperlink reference to the parent specification containing the target characteristic.
parentSpecificationId	A string. Unique identifier of the parent specification containing the target characteristic.
relationshipType	A string. Type of relationship such as aggregation, migration, substitution, dependency, exclusivity.
validFor	A time period. The period for which the object is valid.

CharacteristicValueSpecification sub-resource

specification of a value (number or text or an object) that can be assigned to a Characteristic.

@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
isDefault	A boolean. If true, the 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 value can take on, such as numeric, text and so forth.

DocumentSpecificationRefOrValue sub-resource

A DocumentSpecificationRefOrValue where you can select between a DocumentSpecification (by Value) or a DocumentSpecificationRef (by Reference).

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. Name given to the specification.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
version	A string. specification version.
description	A string. Description of the specification.
isBundle	A boolean. isBundle determines whether specification represents a single specification (false), or a bundle of specifications (true).

lastUpdate	A date time (DateTime). Date and time of the last update of the specification.
lifecycleStatus	A document specification status type (DocumentSpecificationStatusType). The current lifecycle status of this DocumentSpecification.
attachment	A list of attachment ref or values (AttachmentRefOrValue [*]). Attachments that may be of relevance to this specification, such as picture, document, media.
constraint	A list of constraint references (ConstraintRef [*]). This is a list of constraint references applied to this specification.
entitySpecRelationship	A list of entity specification relationships (EntitySpecificationRelationship [*]). Relationship to another specification.
relatedParty	A list of related parties (RelatedParty [*]). Parties who manage or otherwise have an interest in this specification.
specCharacteristic	A list of characteristic specifications (CharacteristicSpecification [*]). List of characteristics that the entity can take.
targetEntitySchema	A target entity schema (TargetEntitySchema). Pointer to a schema that defines the target entity.
validFor	A time period. The period for which this REST resource is valid.

EntitySpecificationRelationship sub-resource

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

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. Name of the related entity.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
associationSpec	An association specification reference (AssociationSpecificationRef). A specification for an association used by this relationship.
relationshipType	A string. Type of relationship such as migration, substitution, dependency, exclusivity.

role	A string. The association role for this entity specification.
validFor	A time period. The period for which the entitySpecRelationship is valid.

ExternalIdentifier sub-resource

An identification of an entity that is owned by or originates in a software system different from the current system, for example a ProductOrder handed off from a commerce platform into an order handling system. The structure identifies the system itself, the nature of the entity within the system (e.g. class name) and the unique ID of the entity within the system. It is anticipated that multiple external IDs can be held for a single entity, e.g. if the entity passed through multiple systems on the way to the current system. In this case the consumer is expected to sequence the IDs in the array in reverse order of provenance, i.e. most recent system first in the list.

href	An uri (Uri). Hyperlink reference.
id	A string. identification of the entity within the external system.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
externalIdentifierType	A string. Type of the identification, typically would be the type of the entity within the external system.
owner	A string. Name of the external system that owns the entity.

Quantity sub-resource

An amount in a given unit.

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

RelatedEntity sub-resource

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.
name	A string. Name of the related entity.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.

@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
role	A string. The role of an entity.

RelatedParty sub-resource

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

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

TargetEntitySchema sub-resource

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

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

AssociationSpecificationRef relationship

reference to an AssociationSpecification object.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. Name of the related entity.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

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

CategoryRef relationship

The category for grouping recommendations.

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

name A string. Name of the related entity.

href An uri (Uri). Hyperlink reference.

id A string. unique identifier.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

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

version A string. Category version.

ConstraintRef relationship

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

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

name A string. Name of the related entity.

href An uri (Uri). Hyperlink reference.

id A string. unique identifier.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

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

version A string. constraint version.

