

TM Forum Specifications

Test Data Management API User Guide

TMF706

Team Approved Date: 29-Jul-2020

Release Status: Production	Approval Status: TM Forum Approved
Version 4.0.0	IPR Mode: RAND

NOTICE

Copyright © TM Forum 2020. 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.

TM FORUM invites any TM FORUM Member or any other party that believes it has patent claims that would necessarily be infringed by implementations of this TM Forum Standards Final Deliverable, to notify the TM FORUM Team Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this deliverable.

The TM FORUM invites any party to contact the TM FORUM Team Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this TM FORUM Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this TM FORUM Standards Final Deliverable. TM FORUM may include such claims on its website but disclaims any obligation to do so.

TM FORUM takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this TM FORUM Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on TM FORUM's procedures with respect to rights in any document or deliverable produced by a TM FORUM Collaboration Project Team can be found on the TM FORUM website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this TM FORUM Standards Final Deliverable, can be obtained from the TM FORUM Team Administrator. TM FORUM makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

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 2
- Table of Contents 4
- List of Tables 6
- Introduction 7
- SAMPLE USE CASES..... 8
- Support of polymorphism and extension patterns 9
- RESOURCE MODEL..... 10
 - Managed Entity and Task Resource Models 10
 - Test Data Instance resource 10
 - Test Data Schema resource 17
 - Notification Resource Models 24
 - Test Data Instance Create Event 25
 - Test Data Instance Change Event 26
 - Test Data Instance Delete Event 26
 - Test Data Instance Attribute Value Change Event 26
 - Test Data Instance State Change Event..... 27
 - Test Data Schema Create Event 27
 - Test Data Schema Change Event 28
 - Test Data Schema Delete Event 28
 - Test Data Schema Attribute Value Change Event 28
 - Test Data Schema State Change Event..... 29
- API OPERATIONS..... 30
 - Operations on Test Data Instance 30
 - List test data instances 30
 - Retrieve test data instance..... 32
 - Create test data instance 33
 - Patch test data instance 35
 - Delete test data instance..... 37
 - Operations on Test Data Schema 37
 - List test data schemas 37

Retrieve test data schema.....	38
Create test data schema.....	39
Patch test data schema	42
Delete test data schema.....	43
API NOTIFICATIONS.....	45
Register listener	45
Unregister listener	46
Publish Event to listener	46
Appendix 1	48
Examples of base64-encoded Attachment content for ONAP VTP	48
Example 1. ONAP VTP test data instance representation example	48
Acknowledgements	49
Version History.....	49
Release History	49
Contributors to Document.....	49

List of Tables

N/A

Introduction

The following document is the specification of the REST API for test data management. It includes the model definition as well as all available operations.

SAMPLE USE CASES

Reader will find example of use cases in TMF913 “Test API Component Suite” 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 `TestDataInstance` and `TestDataSchema` inheriting properties from the abstract `ManagedArtifact` 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 `ManagedArtifact` instances some may be instances of `TestDataInstance` where other could be instances of `TestDataSchema`. 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 `@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.

