Frameworx Specification

Appointment API REST Specification

TMF646
Release 16.0.1
October 2016

Latest Update: Frameworx Release 16	**TM Forum Approved**
Version 1.0.2	**IPR Mode: RAND**

NOTICE

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:

240 Headquarters Plaza,
East Tower – 10th Floor,
Morristown, NJ 07960 USA
Tel No. +1 973 944 5100
Fax No. +1 973 944 5110
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 appointment management. It includes the model definition as well as all available operations. Possible actions are to check free slots and, then, creating, updating and retrieving appointment.

The appointment API provides a standardized mechanism to book an appointment with all the necessary appointment characteristics. First, the API consists in searching free slots based on parameters, as for example a party. Then, the appointment is created. The appointment has characteristics such as nature of appointment, place of appointment…

Appointment API performs the following operations:

- Retrieve free slots depending on filter criteria
- Create an appointment
- Cancel an appointment
- Update an appointment
- Reschedule an appointment.

SAMPLE USE CASES

The following table maps out the use cases :

Use case
Free slots are checked according to criteria
A new appointment is created
An existing appointment should be updated because its status has changed or party availabilities have changed
An appointment or a collection of appointment should be retrieved
An existing appointment is cancelled
An existing appointment has to be rescheduled

RESOURCE MODEL

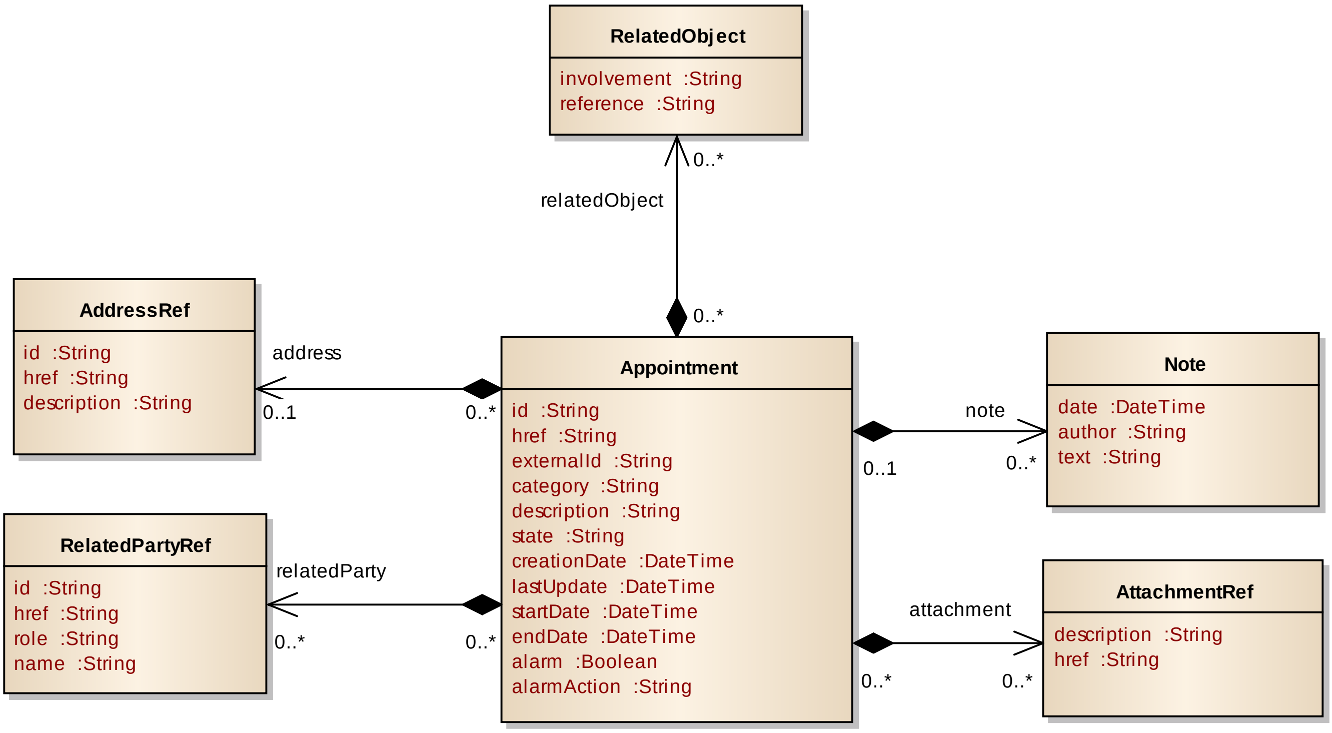
Managed Entity and Task Resource Models

APPOINTMENT RESOURCE

Structured textual way of describing what is an appointment.