DocumentRef relationship

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

Json representation sample

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

```
{
  "id": "9555-3c9c417feead",
  "href": "https://serverRoot/tmf-api/document/v4/document/9555-3c9c417feead",
  "creationDate": "2008-09-14T10:00:00.000Z",
  "name": "Designing Web APIs: Building APIs That Developers Love",
  "description": "Designing an API is complicated to begin with, but evolving your API design over time makes the process even more difficult. There are several books on the topic, but ...",
  "documentType": "paperback",
  "lastUpdate": "2021-11-05T10:15:26.007Z",
  "version": "1",
  "status": "published",
  "attachment": [
    {
      "id": "1492026921",
      "attachmentType": "url",
      "name": "Link to paperback",
      "url": "https://www.amazon.co.uk/Designing-Web-APIs-Building-Developers/dp/1492026921",
      "validFor": {
        "endDateTime": "2008-09-14T10:00:00.000Z",
        "startDateTime": "2024-11-05T10:15:26.007Z"
      }
    }
  ],
  "category": [
    {
      "id": "269422",
      "name": "Computing & Internet > Web Development > Web Design"
    }
  ],
  "characteristic": [
    {
      "name": "pages",
      "valueType": "integer",
      "value": 200
    }
  ],
}
```

```

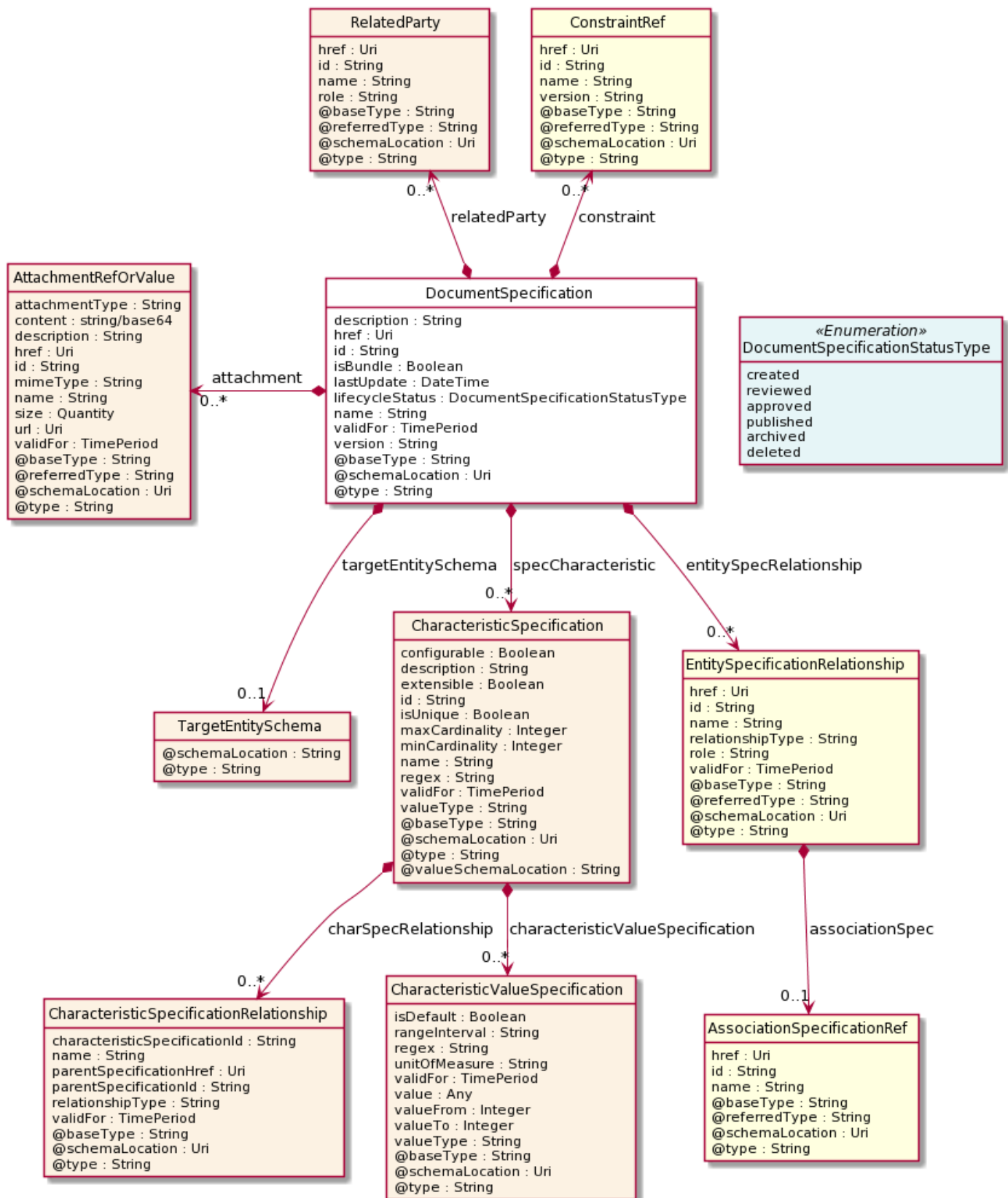
{
  "name": "language",
  "valueType": "string",
  "value": "en"
},
{
  "name": "isbn-10",
  "valueType": "string",
  "value": "9781492026921"
}
],
"documentSpecification": {
  "id": "a65d-a08c744f2fe7"
},
"relatedParty": [
  {
    "id": "8e49-d1229f59f42d",
    "href": "https://serverRoot/tmf-api/party/8e49-d1229f59f42d",
    "name": "Brenda Jin",
    "role": "author",
    "@referredType": "Individual"
  },
  {
    "id": "8158-3f3f74ebfc25",
    "href": "https://serverRoot/tmf-api/party/8158-3f3f74ebfc25",
    "name": "O'Reilly",
    "role": "publisher",
    "@referredType": "Organisation"
  }
],
"@type": "Document"
}

```

Document Specification resource

A DocumentSpecification provides the characteristics and constraints that describe a particular type of Document. It follows the EntitySpecification pattern, so allows you to define the characteristics expected, as well as the nature of the values of those characteristics (eg: integer range [valueFrom/valueTo], regular expression or unitOfMeasure).