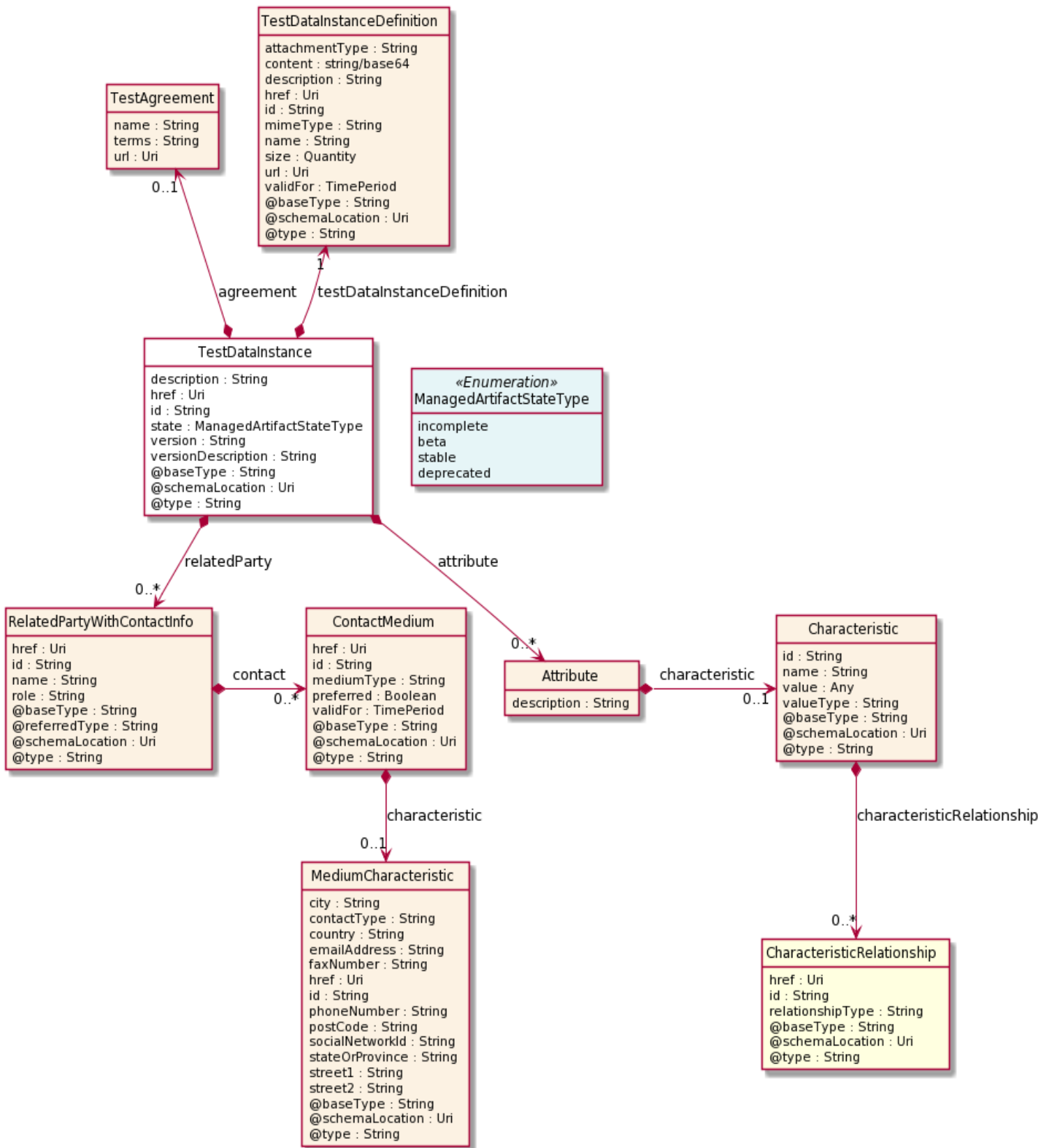
RESOURCE MODEL

Managed Entity and Task Resource Models

Test Data Instance resource

A managed test data instance resource.

Resource model



Field descriptionsTestDataInstance fields

description	A string. The description for the artifact.
version	A string. The artifact version.
versionDescription	A string. The artifact version description.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
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.
testDataInstanceDefinition	A test data instance definition (TestDataInstanceDefinition). Test data instance definition.
agreement	A test agreement (TestAgreement).
attribute	A list of attributes (Attribute [*]).
relatedParty	A list of related party with contact infos (RelatedPartyWithContactInfo [*]). Related party specialization that includes contact information.
state	A managed artifact state type (ManagedArtifactStateType). Possible values for the state of a managed artifact: 'incomplete', 'beta', 'stable', 'deprecated'.

Attribute sub-resource

characteristic	A characteristic (Characteristic). Describes a given characteristic of an object or entity through a name/value pair.
description	A string. The attribute description.

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.

ContactMedium sub-resource

Indicates the contact medium that could be used to contact the party.

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.

characteristic	A medium characteristic (MediumCharacteristic). Any additional characteristic(s) of this contact medium.
mediumType	A string. Type of the contact medium, such as: email address, telephone number, postal address.
preferred	A boolean. If true, indicates that is the preferred contact medium.
validFor	A time period. The time period that the contact medium is valid for.

MediumCharacteristic sub-resource

Describes the contact medium characteristics that could be used to contact a party (an individual or an organization).

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.
city	A string. The city.
contactType	A string. The type of contact, for example: phone number such as mobile, fixed home, fixed office. postal address such as shipping installation....
country	A string. The country.
emailAddress	A string. Full email address in standard format.
faxNumber	A string. The fax number of the contact.
phoneNumber	A string. The primary phone number of the contact.
postCode	A string. Postcode.
socialNetworkId	A string. Identifier as a member of a social network.
stateOrProvince	A string. State or province.
street1	A string. Describes the street.
street2	A string. Complementary street description.

Quantity sub-resource

An amount in a given unit.

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

RelatedPartyWithContactInfo sub-resource

Related party specialization that includes contact information.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
role	A string. Role played by the related party.
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.
contact	A list of contact mediums (ContactMedium [*]). Indicates the contact medium that could be used to contact the party.

TestAgreement sub-resource

name	A string. The agreement name.
terms	A string. The terms of the agreement.
url	An uri (Uri). The agreement URL.

