TM Forum Specification

Event Management API User Guide

TMF688

Team Approved Date: 07-Dec-2020

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

NOTICE

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:

181 New Road, Suite 304
Parsippany, NJ 07054 USA
Tel No. +1 973 944 5100
Fax No. +1 973 998 7916
TM Forum Web Page: www.tmforum.org

Table of Contents

List of Tables

N/A

Introduction

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

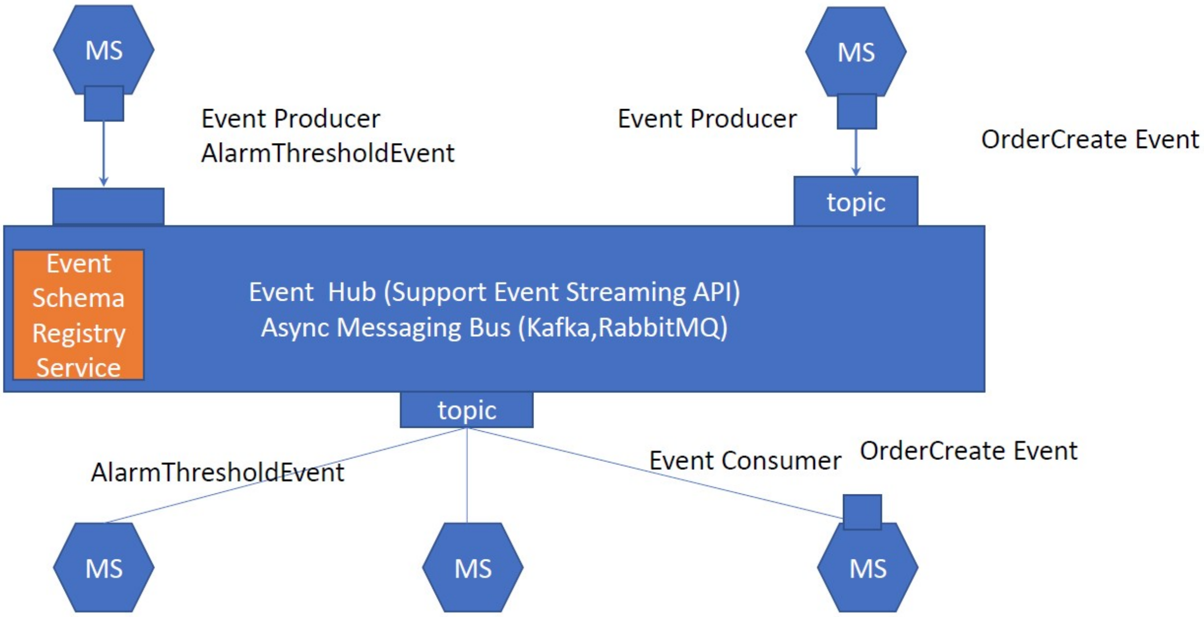
TMF APIs already support the capability of publishing Events to subscribers using pub/hub pattern over REST (using a Hub Resource see Design Guideline Part 1, section 12).
The current pub/hub subscribe pattern uses call-back based mechanism to deliver events to subscribers.

The Event API supports the large scale distributed publish and subscription for events generated by many TMF Open APIs.

This support for large scale publishing of multiple event types enable different scenarios enabling Event based architectures.

The Open Event Model provides a rich definition of Events, their payloads, their metadata using the concept of schema defined events-building on our polymorphic design pattern.

The Event API can be implemented using scalable event frameworks like Kafka or Rabbit MQ.



SAMPLE USE CASES

Event queries (Use Case 1)

The Event API can be used to query the resource instances to acquire event states and the attributes. For example, acquiring a list of AlarmNotificationEvents by filtering the eventType.

Event create (Use Case 2)