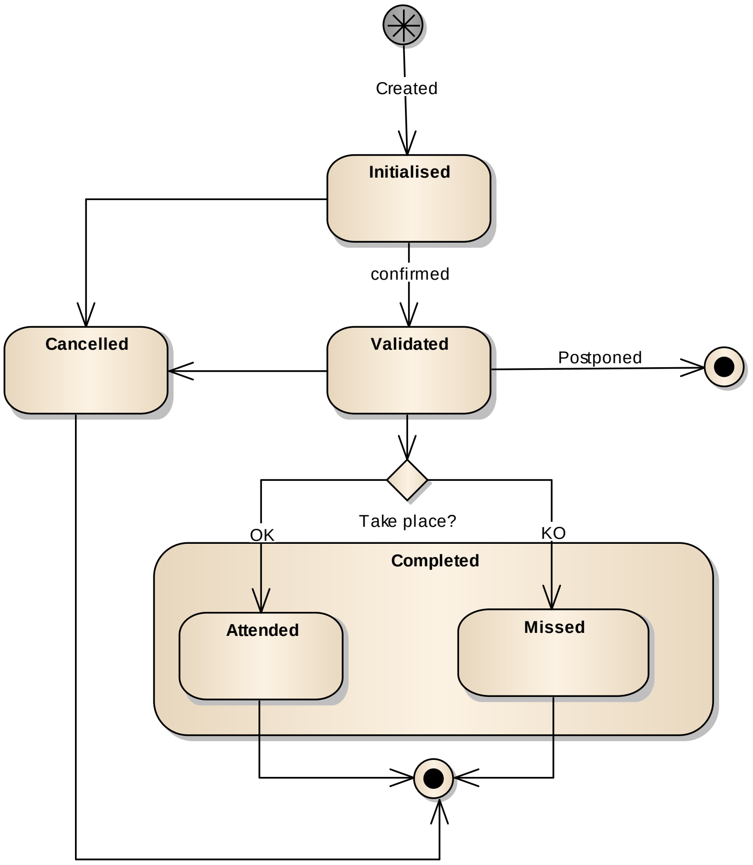
An appointment is a meeting with several persons, in one place, in order to do an action (an intervention, a sale, …). This action has a root, for example a trouble ticket.

Resource model



Lifecycle

The appointment lifecycle is tracked by mean of the state field. Typical lifecycle values are : initialized, validated, cancelled, attended and missed. The state machine specifying the typical state change transitions is provided below.



State	Description
initialised	When an appointment is created, the status is ‘initialised’
validated	When an appointment is confirmed by all parties, the status is ‘validated’
cancelled	When an appointment is not confirmed by at least one party, it is ‘cancelled’
attended	When an appointment took place and it is OK , the status is ‘attended’
missed	When an appointment took place and it is KO, the status is ‘missed’

Json representation sample

We provide below the json representation of an sample of an Appointment resource object :

```
{
   "id":"21",
   "href":"https://host:port/appointment/appointment/21",
   "externalId":"anExternalIDIfNeeded432113",
   "category":"intervention",
   "description":"A useful text to describe the appointment",
   "state":"missed",
   "creationDate":"2015-09-01T14:40:43.071Z",
   "lastUpdate":"2015-09-01T14:40:43.071Z",
   "startDate":"2015-09-01T14:00:43.071Z",
   "endDate":"2015-09-01T16:00:43.071Z",
   "alarm": true,
   "alarmAction":"smsToCustomer",
   "attachment":[
      {
         "description":"Short description of the document attached to the appointment",
         "href":"http://server/path/document1.pdf"
      }
   ],
   "relatedParty": [
      {
         "id":"32",
         "href":"https://host:port/partyManagement/individual/32",
         "role":"customer",
         "name":"John Doe"
      }
   ],
   "address":{
      "id":"7660828",
       "href" : "https://host:port/address/address/7660828",
       "description":"Complete address of the appointment"
   },
   "relatedObject":[
      {
         "involvement":"problemToSolve",
         "reference": "https://host:port/troubleTicket/troubleTicket/789745465"
      }
   ],
   "note":[
      {
         "date":"2015-09-01T14:40:43.071Z",
         "author":"Arthur Ewans",
         "text":"Already called the expert "
      }

   ]
}
```