TestDataInstanceDefinition sub-resource

Test data instance definition.

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.

description	A string. A narrative text describing the content of the attachment.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
mimeType	A string. Attachment mime type such as extension file for video, picture and document.
name	A string. The name of the attachment.
url	An uri (Uri). Uniform Resource Locator, is a web page address (a subset of URI).
@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.
size	A quantity (Quantity). The size of the attachment.
validFor	A time period. The period of time for which the attachment is valid.

Json representation sample

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

```
{
  "id": "2db74193-e5fb-462a-98e0-6b1ed970dfc7",
  "href": "https://mycsp:8080/tmf-api/testData/v4/testDataInstance/2db74193-e5fb-462a-98e0-6b1ed970dfc7",
  "description": "Input parameters for test case execution",
  "version": "1.2.3",
  "relatedParty": [
    {
      "name": "testUser",
      "role": "Owner",
      "@referredType": "Individual",
      "contact": [
        {
          "mediumType": "email",
          "characteristic": {
            "emailAddress": "testUser@abc.com"
          }
        }
      ]
    }
  ],
  "state": "stable",
  "testDataInstanceDefinition": {
    "attachmentType": "json",
    "content": "<base64-encoded test data instance>",
    "mimeType": "base64"
  }
}
```

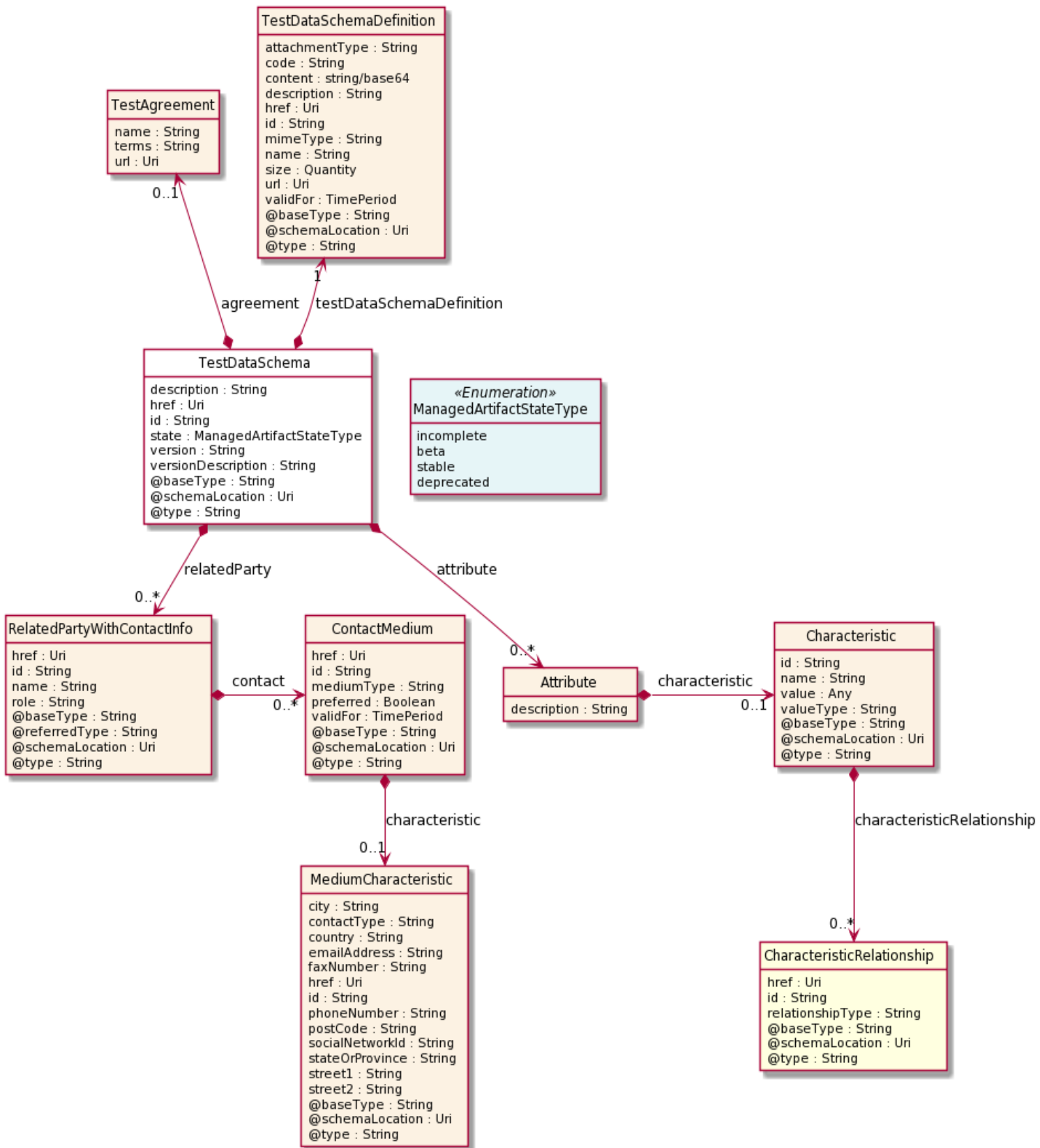


```
}  
}
```

Test Data Schema resource

A managed test data schema resource.