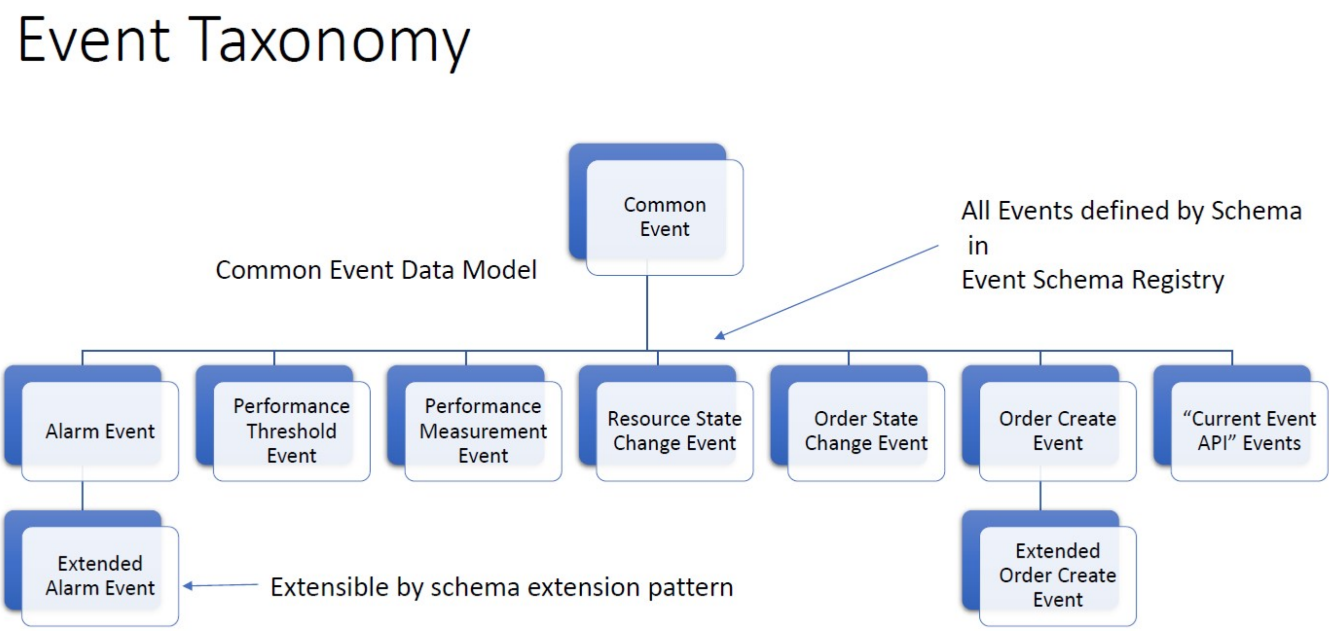
The Event API can be used to create to store the event instances. For example, create a production topic first and post an alarm Event into that topic/event resource.

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 the “Alarm Event” or the “Order Create Event are all instances of Events derived from the Common Event type.

This means the common event data model can be extended by any kind of event payload because of the polymorphic data structure definition.

By that polymorphic pattern the common event data model describes the event taxonomy of any event type.


Event Taxonomy
Common Event
Common Event Data Model
All Events defined by Schema in Event Schema Registry
Alarm Event
Performance Threshold Event
Performance Measurement Event
Resource State Change Event
Order State Change Event
Order Create Event
“Current Event API” Events
Extended Alarm Event
Extensible by schema extension pattern
Extended Order Create Event

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 Event instances some may be instances of Alarm Events where other could be instances of OrderCreation Events. 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 AlarmRef 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 AlarmRef or OrderCreateRef, 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.