Field descriptions

Appointment resource

Field	Description
id	A string. Unique identifier of the appointment
href	A string. Unique URI used to access to the appointment resource
externalId	A string. External reference known by the customer
description	A string. Short free text describing the appointment
category	A string. Business category : intervention for example or to be more precise afterSalesIntervention, orderDeliveryIntervention,…
state	A string. State corresponding to appointment lifecycle
creationDate	A date time. Appointment creation date
lastUpdate	A date time. Date of last appointment update
startDate	A date time. Appointment beginning date
endDate	A date time. Appointment end date
alarm	A boolean. Indicates if there is a reminder

alarmAction	A string. Action to be invoked when an alarm is triggered for all participants (send mail, send sms,…)
relatedParty	A list of related party references. Parties who participate to appointment. It can be a person (customer,…) or a team of persons (intervention team,…). There are at least two parties involved in an appointment
relatedObject	A list of related object references. Other resources linked to the appointment. For example, it can be an orderToDeliver for an order or a problemToSolve for a trouble ticket
attachment	A list of attachment references. URI related to attached documents to the appointment
address	An address reference. Place of appointment
note	A list of notes. Extra information about the appointment

relatedParty sub-resource

Field	Description
id	A string. Unique identifier of the party
href	A string. Unique URI used to access to the party resource
role	A string. Role played by the party (customer for example)
name	A string. Name of the party

relatedObject sub-resource

Field	Description
involvement	A string. Related object involved
reference	A string. An unique URI used to access to the corresponding resource of the related object

attachment sub-resource

Field	Description
href	A string. URI related to attached document to the appointment
description	A string. A short description of the attached document

Address sub-resource

Field	Description
id	A string. Unique identifier of the appointment address
href	A string. Unique URI used to access to the address resource
description	A string. Short text giving the complete address of the appointment

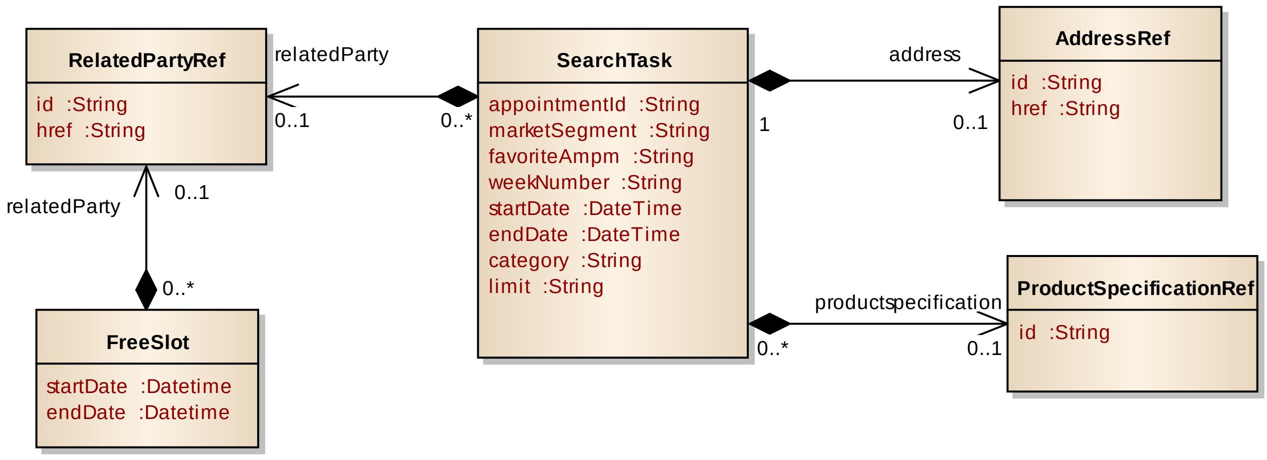
Note sub-resource

Field	Description
date	A datetime. Date of the note
author	A string. Author name
text	A string. Free text

SEARCH TASK RESOURCE

This task resource is used to look for free slots before booking an appointment (cf. operations).

Resource model



Lifecycle

No state machine for the resource search task.

Json representation samples

We provide below the json representation of a sample **of search task input** :

```
{
   "marketSegment":"B2C",
   "favoriteAmpm":"pm",
   "weekNumber":"38",
   "startDate":"2015-09-01T14:00:43.071Z",
   "endDate":"2015-09-01T16:00:43.071Z",
   "category":"intervention",
   "limit":"10",
   "productSpecification":{
      "id":"42"
   },
   "address":{
      "id": "7660828 " ,
      "href": "https://host:port/address/address/7660828",
      "description": "Complete address of the appointment"
   },
   "relatedParty":{
      "id":"32",
      "href":"https://host:port/partyManagement/individual/32"
   }
}
```