Resource model



Field descriptionsTestDataSchema fields

description	A string. The description for the artifact.
version	A string. The artifact version.
versionDescription	A string. The artifact version description.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
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.
testDataSchemaDefinition	A test data schema definition (TestDataSchemaDefinition). Test data schema definition.
agreement	A test agreement (TestAgreement).
attribute	A list of attributes (Attribute [*]).
relatedParty	A list of related party with contact infos (RelatedPartyWithContactInfo [*]). Related party specialization that includes contact information.
state	A managed artifact state type (ManagedArtifactStateType). Possible values for the state of a managed artifact: 'incomplete', 'beta', 'stable', 'deprecated'.

Attribute sub-resource

characteristic	A characteristic (Characteristic). Describes a given characteristic of an object or entity through a name/value pair.
description	A string. The attribute description.

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.

ContactMedium sub-resource

Indicates the contact medium that could be used to contact the party.

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.

characteristic	A medium characteristic (MediumCharacteristic). Any additional characteristic(s) of this contact medium.
mediumType	A string. Type of the contact medium, such as: email address, telephone number, postal address.
preferred	A boolean. If true, indicates that is the preferred contact medium.
validFor	A time period. The time period that the contact medium is valid for.

MediumCharacteristic sub-resource

Describes the contact medium characteristics that could be used to contact a party (an individual or an organization).

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.
city	A string. The city.
contactType	A string. The type of contact, for example: phone number such as mobile, fixed home, fixed office. postal address such as shipping installation....
country	A string. The country.
emailAddress	A string. Full email address in standard format.
faxNumber	A string. The fax number of the contact.
phoneNumber	A string. The primary phone number of the contact.
postCode	A string. Postcode.
socialNetworkId	A string. Identifier as a member of a social network.
stateOrProvince	A string. State or province.
street1	A string. Describes the street.
street2	A string. Complementary street description.

Quantity sub-resource

An amount in a given unit.

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

RelatedPartyWithContactInfo sub-resource

Related party specialization that includes contact information.

@referredType	A string. The actual type of the target instance when needed for disambiguation.
role	A string. Role played by the related party.
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.
contact	A list of contact mediums (ContactMedium [*]). Indicates the contact medium that could be used to contact the party.

TestAgreement sub-resource

name	A string. The agreement name.
terms	A string. The terms of the agreement.
url	An uri (Uri). The agreement URL.

TestDataSchemaDefinition sub-resource

Test data schema definition.

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.

description	A string. A narrative text describing the content of the attachment.
href	An uri (Uri). Hyperlink reference.
id	A string. unique identifier.
contentType	A string. Attachment mime type such as extension file for video, picture and document.
name	A string. The name of the attachment.
url	An uri (Uri). Uniform Resource Locator, is a web page address (a subset of URI).
@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.
code	A string. The test data schema code.
size	A quantity (Quantity). The size of the attachment.
validFor	A time period. The period of time for which the attachment is valid.

Json representation sample

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

```
{
  "id": "f7a4914b-e2b0-43db-92ec-5457688a9cde",
  "href": "https://mycsp:8080/tmf-api/testData/v4/testDataSchema/f7a4914b-e2b0-43db-92ec-5457688a9cde",
  "description": "Sample test data schema",
  "version": "1.2.3",
  "relatedParty": [
    {
      "name": "testUser",
      "role": "Owner",
      "@referredType": "Individual",
      "contact": [
        {
          "mediumType": "email",
          "characteristic": {
            "emailAddress": "testUser@abc.com"
          }
        }
      ]
    }
  ]
},
"state": "stable",
"testDataSchemaDefinition": {
  "attachmentType": "json",
```

```
"content": "<base64-encoded test data schema>",  
"mimeType": "base64"  
}  
}
```

Notification Resource Models

10 notifications are defined for this API

Notifications related to TestDataInstance:

- TestDataInstanceCreateEvent
- TestDataInstanceChangeEvent
- TestDataInstanceDeleteEvent
- TestDataInstanceAttributeValueChangeEvent
- TestDataInstanceStateChangeEvent

