

TM Forum Specification

Appointment API REST Specification

TMF646
Release 18.0.0
June 2018

Latest Update: TM Forum Release 18.0.0	Member Evaluation
Version 3.0.1	IPR Mode: RAND

NOTICE

Copyright © TM Forum 2018. All Rights Reserved.

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

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

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

Direct inquiries to the TM Forum office:

4 Century Drive, Suite 100
Parsippany, NJ 07054, USA
Tel No. +1 973 944 5100
Fax No. +1 973 944 5110
TM Forum Web Page: www.tmforum.org

TABLE OF CONTENTS

NOTICE.....	2
TABLE OF CONTENTS	3
LIST OF TABLES	5
INTRODUCTION	6
SAMPLE USE CASES.....	7
RESOURCE MODEL.....	9
Managed Entity and Task Resource Models.....	9
appointment resource	9
search time slot resource.....	17
Notification Resource Models	22
Appointment state change Notification.....	24
Appointment creation Notification.....	24
Appointment attribute value change Notification.....	24
Appointment remove Notification.....	25
Search time slot state change Notification	25
Search time slot creation Notification	26
Search time slot state remove Notification	26
API OPERATIONS.....	27
Operations on Appointment	28
Create an appointment	28
patch an appointment.....	33
Retrieve an appointment	38
List appointments.....	40
Delete an appointment	41
Create a search time slot	42
List search time slots.....	45
Retrieve a search time slot.....	47
patch a search time slot	48
Delete a search time slot.....	50
API NOTIFICATIONS	52

Appointment API REST Specification

Register listener	52
Unregister listener	52
Publish Event to listener	53
ACKNOWLEDGEMENTS	55
DOCUMENT HISTORY	55
Version History	55
Release History	55

LIST OF TABLES

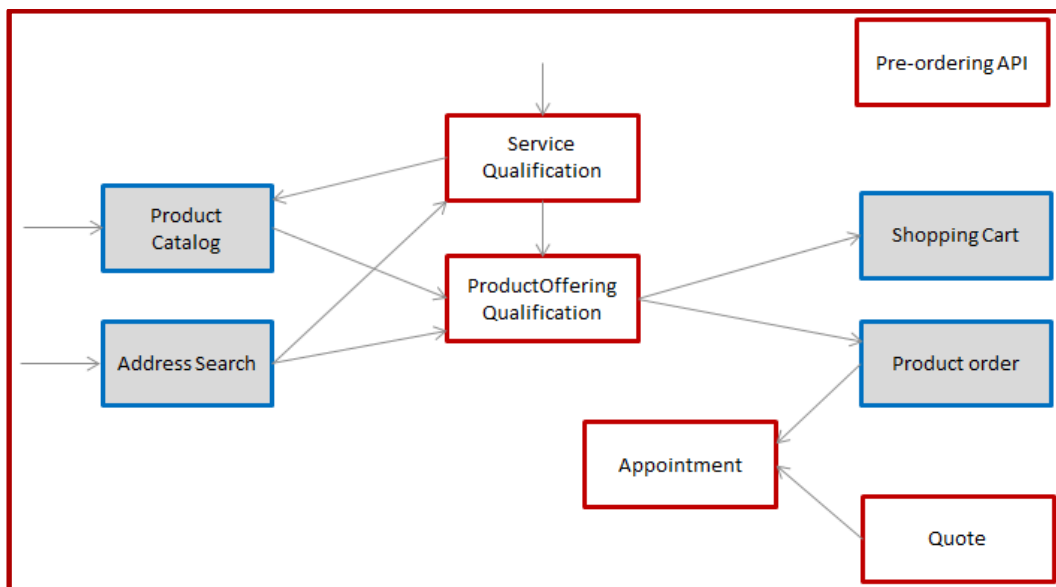
N/A

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.

In the Open Digital Economy where multiple actors (SDPs, CSPs, ...) may be involved with the delivery of an end-to-end service, those actors need to collaborate and interact with the customer as needed.