Resource model



Field descriptions

DocumentSpecification fields

description	A string. Description of the specification.
isBundle	A boolean. isBundle determines whether specification represents a single specification (false), or a bundle of specifications (true).
lastUpdate	A date time (DateTime). Date and time of the last update of the specification.
lifecycleStatus	A document specification status type (DocumentSpecificationStatusType). The current lifecycle status of this DocumentSpecification.
name	A string. The name of the DocumentSpecification.
version	A string. The version of the DocumentSpecification.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
lifecycleStatus	A document specification status type (DocumentSpecificationStatusType). The current lifecycle status of this DocumentSpecification.
name	A string. The name of the DocumentSpecification.
version	A string. The version of the DocumentSpecification.
attachment	A list of attachment ref or values (AttachmentRefOrValue [*]). Attachments that may be of relevance to this specification, such as picture, document, media.
constraint	A list of constraint references (ConstraintRef [*]). This is a list of constraint references applied to this specification.
entitySpecRelationship	A list of entity specification relationships (EntitySpecificationRelationship [*]). Relationship to another specification.
relatedParty	A list of related parties (RelatedParty [*]). Parties who manage or otherwise have an interest in this specification.
specCharacteristic	A list of characteristic specifications (CharacteristicSpecification [*]). List of characteristics that the entity can take.

targetEntitySchema	A target entity schema (TargetEntitySchema). Pointer to a schema that defines the target entity.
validFor	A time period. The period for which this REST resource is valid.

AttachmentRefOrValue sub-resource

An attachment by value or by reference. An attachment complements the description of an element, for example through a document, a video, a picture.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
description	A string. A narrative text describing the content of the attachment.
href	An uri (Uri). URI for this Attachment.
id	A string. Unique identifier for this particular attachment.
url	An uri (Uri). Remote reference to the content if web-addressable.
name	A string. The name of the attachment.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
attachmentType	A string. Attachment type such as video, picture.
content	A base 6 4 (Base64). The actual contents of the attachment object, if embedded, encoded as base64.
contentType	A string. Attachment mime type such as extension file for video, picture and document.
size	A quantity (Quantity). The size of the attachment.
validFor	A time period. The period of time for which the attachment is valid.

CharacteristicSpecification sub-resource

This class defines a characteristic specification.

@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 target Characteristic is configurable.

description	A string. A narrative that explains the CharacteristicSpecification.
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 a resource.
id	A string. Unique ID for the characteristic.
isUnique	A boolean. Specifies if the value of this characteristic is unique across all entities instantiated from the specification that uses this characteristic. For example, consider a ProductSpecification for a set-top box, with a CharacteristicSpecification cardID. Each set-top box must have a different value for cardID, so this isUnique attribute would be set to true for the characteristic.
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.
valueType	A string. A kind of value that the characteristic can take on, such as numeric, text and so forth.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
charSpecRelationship	A list of characteristic specification relationships (CharacteristicSpecificationRelationship [*]). An aggregation, migration, substitution, dependency or exclusivity relationship between/among Specification Characteristics.
characteristicValueSpecification	A list of characteristic value specifications (CharacteristicValueSpecification [*]). A CharacteristicValueSpecification object is used to define a set of attributes, each of which can be assigned to a corresponding set of attributes in a CharacteristicSpecification object. The values of the attributes in the

CharacteristicValueSpecification object describe the values of the attributes that a corresponding Characteristic object can take on.

validFor A time period. The period of time for which a characteristic is applicable.

CharacteristicSpecificationRelationship sub-resource

An aggregation, migration, substitution, dependency or exclusivity relationship between/among Characteristic specifications. The specification characteristic is embedded within the specification whose ID and href are in this entity, and identified by its ID.

@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
characteristicSpecificationId	A string. Unique identifier of the characteristic within the specification.
name	A string. Name of the target characteristic within the specification.
parentSpecificationHref	An uri (Uri). Hyperlink reference to the parent specification containing the target characteristic.
parentSpecificationId	A string. Unique identifier of the parent specification containing the target characteristic.
relationshipType	A string. Type of relationship such as aggregation, migration, substitution, dependency, exclusivity.
validFor	A time period. The period for which the object is valid.

CharacteristicValueSpecification sub-resource

specification of a value (number or text or an object) that can be assigned to a Characteristic.

@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
isDefault	A boolean. If true, the 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 value can take on, such as numeric, text and so forth.

EntitySpecificationRelationship sub-resource

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

@referredType	A string. The actual type of the target instance when needed for disambiguation.
name	A string. Name of the related entity.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
associationSpec	An association specification reference (AssociationSpecificationRef). A specification for an association used by this relationship.
relationshipType	A string. Type of relationship such as migration, substitution, dependency, exclusivity.
role	A string. The association role for this entity specification.

validFor A time period. The period for which the entitySpecRelationship is valid.