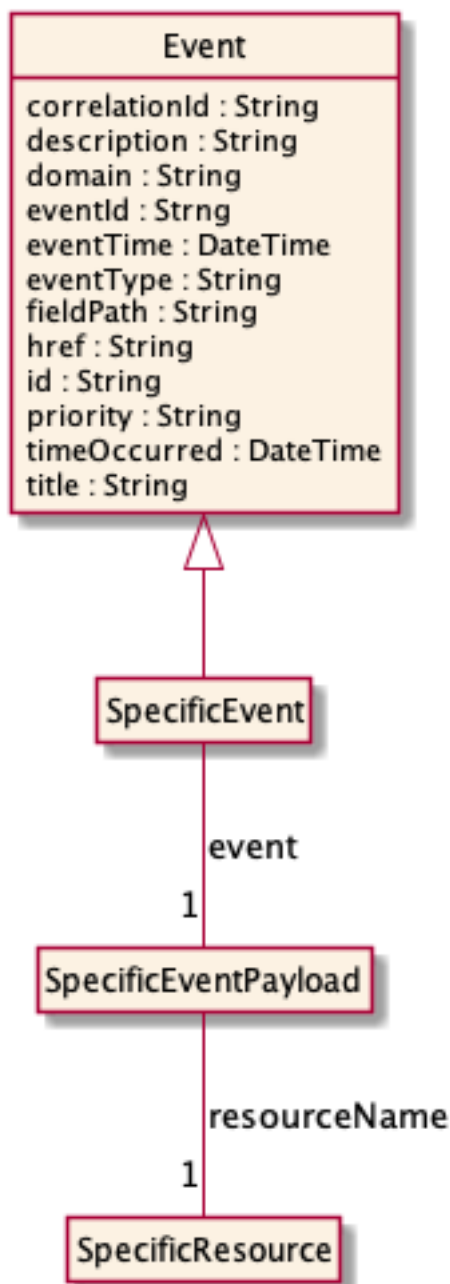
Notifications related to TestDataSchema:

- TestDataSchemaCreateEvent
- TestDataSchemaChangeEvent
- TestDataSchemaDeleteEvent
- TestDataSchemaAttributeValueChangeEvent
- TestDataSchemaStateChangeEvent

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).



Test Data Instance Create Event

Notification TestDataInstanceCreateEvent case for resource TestDataInstance

Json representation sample

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

```

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

```

```
"eventType":"TestDataInstanceCreateEvent",
"event": {
  "testDataInstance" :
    [-- SEE TestDataInstance RESOURCE SAMPLE --]
}
}
```

Test Data Instance Change Event

Notification TestDataInstanceChangeEvent case for resource TestDataInstance

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"TestDataInstanceChangeEvent",
  "event": {
    "testDataInstance" :
      [-- SEE TestDataInstance RESOURCE SAMPLE --]
  }
}
```

Test Data Instance Delete Event

Notification TestDataInstanceDeleteEvent case for resource TestDataInstance

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"TestDataInstanceDeleteEvent",
  "event": {
    "testDataInstance" :
      [-- SEE TestDataInstance RESOURCE SAMPLE --]
  }
}
```

Test Data Instance Attribute Value Change Event

Notification TestDataInstanceAttributeValueChangeEvent case for resource TestDataInstance

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"TestDataInstanceAttributeValueChangeEvent",
  "event": {
    "testDataInstance" :
      {-- SEE TestDataInstance RESOURCE SAMPLE --}
  }
}
```

Test Data Instance State Change Event

Notification TestDataInstanceStateChangeEvent case for resource TestDataInstance

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"TestDataInstanceStateChangeEvent",
  "event": {
    "testDataInstance" :
      {-- SEE TestDataInstance RESOURCE SAMPLE --}
  }
}
```

Test Data Schema Create Event

Notification TestDataSchemaCreateEvent case for resource TestDataSchema

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"TestDataSchemaCreateEvent",
  "event": {
    "testDataSchema" :
      {-- SEE TestDataSchema RESOURCE SAMPLE --}
  }
}
```

```
}  
}
```

Test Data Schema Change Event

Notification TestDataSchemaChangeEvent case for resource TestDataSchema

Json representation sample

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

```
{  
  "eventId":"00001",  
  "eventTime":"2015-11-16T16:42:25-04:00",  
  "eventType":"TestDataSchemaChangeEvent",  
  "event": {  
    "testDataSchema" :  
      {-- SEE TestDataSchema RESOURCE SAMPLE --}  
  }  
}
```

Test Data Schema Delete Event

Notification TestDataSchemaDeleteEvent case for resource TestDataSchema

Json representation sample

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