Field descriptions

Field	Description
marketSegment	A string. Market segment linked to the appointment
favoriteAmpm	A string. Favorite moment of the day for the party : am (for slot in the morning) or pm (for slot in the afternoon)
weekNumber	A string. Week number where free slots are searched
startDate	A datetime. Beginning date of the period for free slots search
endDate	A datetime. End date of the period for free slots search
category	A string. Appointment category
limit	A string. Limit number of free slots to be searched
productSpecification	A product specification reference. Product concerned by the appointment
address	An address reference. Appointment place - See appointment resource for fields description of this sub-resource
relatedParty	A party reference. Party who is the owner of the calendar on which we want to plan an appointment (for example an intervention team) – See appointment resource for fields description of this sub-resource

We provide below the json representation of a sample of **search task output** (a list of free slots) :

```
{
   "freeSlot":[
      {
         "startDate":"2015-09-01T14:00:43.071Z",
         "endDate":"2015-09-01T16:00:43.071Z",
         "relatedParty":{
             "id":"32",
             "href":"https://host:port/partyManagement/individual/32"
         }
      }
   ]
}
```

Field descriptions

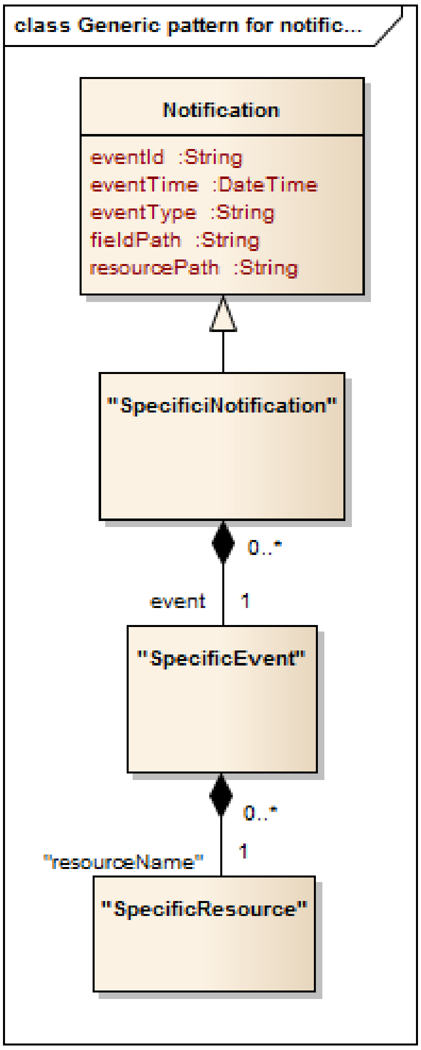
Field	Description
startDate	A datetime. Beginning date of the free slot
endDate	A datetime. End date of the free slot
relatedParty	A reference to a party. Party available during this free slot (for example an intervention team) – See appointment resource for fields description of this sub-resource

Notification Resource Models

Five notifications are defined for this API :

- appointmentStateChangeNotification
- appointmentRescheduleNotification
- appointmentCategoryChangeNotification
- appointmentCreationNotification
- appointmentAttributeValueChangeNotification

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



APPOINTMENT STATE CHANGE NOTIFICATION

Notification sent when the state of an appointment is updated.

Json representation samples

We provide below the json representation of an example of an “appointmentStateChangeNotification” notification object :

```
{
      "eventId":"00001",
      "eventTime":"2013-04-19T16:42:25-04:00",
      "eventType":"appointmentStateChangeNotification",
      "event": {
         "appointment": {
             "id":"21",
             "href":" https://host:port/appointment/appointment/21",
             "state":"missed"
         }
      }
}
```

APPOINTMENT RESCHEDULE NOTIFICATION

Notification sent when an appointment is rescheduled.

Json representation samples

We provide below the json representation of an example of an “appointmentRescheduleNotification” notification object :

```
{
      "eventId":"00001",
      "eventTime":"2013-04-19T16:42:25-04:00",
      "eventType":"appointmentRescheduleNotification",
      "event": {
         "appointment": {
             "id":"21",
             "href":" https://host:port/appointment/appointment/21",
             "startDate":"2015-09-01T14:00:00.071Z",
             "endDate":"2015-09-01T16:00:00.071Z"
         }
      }
}
```

APPOINTMENT CATEGORY CHANGE NOTIFICATION

Notification sent when the category of an appointment is updated.

Json representation samples

We provide below the json representation of an example of an “appointmentCategoryChangeNotification” notification object :

```
{
        "eventId":"00001",
        "eventTime":"2013-04-19T16:42:25-04:00",
        "eventType":"appointmentCategoryChangeNotification",
        "event": {
           "appointment": {
               "id":"21",
               "href":" https://host:port/appointment/appointment/21",
               "category":"intervention"
           }
        }
}
```

APPOINTMENT CREATION NOTIFICATION

Notification sent when a new appointment is created.