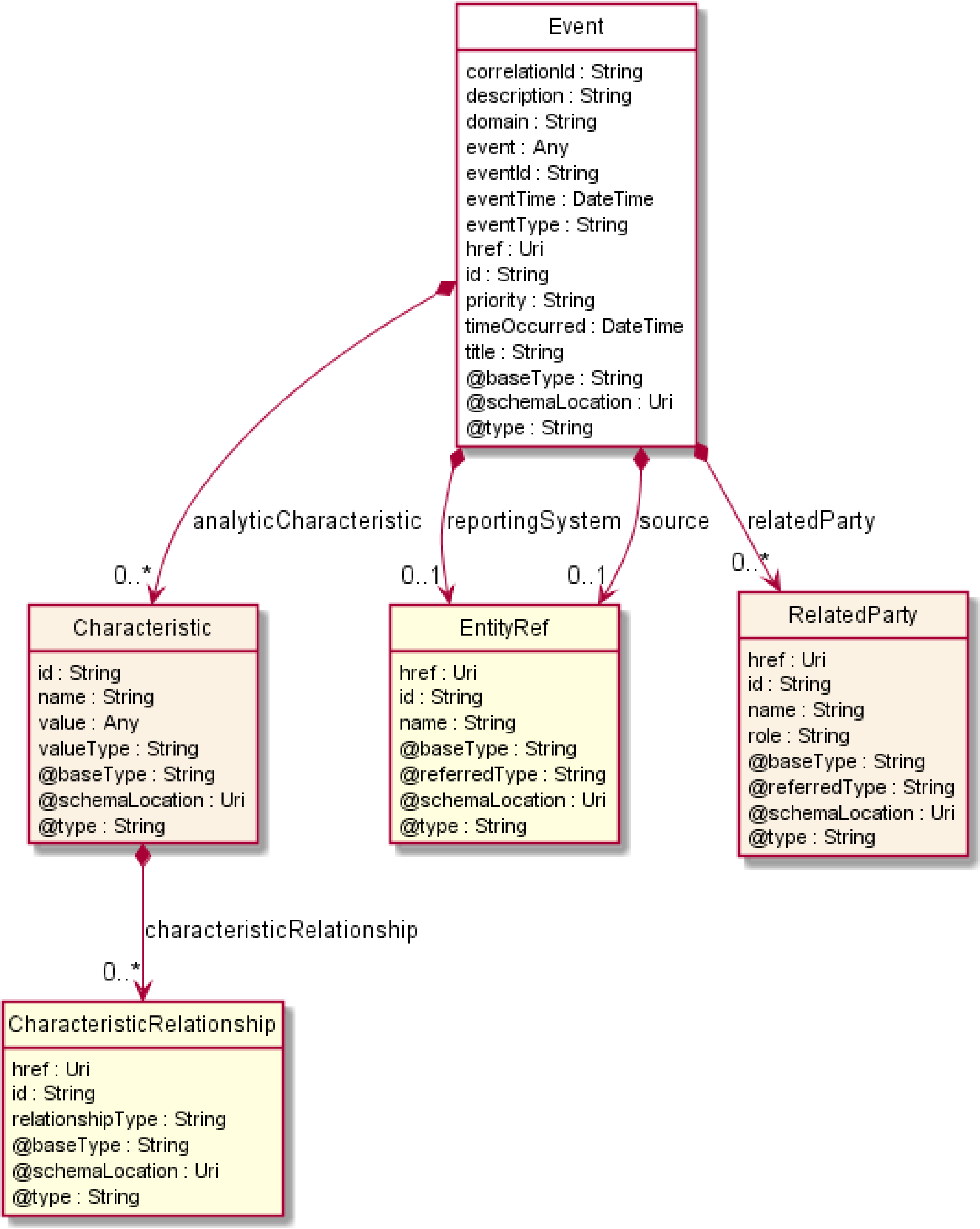
RESOURCE MODEL

Managed Entity and Task Resource Models

Event resource

The base Event with all the common attributes.

Resource model


Event
correlationId : String
description : String
domain : String
event : Any
eventId : String
eventTime : DateTime
eventType : String
href : Uri
id : String
priority : String
timeOccurred : DateTime
title : String
@baseType : String
@schemaLocation : Uri
@type : String
analyticCharacteristic
0..*
reportingSystem
0..1
source
0..1
relatedParty
0..*
Characteristic
id : String
name : String
value : Any
valueType : String
@baseType : String
@schemaLocation : Uri
@type : String
EntityRef
href : Uri
id : String
name : String
@baseType : String
@referredType : String
@schemaLocation : Uri
@type : String
RelatedParty
href : Uri
id : String
name : String
role : String
@baseType : String
@referredType : String
@schemaLocation : Uri
@type : String
characteristicRelationship
0..*
CharacteristicRelationship
href : Uri
id : String
relationshipType : String
@baseType : String
@schemaLocation : Uri
@type : String

Field descriptions

Event 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.

analyticCharacteristic A list of characteristics (Characteristic [*]). Describes a given characteristic of an object or entity through a name/value pair.

correlationId A string. The correlation id for this event.

description A string. An explanatory of the event.

domain A string. The domain of the event.

event An any (Any). The event linked to the involved resource object.

eventId A string. The identifier of the notification.

eventTime A date time (DateTime). Time of the event occurrence.

eventType A string. The type of the notification.

priority A string. A priority.

relatedParty A list of related parties (RelatedParty [*]). Related Entity reference. A related party defines party or party role linked to a specific entity.

reportingSystem An entity reference (EntityRef). Reporting System described by EntityRef.

source An entity reference (EntityRef). Source Entity described by EntityRef.

timeOccurred A date time (DateTime). The time the event occurred.

title A string. The title of the event.

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.

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.

EntityRef relationship

Entity reference schema to be use for all entityRef class.

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.

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

name A string. Name of the related entity.