```
{  
  "eventId":"00001",  
  "eventTime":"2015-11-16T16:42:25-04:00",  
  "eventType":"TestDataSchemaDeleteEvent",  
  "event": {  
    "testDataSchema" :  
      {-- SEE TestDataSchema RESOURCE SAMPLE --}  
  }  
}
```

Test Data Schema Attribute Value Change Event

Notification TestDataSchemaAttributeValueChangeEvent case for resource TestDataSchema

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"TestDataSchemaAttributeValueChangeEvent",
  "event": {
    "testDataSchema" :
      {-- SEE TestDataSchema RESOURCE SAMPLE --}
  }
}
```

Test Data Schema State Change Event

Notification TestDataSchemaStateChangeEvent case for resource TestDataSchema

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2015-11-16T16:42:25-04:00",
  "eventType":"TestDataSchemaStateChangeEvent",
  "event": {
    "testDataSchema" :
      {-- SEE TestDataSchema RESOURCE SAMPLE --}
  }
}
```

API OPERATIONS

Remember the following Uniform Contract:

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

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

Notifications are also described in a subsequent section.

Operations on Test Data Instance

List test data instances

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

Description

This operation list test data instance 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 a list of test data instances.

Request
GET /tmf-api/testData/v4/testDataInstance Accept: application/json
Response
200 <pre>[{ "id": "2db74193-e5fb-462a-98e0-6b1ed970dfc7", "href": "https://mycsp:8080/tmf-api/testData/v4/testDataInstance/2db74193-e5fb-462a-98e0-6b1ed970dfc7", "description": "Input parameters for test case execution", "version": "1.2.3", "relatedParty": [{ "name": "testUser", "role": "Owner", "@referredType": "Individual", "contact": [{ "mediumType": "email", "characteristic": { "emailAddress": "testUser@abc.com" } }] }] }, { "state": "stable", "testDataInstanceDefinition": { "content": "<base64-encoded test data instance> - please see Example 1 in Appendix 1", "mimeType": "base64", "url": "http://myservice.com/artifact/1234" } }]</pre>

Retrieve test data instance

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

Description

This operation retrieves a test data instance entity.

Attribute selection is enabled for all first level attributes.

Filtering on sub-resources may be available depending on the compliance level supported by an implementation.

Usage Samples

Here's an example of a request for retrieving a specific test data instance.

Request
<pre>GET /tmf-api/testData/v4/testDataInstance/2db74193-e5fb-462a-98e0-6b1ed970dfc7 Accept: application/json</pre>
Response
<pre>200 { "id": "2db74193-e5fb-462a-98e0-6b1ed970dfc7", "href": "https://mycsp:8080/tmf-api/testData/v4/testDataInstance/2db74193-e5fb-462a-98e0-6b1ed970dfc7", "description": "Input parameters for test case execution", "version": "1.2.3", "relatedParty": [{ "name": "testUser", "role": "Owner", "@referredType": "Individual", "contact": [{ "mediumType": "email", "characteristic": { "emailAddress": "testUser@abc.com" } }] }], "state": "stable", "testDataInstanceDefinition": { "content": "<base64-encoded test data instance> - please see Example 1 in Appendix 1", "mimeType": "base64",</pre>


```

    "url": "http://myservice.com/artifact/1234"
  }
}
    
```

Create test data instance

POST /testDataInstance

Description

This operation creates a test data instance entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a TestDataInstance, 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
description	
version	
termsOfService	
contact	
license	
testDataInstanceDefinition	

Non Mandatory Attributes	Rule
versionDescription	
@baseType	
@schemaLocation	
@type	
agreement	
attribute	
relatedParty	
state	

Usage Samples

Here's an example of a request for creating a test data instance.

```

Request

POST /tmf-api/testData/v4/testDataInstance
Content-Type: application/json

{
    
```

```

"description": "Input parameters for test case execution",
"version": "1.2.3",
"relatedParty": [
  {
    "name": "testUser",
    "@referredType": "Individual",
    "role": "Owner",
    "contact": [
      {
        "mediumType": "email",
        "characteristic": {
          "emailAddress": "testUser@abc.com"
        }
      }
    ]
  }
],
"state": "stable",
"testDataInstanceDefinition": {
  "attachmentType": "json",
  "content": "<base64-encoded test data instance> - please see Example 1 in Appendix 1",
  "mimeType": "base64"
}
}

```

Response

201