Json representation samples

We provide below the json representation of an example of an “appointmentCreationNotification” notification object :

```
{
        "eventId":"00001",
        "eventTime":"2013-04-19T16:42:25-04:00",
        "eventType":"appointmentCreationNotification",
        "event": {
           "appointment": {
               "id":"21",
               "href":" https://host:port/appointment/appointment/21",
               {--SEE Appointment Resource sample --}
           }
        }
}
```

APPOINTMENT ATTRIBUTE VALUE CHANGE NOTIFICATION

Notification sent when an appointment attribute is updated.

Json representation samples

We provide below the json representation of an example of an “appointmentAttributeValueChangeNotification” notification object :

```
{
      "eventId":"00001",
      "eventTime":"2013-04-19T16:42:25-04:00",
      "eventType":"appointmentAttributeValueChangeNotification",
      "event": {
         "appointment": {
             "id":"21",
             "href":"https://host:port/appointment/appointment/21",
             "alarm":"false"
         }
      }
}
```

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

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

Notifications are also described in a subsequent section.

OPERATIONS ON APPOINTMENT

Summary of operations

(1) A party (customer service representative, customer, etc.) wants to check free periods into a calendar

A party wants to book an appointment for a customer: the party checks free periods in a calendar.

This calendar can be one of a single person, or an aggregation of persons (a team).

In case of a team calendar, the party identification (competent/relevant team to perform an intervention) is realized via a context: a product specification (FTTH, Copper, etc.), a marketSegment (Pro / Residential), a place/localization, etc.

(2) A party wants to create an appointment

A party books a slot, this slot will be used to realize a task (an intervention, etc.) or to meet a customer service representative.

his booking is done on an organization calendar (a team, a shop, etc.).

(3) A party wants to delete an appointment

If, for example, the customer is not available anymore, the party can cancel the appointment.

(4) A party wants to update an appointment

If, for example, the customer is not available anymore, the party can update the appointment by changing the slot.

(5) A party wants to find appointments with criterias

A party can search all the appointments booked by a customer in a determined period for example.

(6) A party wants to reschedule an appointment

Availabilities of parties have changed. The appointment must be rescheduled.

CHECK FREE SLOTS INTO A CALENDAR (1)

POST /api/freeSlot/search

Description

This operation is used to retrieve relevant free slots, available to book an appointment on, and matching a set of criteria.

Mandatory and Non Mandatory Attributes

The following table provides the list of mandatory and non-mandatory attributes to create an appointment, including any possible rule conditions and applicable default values.

Mandatory Attributes	Rule
marketSegment	
productSpecification	
address	Either id or href must be filled
category	

Non Mandatory Attributes	Rule
startDate	StartDate is mandatory if weekNumber is empty. If startDate is filled, endDate must be filled
endDate	EndDate is mandatory if weekNumber is empty. If endDate is filled, startDate must be filled
weekNumber	WeekNumber is mandatory if startDate and endDate are empty
favoriteAmpm	
relatedParty	When a party is given, either id or href must be filled
limit	

Default Values Summary

The following table summarizes the default values applicable to optional attributes of the resource (or sub-resources) :

Attributes	Default Value
limit	Value to be defined by the project

Behavior

- Returns HTTP/1.1 status code 200 if the request was successful
- Returns HTTP/1.1 status code 4xx if an error occured, for example to cover these functional error cases :
  - startDate must not be in the past
  - endDate must not be in the past
  - endDate must be superior to startDate

Usage Samples

Request

```
POST /api/freeSlot/search
Content-type: application/json
{
       "marketSegment" : "B2C",
       "favoriteAmpm": "pm",
       "weekNumber": "36",
       "category": "intervention",
       "limit": "5",
       "productSpecification": {
          "id":"productSpec42"
       },
       "address":{
          "id":"7660828",
          "href":"https://host:port/address/address/7660828",
          “description”: “Complete address of the appointment”
       },
       “relatedParty” : {
         “id” : “32”,
         “href”:”https://host:port/partyManagement/individual/32”
      }
}
```