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.

name A string. Name of the related entity.

href An uri (Uri). Hyperlink reference.

id A string. unique identifier.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

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

role A string. Role played by the related party.

TargetEntitySchema sub-resource

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

@schemaLocation A string. This field provides a link to the schema describing the target entity.

@type A string. Class type of the target entity.

AssociationSpecificationRef relationship

reference to an AssociationSpecification object.

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

name A string. Name of the related entity.

href An uri (Uri). Hyperlink reference.

id A string. unique identifier.

@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.

ConstraintRef relationship

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

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

Json representation sample

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

```
{
  "id": "9b12-789948415bd7",
  "href": "https://serverRoot/tmf-api/document/v4/documentSpecification/9b12-789948415bd7",
  "name": "paperback",
  "description": "Specification for a Paperback Book",
  "isBundle": false,
  "lifecycleStatus": "published",
  "lastUpdate": "2021-11-05T11:03:57.693Z",
  "version": "1.0.0",
  "specCharacteristic": [
    {
      "name": "pages",
      "description": "The number of pages in this paperback book",
      "configurable": true,
      "valueType": "integer",
      "extensible": false,
      "isUnique": true,
      "maxCardinality": 1,
      "minCardinality": 1,
      "characteristicValueSpecification": [
        {

```

```

        "unitOfMeasure": "pages",
        "valueFrom": 1
    }
],
},
{
    "name": "language",
    "description": "The 2-character ISO 639-1 language code that this book is published in",
    "configurable": true,
    "valueType": "string",
    "characteristicValueSpecification": [
        {
            "regex": "[a-z]{2}"
        }
    ],
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1
},
{
    "name": "isbn-10",
    "description": "The 10-digit ISO 2108 International Standard Book Number (ISBN) of this book",
    "configurable": true,
    "valueType": "string",
    "characteristicValueSpecification": [
        {
            "regex": "[0-9]{10}"
        }
    ],
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1
}
]
}

```

Notification Resource Models

7 notifications are defined for this API

Notifications related to Document:

- DocumentCreateEvent
- DocumentChangeEvent
- DocumentBatchEvent
- DocumentDeleteEvent

Notifications related to DocumentSpecification:

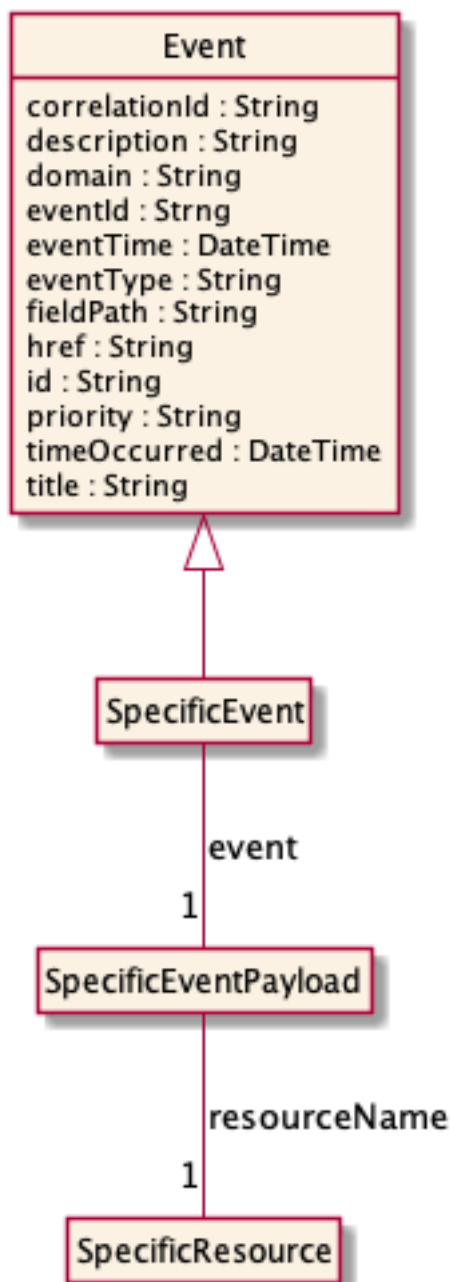
- DocumentSpecificationCreateEvent

- DocumentSpecificationAttributeValueChangeEvent
- DocumentSpecificationDeleteEvent

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

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

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



Document Create Event

Notification DocumentCreateEvent case for resource Document

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"DocumentCreateEvent",
  "event": {
    "document" :
      {-- SEE Document RESOURCE SAMPLE --}
  }
}
```

Document Change Event

Notification DocumentChangeEvent case for resource Document

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"DocumentChangeEvent",
  "event": {
    "document" :
      {-- SEE Document RESOURCE SAMPLE --}
  }
}
```