```

{
  "id": "2db74193-e5fb-462a-98e0-6b1ed970dfc7",
  "href": "https://mycsp:8080/tmf-api/testData/v4/testDataInstance/2db74193-e5fb-462a-98e0-6b1ed970dfc7",
  "description": "Input parameters for test case execution",
  "version": "1.2.3",
  "relatedParty": [
    {
      "name": "testUser",
      "role": "Owner",
      "@referredType": "Individual",
      "contact": [
        {
          "mediumType": "email",
          "characteristic": {
            "emailAddress": "testUser@abc.com"
          }
        }
      ]
    }
  ],
  "state": "stable",
  "testDataInstanceDefinition": {
    "content": "<base64-encoded test data instance> - please see Example 1 in Appendix 1",
  }
}

```

```

"mimeType": "base64",
"url": "http://myservice.com/artifact/1234"
}
}

```

Patch test data instance

PATCH /testDataInstance/{id}

Description

This operation allows partial updates of a test data instance 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	
versionDescription	
@baseType	
@schemaLocation	
@type	
testDataInstanceDefinition	
agreement	
attribute	
relatedParty	
state	

Non Patchable Attributes	Rule
id	
href	
version	

Usage Samples

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

Request

PATCH /tmf-api/testData/v4/testDataInstance/2db74193-e5fb-462a-98e0-6b1ed970dfc7
Content-Type: application/merge-patch+json

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

Response

200

```
{
  "id": "2db74193-e5fb-462a-98e0-6b1ed970dfc7",
  "href": "https://mycsp:8080/tmf-api/testData/v4/testDataInstance/2db74193-e5fb-462a-98e0-6b1ed970dfc7",
  "description": "Input parameters for test case execution",
  "version": "1.2.3",
  "relatedParty": [
    {
      "name": "testUser",
      "role": "Owner",
      "@referredType": "Individual",
      "contact": [
        {
          "mediumType": "email",
          "characteristic": {
            "emailAddress": "testUser@abc.com"
          }
        }
      ]
    }
  ],
  "state": "stable",
  "testDataInstanceDefinition": {
    "attachmentType": "json",
    "content": "<base64-encoded test data instance>",
    "mimeType": "base64"
  },
  "name": "new name"
}
```

Delete test data instance

DELETE /testDataInstance/{id}

Description

This operation deletes a test data instance entity.

Usage Samples

Here's an example of a request for deleting a test data instance.

Request
DELETE /tmf-api/testData/v4/testDataInstance/2db74193-e5fb-462a-98e0-6b1ed970dfc7
Response
204

Operations on Test Data Schema

List test data schemas

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

Description

This operation list test data schema 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 a list of test data schemas.

Request
GET /tmf-api/testData/v4/testDataSchema Accept: application/json

Response

200

```
[
  {
    "id": "f7a4914b-e2b0-43db-92ec-5457688a9cde",
    "href": "https://mycsp:8080/tmf-api/testData/v4/testDataSchema/f7a4914b-e2b0-43db-92ec-5457688a9cde",
    "description": "Sample test data schema",
    "version": "1.2.3",
    "relatedParty": [
      {
        "name": "testUser",
        "role": "Owner",
        "@referredType": "Individual",
        "contact": [
          {
            "mediumType": "email",
            "characteristic": {
              "emailAddress": "testUser@abc.com"
            }
          }
        ]
      }
    ]
  },
  "state": "stable",
  "testDataSchemaDefinition": {
    "content": "<base64-encoded test data schema>",
    "mimeType": "base64",
    "url": "http://myservice.com/testDataSchema/1234"
  }
}
]
```

Retrieve test data schema**GET /testDataSchema/{id}?fields=...&{filtering}****Description**

This operation retrieves a test data schema entity.

Attribute selection is enabled for all first level attributes.

Filtering on sub-resources may be available depending on the compliance level supported by an implementation.

Usage Samples

Here's an example of a request for retrieving a specific test data schema.

Request
GET /tmf-api/testData/v4/testDataSchema/f7a4914b-e2b0-43db-92ec-5457688a9cde Accept: application/json
Response
200 <pre>{ "id": "f7a4914b-e2b0-43db-92ec-5457688a9cde", "href": "https://mycsp:8080/tmf-api/testData/v4/testDataSchema/f7a4914b-e2b0-43db-92ec-5457688a9cde", "description": "Sample test data schema", "version": "1.2.3", "relatedParty": [{ "name": "testUser", "role": "Owner", "@referredType": "Individual", "contact": [{ "mediumType": "email", "characteristic": { "emailAddress": "testUser@abc.com" } }] }], "state": "stable", "testDataSchemaDefinition": { "content": "<base64-encoded test data schema>", "mimeType": "base64", "url": "http://myservice.com/testDataSchema/1234" } }</pre>

Create test data schema

POST /testDataSchema

Description

This operation creates a test data schema entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a TestDataSchema, 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
description	
version	
termsOfService	
contact	
license	
testDataSchemaDefinition	

Non Mandatory Attributes	Rule
versionDescription	
@baseType	
@schemaLocation	
@type	
agreement	
attribute	
relatedParty	
state	

Usage Samples

Here's an example of a request for creating a test data schema.