Following diagram shows Appointment API within pre-ordering API package:



Appointment API is one of Pre-Ordering Management API Family. An Appointment is an arrangement to do something or meet someone at a particular time and place. It is previously made during an interaction or may be necessary to solve a customer problem or to deliver a product order item. The appointment API goal is to manage an appointment with all the necessary characteristics. First, the API consists in searching free time slots based on given parameters. Then, the appointment is created. The appointment has characteristics such as nature of appointment, place of appointment...

Appointment API allows to :

- Retrieve available time slots depending on filter criteria before creating or rescheduling an appointment
- Retrieve an existing appointment or a collection of appointments
- Create an appointment
- Update an appointment to cancel or reschedule it for example
- Delete an appointment

SAMPLE USE CASES

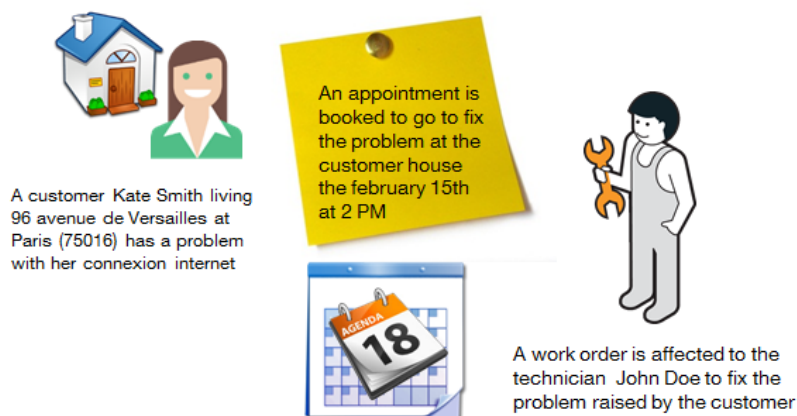
The use cases are :

- Create a new appointment,
- Update an existing appointment because its status has changed (cancel an appointment for example) or party availabilities have changed (reschedule an appointment for example)
- Retrieve a specific appointment or a collection of appointments

Before creating or rescheduling an appointment, the available time slots could be checked according to criteria (like the reference of a specific technician for example,...).

To illustrate the JSON representation of resource concerned by this API, we use the two uses cases detailed below :

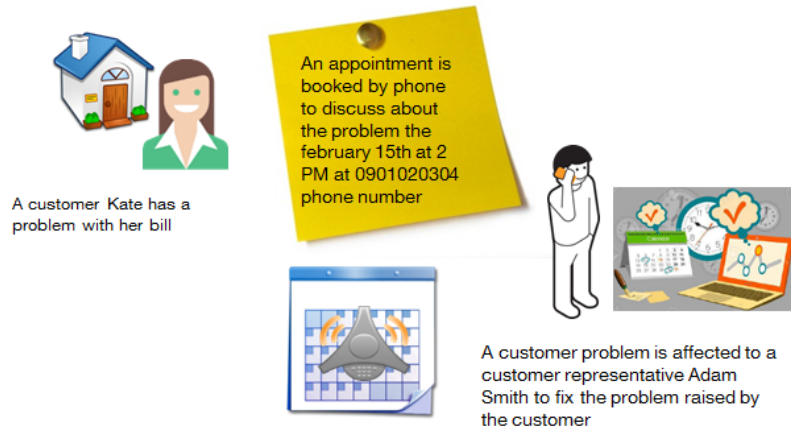
Use case 1 (UC1) : an appointment at a place



Kate wishes that the appointment will be booked between the 13 and 16 of February.