Json representation sample

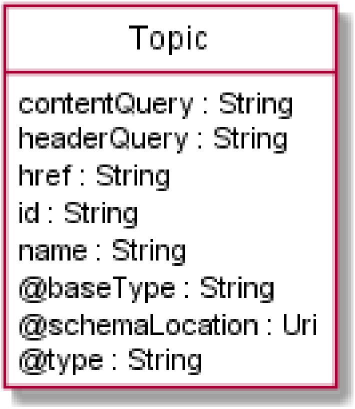
We provide below the json representation of an example of an 'Event' resource object

```
{
  "@type": "AlarmCreateEvent",
  "@schemaLocation": "http://xx/Event.schema.json",
  "@baseType": "event",
  "eventId": "256c42f0-7cae-4cfe-8b96-f5773796f8ff",
  "eventTime": "2019-11-02T14:20:54",
  "eventType": "AlarmCreateNotification",
  "correlationId": "238764827364827t367",
  "domain": "domain-x",
  "title": "Mail service not responding",
  "description": "The mail service is no longer responding and sending mails",
  "timeOccurred": "2019-11-02T14:20:54",
  "timeReceived": "2019-11-02T14:21:08",
  "priority": "Normal",
  "source": {
    "id": "12345",
    "href": ".../relatedEntity/12345"
  },
  "reportingSystem": {
    "id": "34534",
    "href": ".../reportingEntity/12345",
    "name": "name"
  },
  "relatedParty": [
    {
      "id": "12312",
      "href": "http://.../party/12345",
      "role": "Owner"
    }
  ],
  "event": {
    "alarm": {
      "@type": "alarm",
      "@schemaLocation": "http://../registry/Alarm.schema.json",
      "@baseType": "alarm",
      "id": "ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
      "href": "http://api/alarm/ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
      "externalAlarmId": "cisco-7609-1937465789",
      "alarmType": "QualityOfServiceAlarm",
      "perceivedSeverity": "CRITICAL",
      "probableCause": "Threshold crossed",
      "specificProblem": "Inbound Traffic threshold crossed"
    }
  }
}
```

Topic resource

Is an event channel provided by the Event Streaming API.

Resource model



Field descriptions

Topic fields

href A string. Reference of the related entity.

id A string. The identifier of the notification.

@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.

contentQuery A string. is the filter that will be applied on the content of the Event.

headerQuery A string. is the filter that will be applied on the Event header properties.

href A string. Reference of the related entity.

id A string. The identifier of the notification.

name A string. use to identify grouping of events, per domain, per event types, per access control-right and so on.

Json representation sample

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

```
{
   "name": "AlarmManagement-Paris",
   "headerQuery": "eventType=AlarmCreateNotification&domain=Paris",
   "contentQuery": "event.alarm.perceivedSeverity=CRITICAL",
   "id": "AlarmTopic",
   "href": "//topic/ AlarmTopic"
}
```

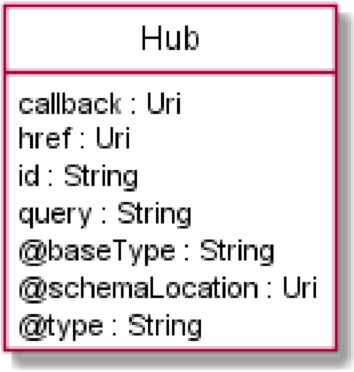
Hub resource

A Hub is used to subscribe to an event notification related to a particular Topic.

The Hub for topic is used to register a subscriber to subscribe the corresponding topic/event resource.

This can be used for sourcing and consuming events only allowed to those subscribers, which are registered into the Hub.

Resource model



Field descriptions

Hub fields

href An uri (Uri). Hyperlink reference.

id A string. The unique-id for your subscription - referenced when updating or deleting a subscription.

@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.

callback An uri (Uri). The URI that will be POSTed to when a notification is triggered.

id A string. The unique-id for your subscription - referenced when updating or deleting a subscription.

query A string. This is a query string used to filter notifications in the context of the notifier.

Json representation sample

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

```
{
    "callback": "http://www.tmforum.org/aListener",
    "id": "5880",
    "query": "event.alarm.alarmType=QualityOfServiceAlarm",
    "href": "//hub/ 5880"
}
```