```

Request
POST /tmf-api/testData/v4/testDataSchema
Content-Type: application/json

{
  "description": "Sample test data schema",
  "version": "1.2.3",
  "relatedParty": [
    {
      "name": "testUser",
      "@referredType": "Individual",
      "role": "Owner",
      "contact": [
        {
          "mediumType": "email",
          "characteristic": {
            "emailAddress": "testUser@abc.com"
          }
        }
      ]
    }
  ]
}
    
```



```
    ]
  }
],
"state": "stable",
"testDataSchemaDefinition": {
  "attachmentType": "json",
  "content": "<base64-encoded test case>",
  "mimeType": "base64"
}
}
```

Response

201

```
{
  "id": "f7a4914b-e2b0-43db-92ec-5457688a9cde",
  "href": "https://mycsp:8080/tmf-api/testData/v4/testDataSchema/f7a4914b-e2b0-43db-92ec-5457688a9cde",
  "description": "Sample test data schema",
  "version": "1.2.3",
  "relatedParty": [
    {
      "name": "testUser",
      "role": "Owner",
      "@referredType": "Individual",
      "contact": [
        {
          "mediumType": "email",
          "characteristic": {
            "emailAddress": "testUser@abc.com"
          }
        }
      ]
    }
  ]
},
"state": "stable",
"testDataSchemaDefinition": {
  "content": "<base64-encoded test data schema>",
  "mimeType": "base64",
  "url": "http://myservice.com/testDataSchema/1234"
}
}
```

Patch test data schema

PATCH /testDataSchema/{id}

Description

This operation allows partial updates of a test data schema 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	
versionDescription	
@baseType	
@schemaLocation	
@type	
testDataSchemaDefinition	
agreement	
attribute	
relatedParty	
state	

Non Patchable Attributes	Rule
id	
href	
version	

Usage Samples

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

Request
PATCH /tmf-api/testData/v4/testDataSchema/f7a4914b-e2b0-43db-92ec-5457688a9cde Content-Type: application/merge-patch+json {

<pre> "name": "new name" } </pre>
<p>Response</p>
<pre> 200 { "id": "f7a4914b-e2b0-43db-92ec-5457688a9cde", "href": "https://mycsp:8080/tmf-api/testData/v4/testDataSchema/f7a4914b-e2b0-43db-92ec-5457688a9cde", "description": "Sample test data schema", "version": "1.2.3", "relatedParty": [{ "name": "testUser", "role": "Owner", "@referredType": "Individual", "contact": [{ "mediumType": "email", "characteristic": { "emailAddress": "testUser@abc.com" } }] }], "state": "stable", "testDataSchemaDefinition": { "attachmentType": "json", "content": "<base64-encoded test data schema>", "mimeType": "base64" }, "name": "new name" } </pre>

Delete test data schema

DELETE /testDataSchema/{id}

Description

This operation deletes a test data schema entity.

Usage Samples

Here's an example of a request for deleting a test data schema.

Request
DELETE /tmf-api/testData/v4/testDataSchema/f7a4914b-e2b0-43db-92ec-5457688a9cde
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
<pre>POST /api/hub Accept: application/json {"callback": "http://in.listener.com"}</pre>
Response
<pre>201 Content-Type: application/json Location: /api/hub/42 {"id": "42", "callback": "http://in.listener.com", "query": null}</pre>

Unregister listener

DELETE /hub/{id}

Description

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

Behavior

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

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

Usage Samples

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

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

Publish Event to listener

POST /client/listener

Description

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

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

Behavior

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

Usage Samples

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

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

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

Appendix 1

Examples of base64-encoded Attachment content for ONAP VTP

Example 1. ONAP VTP test data instance representation example

Encoded value
ewogICAgImFyZzEiOiAic2FtcGxIHZhbHVllgp9Cg==
Original /decoded JSON object
<pre>{ "arg1": "sample value" }</pre>

Acknowledgements

Version History

Version Number	Date	Release led by:	Description
4.0.0	29-Jul-2020	Pierre Gauthier TM Forum pgauthier@tmforum.org Edward Pershwitz ITEA Technologies edward.pershwitz@iteatechnologies.com	First Release of the Document.

Release History

Release Status	Date	Release led by:	Description
Pre-production	29-Jul-2020	Pierre Gauthier Edward Pershwitz	First Release of the Document.
Production	23-Sep-2020	Adrienne Walcott	Updated to reflect TM Forum Approved Status

Contributors to Document

Kanagaraj Manickam	Huawei Technologies Co., Ltd
--------------------	------------------------------