Appointment API REST Specification

Use case 2 (UC2) : an appointment by phone



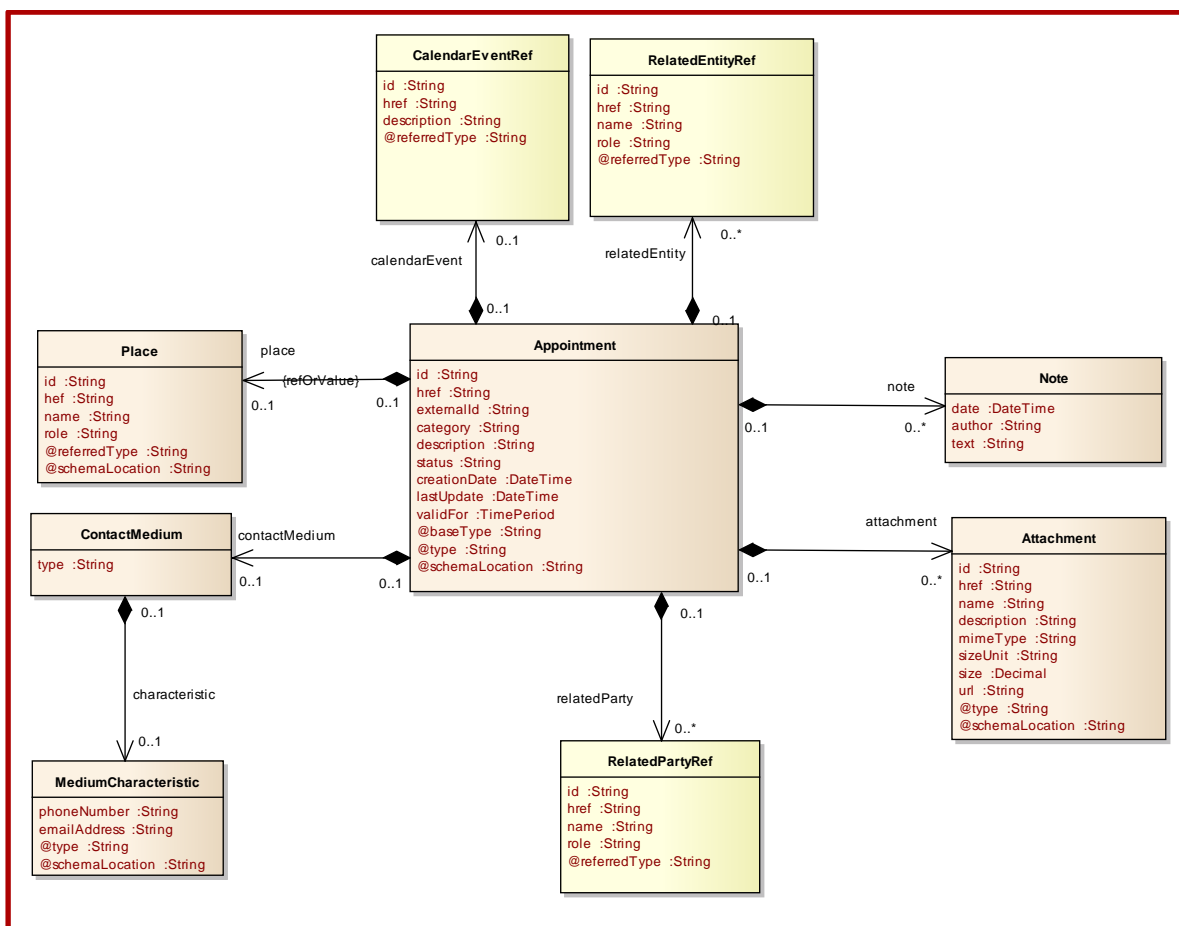
RESOURCE MODEL

Managed Entity and Task Resource Models

APPOINTMENT RESOURCE

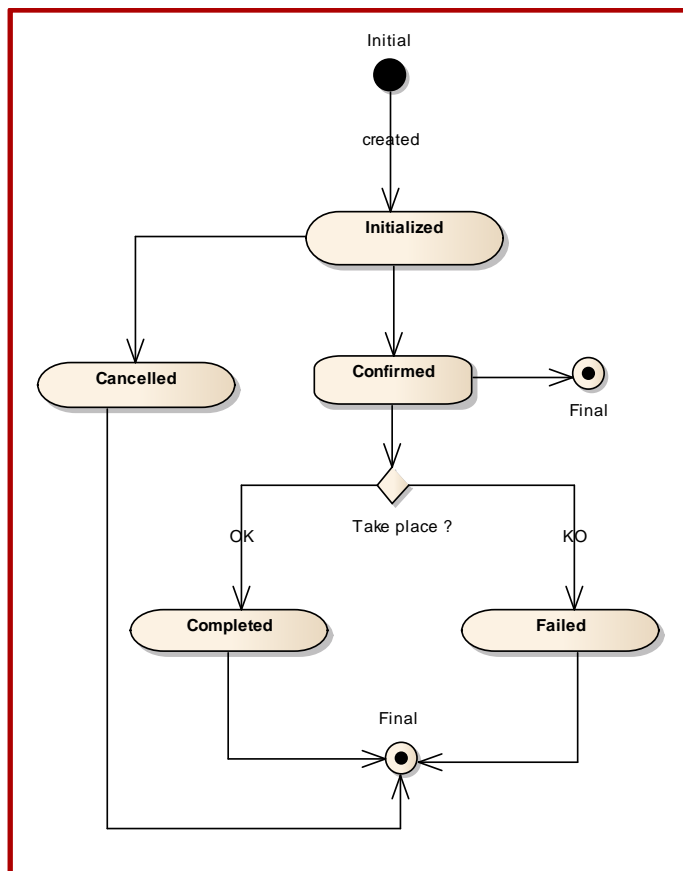
An Appointment is an arrangement to do something or meet someone at a particular time, at a place (for face to face appointment) or through a contact medium (for phone appointment).

Resource model



Lifecycle

The appointment lifecycle is tracked by mean of the status field. The state machine specifying the typical status change transitions is provided below. Note that an implementation of the specification may enrich the list of status depicted in the diagram.



State	Description
initialized	When an appointment is created, the status is 'initialised'
confirmed	When an appointment is confirmed by all parties, the status is 'confirmed'
cancelled	When an appointment is not confirmed by at least one party, it is 'cancelled'
completed	When an appointment took place and it is OK, the status is 'completed'
failed	When an appointment was confirmed and did not take place or was KO, the status is 'failed'

Field descriptions

[Appointment fields](#)

- 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.

Appointment API REST Specification

description	A string. Short free text describing the appointment.
category	A string. Business category : intervention for example or to be more precise afterSalesIntervention, orderDeliveryIntervention,...
status	A string. Lifecycle status of the appointment.
creationDate	A date time (DateTime). Appointment creation date.
lastUpdate	A date time (DateTime). Date of the appointment last update.
validFor	A time period (TimePeriod). Appointment beginning date time and end date time.
@baseType	A string. Indicates the base type of the resource. Here can be appointment.
@type	A string. Indicates the type of resource. Here can be appointment.
@schemaLocation	A string. Link to schema describing this REST resource.