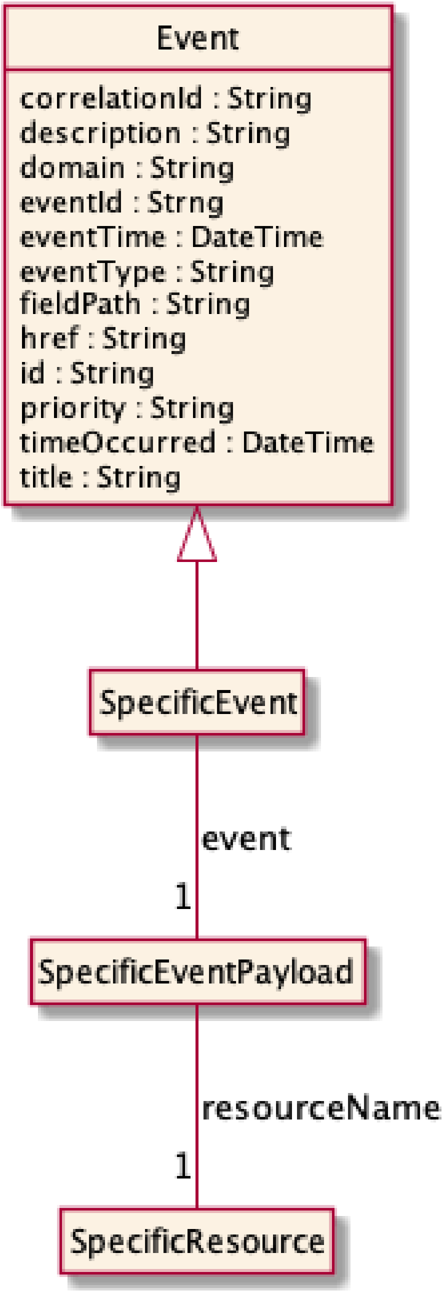
Notification Resource Models

4 notifications are defined for this API

Notifications related to Event:
- EventCreateEvent

Notifications related to Topic:
- TopicCreateEvent
- TopicChangeEvent
- TopicDeleteEvent

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).



Event Create Event

Notification EventCreateEvent case for resource Event

Json representation sample

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

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

Topic Create Event

Notification TopicCreateEvent case for resource Topic

Json representation sample

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

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"TopicCreateEvent",
    "event": {
      "topic" :
         {-- SEE Topic RESOURCE SAMPLE --}
   }
}
```

Topic Change Event

Notification TopicChangeEvent case for resource Topic

Json representation sample

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

```
{
   "eventId":"00001",
   "eventTime":"2015-11-16T16:42:25-04:00",
   "eventType":"TopicChangeEvent",
    "event": {
      "topic" :
         {-- SEE Topic RESOURCE SAMPLE --}
    }
}
```

Topic Delete Event

Notification TopicDeleteEvent case for resource Topic

Json representation sample

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

```
{
    "eventId":"00001",
    "eventTime":"2015-11-16T16:42:25-04:00",
    "eventType":"TopicDeleteEvent",
     "event": {
        "topic" :
            {-- SEE Topic RESOURCE SAMPLE --}
    }
}
```

Event Management by Topics

Topic/Event Resource Graph

Event Management supports the Event streaming concept by the Topic resource. The Topic resource is the target container for the event resource. The topic/event resource can be used for storing different events into different topics separated into domains. (e.g., Party, Resource, Product,..) A topic/event resource graph can be set-up for further processing and analytic functions.

The different topics can also be used for different access control on the topic/event resource because of the different endpoints of the Event API by topics.

Example: Creation of a “Resource” topic for Alarm events:

REQUEST

```
POST //eventhub/eventapi/topic
{
  "contentQuery": "Alarm.priority=normal",
  "headerQuery":  “eventType=AlarmCreateNotification”
  "name": "Resource"
  "@baseType": "topic",
  "@schemaLocation": "http://schemaregistry/Topic.schema.json",
  "@type": "topic"
}
```

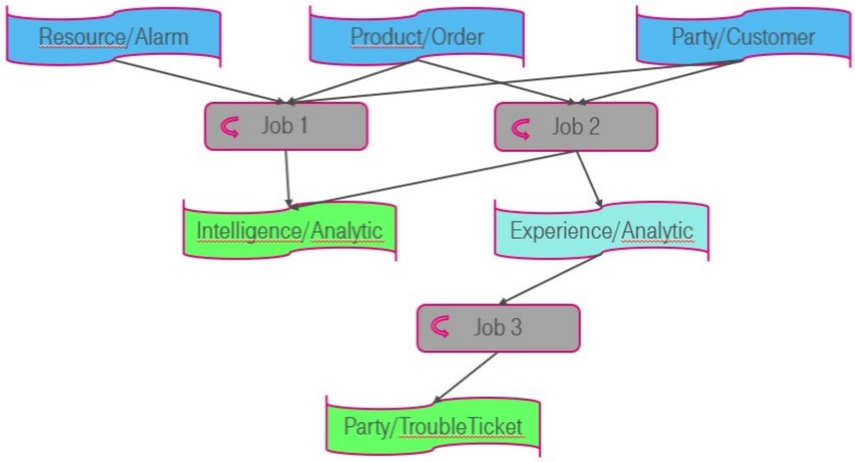
RESPONSE

```
201 Created
Content-Type: application/json

        {
            "id": "13",
            "href": "http://eventhub/eventapi/topic/13",
          "contentQuery": " Alarm.priority=normal ",
           "headerQuery": "eventType=AlarmCreateNotification",
           "name": "Resource",
           "@baseType": "topic",
           "@schemaLocation": "http://schemaregistry/Topic.schema.json",
           "@type": "topic"
        }
```

The example below shows how different event streams (e.g., Party/Customer & Resource/Alarm) might be used to create new event stream (e.g., experience Analytics by Job2) and to create a further event stream (Party/TroubleTicket by Job 3) holding information for customer trouble tickets into the Party/TroubleTicket event resource.

An event streaming transformation graph is shown below by topic/event resources.



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
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 Event

List events