Response

```
200
Content-Type: application/json
{
   “freeSlot”:[
      {
       “relatedParty” : {
         “id”: “32”,
         “href”:”https://host:port/partyManagement/individual/32”
        },
        „startDate”:”2015-09-01T14:00:00.071Z”,
        „endDate”:”2015-09-01T16:00:00.071Z”
      },
      {
       „relatedParty” : {
         “id”:”32”,
         “href”:”https://host:port/partyManagement/individual/32”
        },
        „startDate”:”2015-09-01T16:00:00.071Z”,
        „endDate”:”2015-09-01T18:00:00.071Z”
      },
      {
       „relatedParty” : {
         “id”:”32”,
         “href”:”https://host:port/partyManagement/individual/32”
        },
        „startDate”:”2015-09-01T14:00:00.071Z”,
        „endDate”:”2015-09-01T16:00:00.071Z”
      },
      {
       „relatedParty” : {
         “id”:”32”,
         “href”:”https://host:port/partyManagement/individual/32”
       },
       „startDate”:”2015-09-03T14:00:00.071Z”,
       „endDate”:”2015-09-03T16:00:00.071Z”
      },
      {
       „relatedParty” : {
         “id”:”32”,
         “href”:”https://host:port/partyManagement/individual/32”
        },
        “startDate”:”2015-09-05T14:00:00.071Z”,
        “endDate”:”2015-09-05T16:00:00.071Z”
      }
   ]
}
```

CREATE AN APPOINTMENT (2)

POST /api/appointment

Description

After checking free slots, this operation is used to create an appointment with all its characteristics.

Mandatory and Non Mandatory Attributes

The following table provides the list of mandatory and non-mandatory attributes when creating an appointment, including any possible rule conditions and applicable default values.

Mandatory Attributes	Rule
category	
startDate	
endDate	
address	Either id or href must be filled at least
relatedParty	At least one party must be linked to the appointment (customer, …)
relatedParty.id	Either id or href must be filled at least
relatedParty.href	Either id or href must be filled at least

Non Mandatory Attributes	Rule

externalId	
description	
status	
alarm	
alarmAction	If alarm is false, alarmAction doesn't appear
attachment.href	
address.description	
relatedParty.role	
relatedParty.name	
relatedObject	For example, orderToDeliver, problemToSolve for trouble ticket
relatedObject.involvement	If a relatedObject is selected, involvement and reference must be filled.
relatedObject.reference	If a relatedObject is selected, involvement and reference must be filled.
note	
note.date	
note.author	
note.text	

Default Values Summary

When creating the appointment, the following table summarizes the default values applicable to optional attributes of the resource (or sub-resources) :

Attributes	Default Value
state	initialized

Behavior

- Returns HTTP/1.1 status code 201 if the request was successful
- Returns HTTP/1.1 status code 4xx if an error occurred, for example to cover these functional use cases :
    - startDate must not be in the past
    - endDate must not be in the past
    - endDate must be superior to startDate
    - appointment on a slot already booked
    - state lifecycle is not respected

Usage Samples

Request

```
POST /api/appointment
Content-type: application/json
{
   "externalId":"anExternalIDIfNeeded432113",
   "category":"intervention",
   "description":"A useful text to describe the appointment",
   "state":"missed",
   "creationDate":"2015-09-01T14:40:43.071Z",
   "lastUpdate":"2015-09-01T14:40:43.071Z",
   "startDate":"2015-09-01T14:00:43.071Z",
   "endDate":"2015-09-01T16:00:43.071Z",
   "alarm": true,
   "alarmAction":"smsToCustomer",
   "attachment":[
      {
         "description":"Short description of the document attached to the appointment",
         "href":"http://server/path/document1.pdf"
      }
   ],
   "relatedParty": [
      {
         "id":"32",
         "href":"https://host:port/partyManagement/individual/32",
         "role":"customer",
         "name":"John Doe"
      }
   ],
   "address":{
      "id":"7660828",
       "href" : "https://host:port/address/address/7660828",
       "description": "Complete address of the appointment"
   },
   "relatedObject":[
      {
         "involvement":"problemToSolve",
         "reference": "https://host:port/troubleTicket/troubleTicket/789745465"
      }
   ],
   "note":[
      {
         "date":"2015-09-01T14:40:43.071Z",
         "author":"Arthur Ewans",
         "text":"Already called the expert"
      }

   ]
}
```

Response

```
201
Content-Type: application/json
{
   "id":"21",
   "href":"https://host:port/appointment/appointment/21",
   "externalId":"anExternalIDIfNeeded432113",
   "category":"intervention",
   "description":"A useful text to describe the appointment",
   "state":"missed",
   "creationDate":"2015-09-01T14:40:43.071Z",
   "lastUpdate":"2015-09-01T14:40:43.071Z",
   "startDate":"2015-09-01T14:00:43.071Z",
   "endDate":"2015-09-01T16:00:43.071Z",
   "alarm": true,
   "alarmAction":"smsToCustomer",
   "attachment":[
      {
         "description":"Short description of the document attached to the appointment",
         "href":"http://server/path/document1.pdf"
      }
   ],
   "relatedParty": [
      {
         "id":"32",
         "href":"https://host:port/partyManagement/individual/32",
         "role":"customer",
         "name":"John Doe"
      }
   ],
   "address":{
      "id":"7660828",
       "href":"https://host:port/address/address/7660828",
       "description": "Complete address of the appointment"
   },
   "relatedObject":[
      {
         "involvement":"problemToSolve",
         "reference": "https://host:port/troubleTicket/troubleTicket/789745465"
      }
   ],
   "note":[
      {
         "date":"2015-09-01T14:40:43.071Z",
         "author":"Arthur Ewans",
         "text":"Already called the expert "
      }

   ]
}
```