calendarEvent	A calendar event reference (CalendarEventRef). The appointment is associated with a calendar event (an happening at a point of time) associated to a calendar entry.
relatedParty	A list of related party references (RelatedPartyRef [*]). Parties who play a role in the appointment.
relatedEntity	A list of related entity references (RelatedEntityRef [*]). Other resources linked to the appointment. For example, it can be an order item for an order or a work order for a trouble ticket.
attachment	A list of attachment references (Attachment[*]). Information and url related to attached documents to the appointment.
place	A place reference or value (Place). Describes a place for the appointment (using either geographicAddress or GeographicLocation) and/or identify an existing place (either an existing geographicAddress or existing GeographicLocation or existing GeographicSite) through its id
contactMedium	A contact medium (ContactMedium). The contact medium can be a phone number for example for an appointment by phone on a conference call system.
note	A list of notes (Note [*]). Extra information about the appointment.

Appointment API REST Specification

[Attachment sub-resource](#)

Attached document to the appointment.

id	A string. Unique identifier of the attached document to the appointment.
href	A string. Reference of the attached document to the appointment.
name	A string. Name of the attached document.
description	A string. A short description of the attached document.
contentType	A string. The mime type of the attached document.
sizeUnit	A string. Unit of measure used for the size
size	A decimal. The size in Bytes of the document or attachment. If this component contains the embedded data then the size is the size of the embedded data; if it is a reference without the data then it is the size of the referenced document.
url	A string. The URL of the attached document.
@type	A string. Indicates the (class) type of the attachment.
@schemaLocation	A string. Link to schema describing this REST resource.