Document Batch Event

Notification DocumentBatchEvent case for resource Document

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"DocumentBatchEvent",
  "event": {
    "document" :
      {-- SEE Document RESOURCE SAMPLE --}
  }
}
```



```
}
```

Document Delete Event

Notification DocumentDeleteEvent case for resource Document

Json representation sample

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

```
{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "DocumentDeleteEvent",
  "event": {
    "document":
      {-- SEE Document RESOURCE SAMPLE --}
  }
}
```

Document Specification Create Event

Notification DocumentSpecificationCreateEvent case for resource DocumentSpecification

Json representation sample

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

```
{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "DocumentSpecificationCreateEvent",
  "event": {
    "documentSpecification":
      {-- SEE DocumentSpecification RESOURCE SAMPLE --}
  }
}
```

Document Specification Attribute Value Change Event

Notification DocumentSpecificationAttributeValueChangeEvent case for resource DocumentSpecification

Json representation sample

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

```
{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "DocumentSpecificationAttributeValueChangeEvent",
  "event": {
    "documentSpecification":
      {-- SEE DocumentSpecification RESOURCE SAMPLE --}
  }
}
```

Document Specification Delete Event

Notification DocumentSpecificationDeleteEvent case for resource DocumentSpecification

Json representation sample

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

```
{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "DocumentSpecificationDeleteEvent",
  "event": {
    "documentSpecification":
      {-- SEE DocumentSpecification RESOURCE SAMPLE --}
  }
}
```

API OPERATIONS

Remember the following Uniform Contract:

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

Operations on Document

List documents

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

Description

This operation list document 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

Retrieving previous Document details for the documentType of 'paperback'

Request
GET /tmf-api/document/v4/document?fields=id,href,lifecycleState,name,creationDate&documentType=paperback Accept: application/json
Response
200 [{ " id": "9555-3c9c417feead", " href": "https://serverRoot/tmf-api/document/v4/document/9555-3c9c417feead", " creationDate": "2008-09-14T10:00:00.000Z", " name": "Designing Web APIs: Building APIs That Developers Love", " status": "published", " @type": "Document" }, { " id": "b64e-211208048e84", " href": "https://serverRoot/tmf-api/document/v4/document/b64e-211208048e84", " creationDate": "2013-09-30T10:00:00.000Z", " name": "RESTful Web APIs: Services for a Changing World", " status": "published", " @type": "Document" }]

Retrieve document

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

Description

This operation retrieves a document 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

A request for a previous Document. The criteria is the Document id 9555-3c9c417feead

Request
GET /tmf-api/document/v4/document/9555-3c9c417feead?fields=id,lifecycleState,name Accept: application/json
Response
200 <pre>{ "id": "9555-3c9c417feead", "name": "Designing Web APIs: Building APIs That Developers Love", "status": "published", "@type": "Document" }</pre>

Create document

POST /document

Description

This operation creates a document entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a Document, 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
@baseType	

@schemaLocation	
@type	
attachment	
category	
characteristic	
creationDate	
description	
document	
documentSpecification	
documentType	
externalIdentifier	
lastUpdate	
relatedEntity	
relatedParty	
status	
version	

Usage Samples

Create a Document for a paperback book - the resulting resource ID is 9555-3c9c417feead

Request
POST /tmf-api/document/v4/document Content-Type: application/json <pre>{ "creationDate": "2008-09-14T10:00:00.000Z", "name": "Designing Web APIs: Building APIs That Developers Love", "description": "Designing an API is complicated to begin with, but evolving your API design over time makes the process even more difficult. There are several books on the topic, but ...", "documentType": "paperback", "lastUpdate": "2021-11-05T10:15:26.007Z", "version": "1", "status": "published", "attachment": [{ "id": "1492026921", "attachmentType": "url", "name": "Link to paperback", "url": "https://www.amazon.co.uk/Designing-Web-APIs-Building-Developers/dp/1492026921", "validFor": { "endDateTime": "2008-09-14T10:00:00.000Z", "startDateTime": "2024-11-05T10:15:26.007Z" } }], "category": [{ "id": "269422" }] }</pre>

```

    }
  ],
  "characteristic": [
    {
      "name": "pages",
      "valueType": "integer",
      "value": 200
    },
    {
      "name": "language",
      "valueType": "string",
      "value": "en"
    },
    {
      "name": "isbn-10",
      "valueType": "string",
      "value": "9781492026921"
    }
  ],
  "documentSpecification": {
    "id": "9b12-789948415bd7"
  },
  "relatedParty": [
    {
      "id": "8e49-d1229f59f42d",
      "href": "https://serverRoot/tmf-api/party/8e49-d1229f59f42d",
      "name": "Brenda Jin",
      "role": "author",
      "@referredType": "Individual"
    },
    {
      "id": "8158-3f3f74ebfc25",
      "href": "https://serverRoot/tmf-api/party/8158-3f3f74ebfc25",
      "name": "O'Reilly",
      "role": "publisher",
      "@referredType": "Organisation"
    }
  ],
  "@type": "Document"
}

```

Response

201