```
GET topic/{topicId}/event?fields=...&{filtering}
```

Description

This operation list event 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 events by the eventType=AlarmCreateNotification.

Request

```
GET /tmf-api/event/v4/event?eventType=AlarmCreateNotification
Accept: application/json
```

Response

```
200

[
  {
    "@type": "AlarmCreateEvent",
    "@schemaLocation": "http://xx/Event.schema.json",
    "@baseType": "event",
    "id": "3e34689f-d4ae-44b7-bc97-d64eeaf2ba76",
    "href": "http://www.tmforum.org/tmf-api/eventStreaming/Event/3e34689f-d4ae-44b7-bc97-d64eeaf2ba76",
    "eventId": "256c42f0-7cae-4cfe-8b96-f5773796f8ff",
    "eventTime": "2016-07-02T14:20:54",
    "eventType": "AlarmCreateNotification",
    "correlationId": "238764827364827t367",
    "domain": "domain-x",
    "title": "Mail service not responding",
    "description": "The mail service is no longer responding and sending mails",
    "timeOccurred": "2016-07-02T14:20:54",
    "timeReceived": "2016-07-02T14:21:08",
    "priority": "Normal",
    "source": {
      "id": "12345",
      "href": ".../relatedEntity/12345"
    },
    "reportingSystem": {
      "id": "34534",
      "href": ".../reportingEntity/12345",
      "name": "name"
    },
    "relatedParty": [
      {
        "id": "12312",
        "href": "http://.../party/12345",
        "role": "Owner"
      }
    ],
    "event": {
      "alarm": {
```

```
                    "@type": "alarm",
                    "@schemaLocation": "http://../registry/Alarm.schema.json",
                    "@baseType": "alarm",
                    "id": "ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
                    "href": "http://api/alarm/ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
                    "externalAlarmId": "cisco-7609-1937465789",
                    "alarmType": "QualityOfServiceAlarm",
                    "perceivedSeverity": "CRITICAL",
                    "probableCause": "Threshold crossed",
                    "specificProblem": "Inbound Traffic threshold crossed"
                }
            }
        }
]
```

Retrieve event

GET topic/{topicId}/event/{id}?fields=...&{filtering}

Description

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

Usage Samples

Here's an example of a request for retrieving an Event.

Request

```
GET /tmf-api/event/v4/event/3e34689f-d4ae-44b7-bc97-d64eeaf2ba76
Accept: application/json
```

Response

```
200

{
    "@type": "AlarmCreateEvent",
    "@schemaLocation": "http://xx/Event.schema.json",
    "@baseType": "event",
    "id": "3e34689f-d4ae-44b7-bc97-d64eeaf2ba76",
    "href": "http://www.tmforum.org/tmf-api/eventStreaming/Event/3e34689f-d4ae-44b7-bc97-d64eeaf2ba76",
    "eventId": "256c42f0-7cae-4cfe-8b96-f5773796f8ff",
    "eventTime": "2016-07-02T14:20:54",
    "eventType": "AlarmCreateNotification",
    "correlationId": "238764827364827t367",
    "domain": "domain-x",
    "title": "Mail service not responding",
    "description": "The mail service is no longer responding and sending mails",
    "timeOccurred": "2016-07-02T14:20:54",
    "timeReceived": "2016-07-02T14:21:08",
    "priority": "Normal",
    "source": {
      "id": "12345",
      "href": ".../relatedEntity/12345"
    },
    "reportingRessource": {
      "href": ".../reportingEntity/12345",
      "name": "name"
    },
    "relatedParty": [
      {
        "role": "owner",
        "href": ".../party/12345"
      }
    ],
    "event": {
      "alarm": {
        "@type": "alarm",
        "@schemaLocation": "http://../registry/Alarm.schema.json",
        "@baseType": "alarm",
        "id": "ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
        "href": "http://api/alarm/ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
        "externalAlarmId": "cisco-7609-1937465789",
        "alarmType": "QualityOfServiceAlarm",
        "perceivedSeverity": "CRITICAL",
        "probableCause": "Threshold crossed",
        "specificProblem": "Inbound Traffic threshold crossed"
      }
    }
}
```

Create event