UPDATE OR CANCEL AN APPOINTMENT (3) (4)

PATCH /api/appointment/{id}

Description

This operation can be used to update partially an appointment if information has changed.
It also can be used to cancel an appointment by modifying its state. The new state is ‘cancelled’.

Patchable and Non Patchable Attributes

The tables below provide the list of patchable and not patchable attributes, including constraint rules on their usage.

Patchable attributes	Rule
category	
description	
state	To manage the appointment process (cf. appointment lifecycle)
startDate	
endDate	
alarm	
alarmAction	
attachment.href	
relatedParty	Only when the party is not the customer
relatedObject	
note	

Non Patchable attributes	Rule
Id	
href	
externalId	
creationDate	
lastUpdate	
address	

Behavior

- Returns HTTP/1.1 status code 200 if the request was successful
- Returns HTTP/1.1 status code 4xx if an error occurred, for example to cover these functional use cases :
    - startDate must not be in the past
    - endDate must not be in the past
    - endDate must be superior to startDate
    - appointment on a slot already booked
    - state lifecycle is not respected

Usage Samples

Request

```
PATCH /api/appointment/21
Content-Type: application/json
{
   "state":"cancelled"
}
```

Response

```
200
Content-Type: application/json
{
   "id":"21",
   "href":"https://host:port/appointment/appointment/21",
   "externalId":"anExternalIDIfNeeded432113",
   "category":"intervention",
   "description":"A useful text to describe the appointment",
   "state":"cancelled",
   "creationDate":"2015-09-01T14:40:43.071Z",
   "lastUpdate":"2015-09-01T14:40:43.071Z",
   "startDate":"2015-09-01T14:00:43.071Z",
   "endDate":"2015-09-01T16:00:43.071Z",
   "alarm": true,
   "alarmAction":"smsToCustomer",
   "attachment":[
      {
         "description":"Short description of the document attached to the appointment",
         "href":"http://server/path/document1.pdf"
      }
   ],
   "relatedParty": [
      {
         "id":"32",
         "href":"https://host:port/partyManagement/individual/32",
         "role":"customer",
         "name":"John Doe"
      }
   ],
   "address":{
      "id":"7660828",
       "href" : "https://host:port/address/address/7660828",
       "description": "Complete address of the appointment"
   },
   "relatedObject":[
      {
         "involvement":"problemToSolve",
         "reference": "https://host:port/troubleTicket/troubleTicket/789745465"
      }
   ],
   "note":[
      {
         "date":"2015-09-01T14:40:43.071Z",
         "author":"Arthur Ewan",
         "text":"Already called the expert "
      }

   ]
}
```

RETRIEVE AN APPOINTMENT

```
GET /api/appointment/{id}
```

Description

This operation is used to search an appointment by its unique identifier.

Behavior

- Returns HTTP/1.1 status code 200 if the request was successful
- Returns HTTP/1.1 status code 404 (Not found) if the appointment does not exist.

Usage Samples

Request

```
GET /api/appointment/21
Accept: application-json
```

Response

```
200
Content-Type: application/json
{
   "id":"21",
   "href":"https://host:port/appointment/appointment/21",
   "externalId":"anExternalIDIfNeeded432113",
   "category":"intervention",
   "description":"A useful text to describe the appointment",
   "state":"missed",
   "creationDate":"2015-09-01T14:40:43.071Z",
   "lastUpdate":"2015-09-01T14:40:43.071Z",
   "startDate":"2015-09-01T14:00:43.071Z",
   "endDate":"2015-09-01T16:00:43.071Z",
   "alarm": true,
   "alarmAction":"smsToCustomer",
   "attachment":[
      {
         "description":"Short description of the document attached to the appointment",
         "href":"http://server/path/document1.pdf"
      }
   ],
   "relatedParty": [
      {
         "id":"32",
         "href":"https://host:port/partyManagement/individual/32",
         "role":"customer",
         "name":"John Doe"
      }
   ],
   "address":{
      "id":"7660828",
       "href" : "https://host:port/address/address/7660828",
       "description": "Complete address of the appointment"
   },
   "relatedObject":[
      {
         "involvement":"problemToSolve",
         "reference": "https://host:port/troubleTicket/troubleTicket/789745465"
      }
   ],
   "note":[
      {
         "date":"2015-09-01T14:40:43.071Z",
         "author":"Arthur Ewan",
         "text":"Already called the expert "
      }

   ]
}
```

LIST APPOINTMENTS (5)

GET /api/appointment? {fields=attributes}&{filtering expression}

Description

This operation is used to retrieve appointments corresponding to given criteria.
Filtering is allowed on all attributes.
Attribute selection is enabled on all attributes.