```

{
  "id": "9555-3c9c417feead",
  "href": "https://serverRoot/tmf-api/document/v4/document/9555-3c9c417feead",
  "creationDate": "2008-09-14T10:00:00.000Z",
  "name": "Designing Web APIs: Building APIs That Developers Love",
  "description": "Designing an API is complicated to begin with, but evolving your API design over time makes the process even more difficult. There are several books on the topic, but ..."
}

```

```
"documentType": "paperback",
"lastUpdate": "2021-11-05T10:15:26.007Z",
"version": "1",
"status": "published",
"attachment": [
  {
    "id": "1492026921",
    "attachmentType": "url",
    "name": "Link to paperback",
    "url": "https://www.amazon.co.uk/Designing-Web-APIs-Building-Developers/dp/1492026921",
    "validFor": {
      "endTime": "2008-09-14T10:00:00.000Z",
      "startTime": "2024-11-05T10:15:26.007Z"
    }
  }
],
"category": [
  {
    "id": "269422",
    "name": "Computing & Internet > Web Development > Web Design"
  }
],
"characteristic": [
  {
    "name": "pages",
    "valueType": "integer",
    "value": 200
  },
  {
    "name": "language",
    "valueType": "string",
    "value": "en"
  },
  {
    "name": "isbn-10",
    "valueType": "string",
    "value": "9781492026921"
  }
],
"documentSpecification": {
  "id": "92225e29-b34e-4d61-a65d-a08c744f2fe7",
  "name": "paperback"
},
"relatedParty": [
  {
    "id": "8e49-d1229f59f42d",
    "href": "https://serverRoot/tmf-api/party/8e49-d1229f59f42d",
    "name": "Brenda Jin",
    "role": "author",
    "@referredType": "Individual"
  },
  {
    "id": "8158-3f3f74ebfc25",
    "href": "https://serverRoot/tmf-api/party/8158-3f3f74ebfc25",
```



```

    "name": "O'Reilly",
    "role": "publisher",
    "@referredType": "Organisation"
  },
  "@type": "Document"
}

```

Patch document

PATCH /document/{id}

Description

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

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

Patchable and Non Patchable Attributes

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

Patchable Attributes	Rule
attachment	
category	
characteristic	
creationDate	
description	
document	
documentSpecification	
documentType	
externalIdentifier	
lastUpdate	
name	
relatedEntity	
relatedParty	
status	
version	

Non Patchable Attributes	Rule
id	
href	

@type	
@baseType	
@schemaLocation	

Usage Samples

Changing the lifecycle state of a previous Document, changing the Document with id 903a-79a12c91b92a

Request
PATCH /tmf-api/document/v4/document/903a-79a12c91b92a Content-Type: application/merge-patch+json <pre>{ "lastUpdate": "2021-11-20T08:00:00.000Z", "status": "archived" }</pre>
Response
200 <pre>{ "id": "903a-79a12c91b92a", "href": "https://serverRoot/tmf-api/document/v4/document/903a-79a12c91b92a", "creationDate": "2008-09-14T10:00:00.000Z", "name": "Designing Web APIs: Building APIs That Developers Love", "description": "Designing an API is complicated to begin with, but evolving your API design over time makes the process even more difficult. There are several books on the topic, but ...", "documentType": "paperback", "lastUpdate": "2021-11-20T08:00:00.000Z", "version": "1", "status": "archived", "attachment": [{ "id": "1492026921", "attachmentType": "url", "name": "Link to paperback", "url": "https://www.amazon.co.uk/Designing-Web-APIs-Building-Developers/dp/1492026921", "validFor": { "endDateTime": "2008-09-14T10:00:00.000Z", "startDateTime": "2024-11-05T10:15:26.007Z" } }], "category": [{ "id": "269422", "name": "Computing & Internet > Web Development > Web Design" }] }</pre>