```
POST topic/{topicId}/event
```

Description

This operation creates an event entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating an Event, 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
@type	
event	
eventId	

Non Mandatory Attributes	Rule
@baseType	
@schemaLocation	
analyticCharacteristic	
correlationId	
description	
domain	
eventTime	
eventType	
priority	
relatedParty	
reportingSystem	
source	
timeOccurred	
title	

Usage Samples

Here's an example of a request for creating an Event.

Request

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

{
   "@type": "AlarmCreateEvent",
   "@schemaLocation": "http://xx/Event.schema.json",
   "@baseType": "event",
   "correlationId": "238764827364827t367",
   "description": "This  base event CreateAlarmNotification Event.",
   "domain": "domain-x",
   "eventId": "256c42f0-7cae-4cfe-8b96-f5773796f8ff",
   "eventTime": "2019-05-15T00:00",
   "eventType": "AlarmCreateNotificaion",
   "priority": "Normal",
   "timeOcurred": "2019-05-15T00:00",
   "title": "Mail service not responding",
   "source": {
      "id": "12345",
      "href": ".../relatedEntity/12345"
   },
   "reportingSystem": {
      "id": "34534",
      "href": ".../reportingEntity/12345",
      "name": "MailServer"
   },
   "relatedParty": [
      {
         "id": "12312",
         "href": "http://.../party/12345",
         "role": "Owner"
      }
   ],
   "event": {
      "alarm": {
         "@type": "alarm",
         "@schemaLocation": "http://../registry/Alarm.schema.json",
         "@baseType": "alarm",
         "id": "ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
         "href": "http://api/alarm/ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
         "externalAlarmId": "cisco-7609-1937465789",
         "alarmType": "QualityOfServiceAlarm",
         "perceivedSeverity": "CRITICAL",
         "probableCause": "Threshold crossed",
         "specificProblem": "Inbound Traffic threshold crossed"
      }
   }
}
```

Response

```
201

{
   "@type": "AlarmCreateEvent",
   "@schemaLocation": "http://xx/Event.schema.json",
   "@baseType": "event",
   "id": "3e34689f-d4ae-44b7-bc97-d64eeaf2ba76",
   "href": "http://www.tmforum.org/tmf-api/eventStreaming/Event/3e34689f-d4ae-44b7-bc97-d64eeaf2ba76",
   "eventId": "256c42f0-7cae-4cfe-8b96-f5773796f8ff",
   "eventTime": "2016-07-02T14:20:54",
   "eventType": "AlarmCreateNotification",
   "correlationId": "238764827364827t367",
   "domain": "domain-x",
   "title": "Mail service not responding",
   "description": "The mail service is no longer responding and sending mails",
   "timeOccurred": "2019-11-02T14:20:54",
   "timeReceived": "2019-11-02T14:21:08",
   "priority": "Normal",
   "source": {
      "id": "12345",
      "href": ".../relatedEntity/12345"
   },
   "reportingSystem": {
      "id": "34534",
      "href": ".../reportingEntity/12345",
      "name": "MailServer"
   },
   "relatedParty": [
      {
         "id": "12312",
         "href": "http://.../party/12345",
         "role": "Owner"
      }
   ],
   "event": {
      "alarm": {
         "@type": "alarm",
         "@schemaLocation": "http://../registry/Alarm.schema.json",
         "@baseType": "alarm",
         "id": "ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
         "href": "http://api/alarm/ROUTER_IF@Cisco-7609-6-4-4-4-14-14-4--Gi9/20@42",
         "externalAlarmId": "cisco-7609-1937465789",
         "alarmType": "QualityOfServiceAlarm",
         "perceivedSeverity": "CRITICAL",
         "probableCause": "Threshold crossed",
         "specificProblem": "Inbound Traffic threshold crossed"
      }
   }
}
```

Operations on Topic

List topics

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

Description

This operation list topic 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 topic(s).

Request

```
GET /tmf-api/event/v4/topic?name=Alarmmanagement-Paris
Accept: application/json
```

Response

```
200

[
  {
    "name": "Root",
    "id": "Root",
    "href": "https://www.tmforum.org/tmf-api/eventStreaming/topic/Root"
  },
  {
    "name": "AlarmManagement-Paris",
    "headerQuery": "eventType=AlarmCreateNotification&domain=Paris",
    "contentQuery": "event.alarm.perceivedSeverity=CRITICAL",
    "id": "13",
    "href": "https://www.tmforum.org/tmf-api/eventStreaming/topic/13"
  }
]
```

Retrieve topic

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

Description

This operation retrieves a topic 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 Topic retrieval.

Request

```
GET /tmf-api/event/v4/topic/13
Accept: application/json
```

Response

```
200

{
   "name": "AlarmManagement-Paris",
   "headerQuery": "eventType=AlarmCreateNotification&domain=Paris",
   "contentQuery": "event.alarm.perceivedSeverity=CRITICAL",
   "id": "13",
   "href": "https://www.tmforum.org/tmf-api/eventStreaming/topic/13"
}
```

Create topic

POST /topic

Description

This operation creates a topic entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a Topic, 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	
contentQuery	
headerQuery	

Usage Samples

Here's an example of a request for creating a Topic.