Behavior

Returns HTTP/1.1 status code 200 if the request was successful
Returns HTTP/1.1 status code 4xx if an error occurred

Usage Samples

Request

```
GET /api/appointment?relatedParty.id=32&relatedParty.role=customer&startDate.gt=2015-08-31&startDate.lt=2015-09-04&fields=id,category,state,startDate,endDate
Accept: application/json
```

Response

```
200
Content-Type: application/json

[
 {
   "id" : "21",
   "category" : "intervention",
   "state":"Validated",
   "startDate" : "2015-09-01T14:00:43.071Z",
   "endDate" : "2015-09-01T16:00:43.071Z"
 }
]
```

RESCHEDULE AN APPOINTMENT (6)

POST /api/freeSlot/search

Description

This operation is used to retrieve relevant free slots, available for rescheduling an existing appointment on.

Mandatory and Non Mandatory Attributes

The following table provides the list of mandatory and non mandatory attributes when creating an appointment, including any possible rule conditions and applicable default values.

Mandatory Attributes	Rule
appointmentId	

Behavior

Returns HTTP/1.1 status code 200 if the request was successful
Returns HTTP/1.1 status code 4xx if an error occurred.

Usage Samples

Request

```
GET /api/freeSlot/search
Content-Type: application/json
{
        "appointmentId": "36"
}
```

Response

```
200
200
Content-Type: application/json

{
   "freeSlot":[
      {
       "relatedParty" : {
         "id" : "32",
         "href" : "https://host:port/partyManagement/individual/32"
       },
       "startDate":"2015-09-01T14:00:00.071Z",
       "endDate":"2015-09-01T16:00:00.071Z"
      },
      {
       "relatedParty" : {
         "id" : "32",
         "href" : "https://host:port/partyManagement/individual/32"
        },
        "startDate":"2015-09-01T16:00:00.071Z",
        "endDate":"2015-09-01T18:00:00.071Z"
      },
      {
       "relatedParty" : {
         "id" : "32",
         "href" : "https://host:port/partyManagement/individual/32"
        },
        "startDate":"2015-09-01T14:00:00.071Z",
        "endDate":"2015-09-01T16:00:00.071Z"
      },
      {
       "relatedParty" : {
         "id" : "32",
         "href" : "https://host:port/partyManagement/individual/32"
        },
        "startDate":"2015-09-03T14:00:00.071Z",
        "endDate":"2015-09-03T16:00:00.071Z"
      },
      {
       "relatedParty" : {
         "id" : "32",
         "href" : "https://host:port/partyManagement/individual/32"
        },
        "startDate":"2015-09-05T14:00:00.071Z",
        "endDate":"2015-09-05T16:00:00.071Z"
      }
   ]
}
```

API NOTIFICATIONS

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/{id}
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.

Behavior

Returns HTTP/1.1 status code 201 if the service is able to set the configuration. The /client/listener url is the callback url passed when registering the listener.

Usage samples

An example of a notification received by the listener. In this example, “EvenType” should be replaced by one of the notification types supported by this API (detailed in Notification resources Model section) and “EVENT BODY” refers to the data structure of the given notification type.

Request

```
POST /client/listener
Accept: application/json

{
  "eventType": "EventType",
   "event": {
     EVENT BODY
   }
}
```

Response

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

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

ACKNOWLEDGMENTS

VERSION HISTORY

Version Number	Date	Modified by	Description
Version 1.0.0	15/04/2016	Pierre Gauthier TM Forum	Final version for Fx16
Version 1.0.1	14/06/2016	Alicja Kawecki TM Forum	Updated cover, header, footer; minor formatting/style corrections prior to publishing for Fx16
Version 1.0.2	05/10/2016	Alicja Kawecki TM Forum	Updated cover and Notice to reflect TM Forum Approved status

RELEASE HISTORY

Release Number	Date	Release led by:	Description
Release 1.0	15/04/2016	Pierre Gauthier TM Forum	First Release of Document

CONTRIBUTORS TO DOCUMENT

This document was prepared by members of the TM Forum API Program team.