```

    }
  ],
  "characteristic": [
    {
      "name": "pages",
      "valueType": "integer",
      "value": 200
    },
    {
      "name": "language",
      "valueType": "string",
      "value": "en"
    },
    {
      "name": "isbn-10",
      "valueType": "string",
      "value": "9781492026921"
    }
  ],
  "documentSpecification": {
    "id": "9b12-789948415bd7",
    "name": "paperback"
  },
  "relatedParty": [
    {
      "id": "8e49-d1229f59f42d",
      "href": "https://serverRoot/tmf-api/party/8e49-d1229f59f42d",
      "name": "Brenda Jin",
      "role": "author",
      "@referredType": "Individual"
    },
    {
      "id": "8158-3f3f74ebfc25",
      "href": "https://serverRoot/tmf-api/party/8158-3f3f74ebfc25",
      "name": "O'Reilly",
      "role": "publisher",
      "@referredType": "Organisation"
    }
  ],
  "@type": "Document"
}

```

Delete document

DELETE /document/{id}

Description

This operation deletes a document entity.

Usage Samples

A request to delete an existing Document

Request
DELETE /tmf-api/document/v4/document/903a-79a12c91b92a
Response
204

Operations on Document Specification

List document specifications

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

Description

This operation list document 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

Retrieving details of DocumentSpecifications which are published

Request
GET /tmf-api/document/v4/documentSpecification?fields=id,description,version&lifecycleState=published Accept: application/json
Response
200 [{ "id": "9b12-789948415bd7", "description": "Specification for a Paperback Book",

```

    "version": "1.0.0"
  },
  {
    "id": "8bf6-39c5d6e79c9f",
    "description": "Specification for a Video Clip",
    "version": "1.5.0"
  },
  {
    "id": "bea7-548abd8d2d76",
    "description": "Specification for a PDF Document",
    "version": "2.1.0"
  }
]

```

Retrieve document specification

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

Description

This operation retrieves a document 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

A request for a specific DocumentSpecification with an id 91b2-43106cbc8191

Request
GET /tmf-api/document/v4/documentSpecification/91b2-43106cbc8191 Accept: application/json
Response
200 <pre> { "id": "91b2-43106cbc8191", "href": "https://serverRoot/tmf-api/document/v4/documentSpecification/91b2-43106cbc8191", "name": "paperback", "description": "Specification for a Paperback Book", "isBundle": false, "lifecycleStatus": "published", "lastUpdate": "2021-11-05T11:03:57.693Z", "version": "1.0.0", </pre>

```

"specCharacteristic": [
  {
    "name": "pages",
    "description": "The number of pages in this paperback book",
    "configurable": true,
    "valueType": "integer",
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1,
    "characteristicValueSpecification": [
      {
        "unitOfMeasure": "pages",
        "valueFrom": 1
      }
    ]
  },
  {
    "name": "language",
    "description": "The 2-character ISO 639-1 language code that this book is published in",
    "configurable": true,
    "valueType": "string",
    "characteristicValueSpecification": [
      {
        "regex": "^[a-z]{2}"
      }
    ],
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1
  },
  {
    "name": "isbn-10",
    "description": "The 10-digit ISO 2108 International Standard Book Number (ISBN) of this book",
    "configurable": true,
    "valueType": "string",
    "characteristicValueSpecification": [
      {
        "regex": "^[0-9]{10}"
      }
    ],
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1
  }
]
}

```

Create document specification

POST /documentSpecification

Description

This operation creates a document specification entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a DocumentSpecification, 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	
@baseType	
@schemaLocation	
@type	
lifecycleStatus	
version	
attachment	
constraint	
entitySpecRelationship	
relatedParty	
specCharacteristic	
targetEntitySchema	
validFor	

Usage Samples

Create a DocumentSpecification for Paperback book - the resulting resource id is 9b12-789948415bd7

Request
POST /tmf-api/document/v4/documentSpecification Content-Type: application/json <pre>{ "name": "paperback", "description": "Specification for a Paperback Book", "isBundle": false, "lifecycleStatus": "published", "lastUpdate": "2021-11-05T11:03:57.693Z",</pre>

```
"version": "1.0.0",
"specCharacteristic": [
  {
    "name": "pages",
    "description": "The number of pages in this paperback book",
    "configurable": true,
    "valueType": "integer",
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1,
    "characteristicValueSpecification": [
      {
        "unitOfMeasure": "pages",
        "valueFrom": 1
      }
    ]
  },
  {
    "name": "language",
    "description": "The 2-character ISO 639-1 language code that this book is published in",
    "configurable": true,
    "valueType": "string",
    "characteristicValueSpecification": [
      {
        "regex": "^[a-z]{2}"
      }
    ],
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1
  },
  {
    "name": "isbn-10",
    "description": "The 10-digit ISO 2108 International Standard Book Number (ISBN) of this book",
    "configurable": true,
    "valueType": "string",
    "characteristicValueSpecification": [
      {
        "regex": "^[0-9]{10}"
      }
    ],
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1
  }
]
}
```


Response

201