Request

```
POST /tmf-api/event/v4/topic
Content-Type: application/json

{
   "name": "AlarmManagement-Paris",
   "headerQuery": "eventType=AlarmCreateNotification&domain=Paris",
    "contentQuery": "event.alarm.perceivedSeverity=CRITICAL"
}
```

Response

```
201

{
    "name": "AlarmManagement-Paris",
    "headerQuery": "eventType=AlarmCreateNotification&domain=Paris",
    "contentQuery": "event.alarm.perceivedSeverity=CRITICAL",
    "id": "13",
    "href": "https://www.tmforum.org/tmf-api/eventStreaming/topic/13"
}
```

Delete topic

`DELETE /topic/{id}`

Description

This operation deletes a topic entity.

Usage Samples

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

Request

```
DELETE /tmf-api/event/v4/topic/13
```

Response

```
204
```

Operations on Hub

List hubs

`GET topic/{topicId}/hub?fields=...&{filtering}`

Description

This operation list hub 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 hub(s).

Request

```
GET /tmf-api/event/v4/hub
Accept: application/json
```

Response

```
200

[
    {
        "callback": "http://www.tmforum.org/aListener",
        "id": "b4bc4d21-76e4-48dc-b6de-3ffcbf2de739",
        "href": "https://www.tmforum.org/tmf-api/eventStreaming/hub/b4bc4d21-76e4-48dc-b6de-3ffcbf2de739",
        "query": "event.alarm.alarmType=QualityOfServiceAlarm"
    }
]
```

Retrieve hub

GET topic/{topicId}/hub/{id}?fields=...&{filtering}

Description

This operation retrieves a hub 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 Hub retrieval.

Request

```
GET /tmf-api/event/v4/hub/b4bc4d21-76e4-48dc-b6de-3ffcbf2de739
Accept: application/json
```

Response

```
200

{
  "callback": "http://www.tmforum.org/aListener",
  "id": "b4bc4d21-76e4-48dc-b6de-3ffcbf2de739",
  "href": "https://www.tmforum.org/tmf-api/eventStreaming/hub/b4bc4d21-76e4-48dc-b6de-3ffcbf2de739",
  "query": "event.alarm.alarmType=QualityOfServiceAlarm"
}
```

Create hub

POST topic/{topicId}/hub

Description

This operation creates a hub entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a Hub, 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
id	
callback	

Non Mandatory Attributes	Rule
@baseType	
@schemaLocation	
@type	
query	

Usage Samples

Here's an example of a request for creating a Hub.

Request

```
POST /tmf-api/event/v4/hub
```

```
Content-Type: application/json

{
   "callback": "http://www.tmforum.org/aListener",
   "query": "event.alarm.alarmType=QualityOfServiceAlarm"
}
```

Response

```
201

{
   "callback": "http://www.tmforum.org/aListener",
   "id": "5880",
   "href": "https://www.tmforum.org/tmf-api/eventStreaming/hub/5880",
   "query": "event.alarm.alarmType=QualityOfServiceAlarm"
}
```

Delete hub

`DELETE topic/{topicId}/hub/{id}`

Description

This operation deletes a hub entity.

Usage Samples

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

Request

```
DELETE /tmf-api/event/v4/hub/b4bc4d21-76e4-48dc-b6de-3ffcbf2de739
```

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

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 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

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.

Acknowledgements

Document History

Version History

Version Number	Date	Release led by:	Description
1.0	15-Apr-2017	Pierre Gauthier TM Forum pgauthier@tmforum.org Mariano Belaunde Orange Labs	First Release of the Document.
2.0	06-Nov-2018	Mariano Belaunde Orange Labs	Alignment with Guidelines 3.0
4.0.0	07-Dec-2020	Thomas Braun Deutsche Telekom AG	Based on the TMF Open API Common Data Model
4.0.0	26-Mar-2021	Adrienne Walcott	Updated to reflect TM Forum Approved Status

Release History

Release Number	Date	Release led by:	Description
Release 1.0	15-Apr-2017	Pierre Gauthier TM Forum pgauthier@tmforum.org Mariano Belaunde Orange Labs	First Release of the Document.
Release 2.0	06-Nov-2018	Mariano Belaunde Orange Labs	Alignment with Guidelines 3.0
Pre-production	07-Dec-2020	Thomas Braun Deutsche Telekom AG	Based on the TMF Open API Common Data Model
Production	26-Mar-2021	Adrienne Walcott	Updated to reflect TM Forum Approved Status

Contributors to Document

Thomas Braun	Deutsche Telekom AG
Pierre Gauthier	TM Forum