```

{
  "id": "9b12-789948415bd7",
  "href": "https://serverRoot/tmf-api/document/v4/documentSpecification/9b12-789948415bd7",
  "name": "paperback",
  "description": "Specification for a Paperback Book",
  "isBundle": false,
  "lifecycleStatus": "published",
  "lastUpdate": "2021-11-05T11:03:57.693Z",
  "version": "1.0.0",
  "specCharacteristic": [
    {
      "name": "pages",
      "description": "The number of pages in this paperback book",
      "configurable": true,
      "valueType": "integer",
      "extensible": false,
      "isUnique": true,
      "maxCardinality": 1,
      "minCardinality": 1,
      "characteristicValueSpecification": [
        {
          "unitOfMeasure": "pages",
          "valueFrom": 1
        }
      ]
    },
    {
      "name": "language",
      "description": "The 2-character ISO 639-1 language code that this book is published in",
      "configurable": true,
      "valueType": "string",
      "characteristicValueSpecification": [
        {
          "regex": "^[a-z]{2}"
        }
      ],
      "extensible": false,
      "isUnique": true,
      "maxCardinality": 1,
      "minCardinality": 1
    },
    {
      "name": "isbn-10",
      "description": "The 10-digit ISO 2108 International Standard Book Number (ISBN) of this book",
      "configurable": true,
      "valueType": "string",
      "characteristicValueSpecification": [
        {
          "regex": "^[0-9]{10}"
        }
      ]
    }
  ]
}

```

```

    }
  },
  "extensible": false,
  "isUnique": true,
  "maxCardinality": 1,
  "minCardinality": 1
}
]
}

```

Patch document specification

PATCH /documentSpecification/{id}

Description

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

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

Patchable and Non Patchable Attributes

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

Patchable Attributes	Rule
description	
isBundle	
lastUpdate	
lifecycleStatus	
name	
version	
lifecycleStatus	
name	
version	
attachment	
constraint	
entitySpecRelationship	
relatedParty	
specCharacteristic	
targetEntitySchema	
validFor	

Non Patchable Attributes	Rule
id	
href	
@type	
@baseType	
@schemaLocation	

Usage Samples

Changing the lifecycle status of an existing DocumentSpecification with an id of 903a-79a12c91b92a

Request
PATCH /tmf-api/document/v4/documentSpecification/903a-79a12c91b92a Content-Type: application/merge-patch+json <pre>{ "lifecycleStatus": "archived", "lastUpdate": "2021-11-20T08:00:00.000Z" }</pre>
Response
200 <pre>{ "id": "903a-79a12c91b92a", "href": "https://serverRoot/tmf-api/document/v4/documentSpecification/903a-79a12c91b92a", "name": "paperback", "description": "Specification for a Paperback Book", "isBundle": false, "lifecycleStatus": "archived", "lastUpdate": "2021-11-20T08:00:00.000Z", "version": "1.0.0", "specCharacteristic": [{ "name": "pages", "description": "The number of pages in this paperback book", "configurable": true, "valueType": "integer", "extensible": false, "isUnique": true, "maxCardinality": 1, "minCardinality": 1, "characteristicValueSpecification": [{ "unitOfMeasure": "pages", "valueFrom": 1 }] }] }</pre>

```

    ]
  },
  {
    "name": "language",
    "description": "The 2-character ISO 639-1 language code that this book is published in",
    "configurable": true,
    "valueType": "string",
    "characteristicValueSpecification": [
      {
        "regex": "^[a-z]{2}"
      }
    ],
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1
  },
  {
    "name": "isbn-10",
    "description": "The 10-digit ISO 2108 International Standard Book Number (ISBN) of this book",
    "configurable": true,
    "valueType": "string",
    "characteristicValueSpecification": [
      {
        "regex": "^[0-9]{10}"
      }
    ],
    "extensible": false,
    "isUnique": true,
    "maxCardinality": 1,
    "minCardinality": 1
  }
]
}

```

Delete document specification

DELETE /documentSpecification/{id}

Description

This operation deletes a document specification entity.

Usage Samples

A request to delete an existing DocumentSpecification with an id of 8b64-0cfd1a90c9ed

Request

DELETE /tmf-api/document/v4/documentSpecification/8b64-0cfd1a90c9ed
Response
204

API NOTIFICATIONS

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

It is assumed that the Pub/Sub uses the Register and UnRegister mechanisms described in the REST Guidelines reproduced below.

Register listener

POST /hub

Description

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

Behavior

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

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

Usage Samples

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

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

Unregister listener

DELETE /hub/{id}

Description

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

Behavior

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

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

Usage Samples

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

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

Publish Event to listener

POST /client/listener

Description

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

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

Behavior

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

Usage Samples

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

Request
<pre>POST /client/listener Accept: application/json { "event": { EVENT BODY }, "eventType": "EVENT_TYPE" }</pre>
Response
201

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

Acknowledgements

Release History

Release Number	Date	Release led by:	Description
Release 1.0	04/15/2017	Pierre Gauthier TM Forum pgauthier@tmforum.org Mariano Belaunde Orange Labs	First Release of the Document.
Release 2.0	11/06/2018	Mariano Belaunde Orange Labs	Alignment with Guidelines 3.0
V4.0.0	11/09/2021	Stephen Harrop Vodafone Group	Major update to accommodate patterns from TMF-API Design Guidelines v4.0 and schema library-based generation.

Contributors to Document

--	--