[Note sub-resource](#)

Extra information about the appointment.

date	A date time (DateTime). Date of the note.
author	A string. Author of the note.
text	A string. Text of the note.

[ContactMedium sub-resource](#)

Indicates the contact medium that could be used for the appointment.

type	A string. Type of the contact medium : phone number or email address.
characteristic	A medium characteristic (MediumCharacteristic). Describes the characteristics of the contact medium used for the appointment.

[MediumCharacteristic sub-resource](#)

Appointment API REST Specification

Describes the characteristics of the contact medium used for the appointment.

phoneNumber	A string. The primary phone number of the contact.
emailAddress	A string. Full email address in standard format.
@type	A string. Indicates the (class) type of the medium characteristic.
@schemaLocation	A string. Link to the schema describing this REST resource.

[Place sub-resource](#)

Describe a place for the appointment (using either geographicAddress or GeographicLocation) and/or identify an existing place (either an existing geographicAddress or existing GeographicLocation or existing GeographicSite) through its id.

id	A string. Unique identifier of a place - Must be a geographicAddress id or GeographicLocation id or GeographicSite id.
href	A string. Reference of a place.
name	A string. Short text describing the place.
role	A string. Role of the place - could be an "installation place" for example.
@referredType	A string. Indicates the (class) type of place.
@schemaLocation	A string. A link to the schema describing this REST resource.

[CalendarEventRef sub-resource](#)

Event in a calendar associated to the appointment.

id	A string. Unique identifier of the calendar event.
href	A string. Reference of the calendar event.
description	A string. A short text describing the calendar event for this appointment.
@referredType	A string. Indicates the (class) type of the calendar event.

[RelatedEntityRef sub-resource](#)

Related entity reference. Related entity linked to the appointment.

Appointment API REST Specification

id	A string. Unique identifier of the related entity.
href	A string. Reference of the related entity.
name	A string. Name of the related entity.
role	A string. Role played by the related entity.
@referredType	A string. Indicates the (class) type of the entity.

[RelatedPartyRef sub-resource](#)

Related party reference. A related party defines a party linked to the appointment.

id	A string. Unique identifier of a related party.
href	A string. Reference of the related party.
name	A string. Name of the related party.
role	A string. Role played by the related party (customer, technician,... for example)
@referredType	A string. Indicates the (class) type of the entity.

Json representation sample

We provide below the json representation of an "Appointment" resource object for UC1:

```
{
  "id": "21",
  "href": "https://host:port/appointment/appointment/21",
  "externalId": "432113",
  "category": "intervention",
  "description": "Fix an internet connexion problem for a customer",
  "status": "accepted",
  "creationDate": "2018-02-01T14:40:43.071Z",
  "lastUpdate": "2018-02-04T14:40:43.071Z",
  "validFor": {
    "startDateTime": "2018-02-15T14:00:00.071Z",
    "endDateTime": "2018-02-15T16:00:00.071Z"
  },
  "attachment": [
    {
      "id": "4444",
      "href": "https://host:port/documentManagement/attachment/4444",
      "name": "ContractForCustomer32",
      "mimeType": "application/pdf",
    }
  ]
}
```

Appointment API REST Specification

```
"sizeUnit": "bytes",
"size": 135,
"url": "https://DocumentManager/4444/CustomerContract32.pdf",
"@type": "customerContract"
}
],
"calendarEvent": {
  "id": "33",
  "href": "https://host:port/timeManagement/calendarEvent/33",
  "description": "Outlook calendar event"
},
"relatedParty": [
  {
    "id": "32",
    "href": "https://host:port/partyManagement/individual/32",
    "name": "Kate Smith",
    "role": "customer"
  },
  {
    "id": "56",
    "href": "https://host:port/partyManagement/individual/56",
    "name": "John Doe",
    "role": "technician"
  }
],
"place": {
  "role": "interventionAddress",
  "@referredType": "geographicAddress",
  "geographicAddress": {
    "streetNr": "96",
    "streetType": "avenue",
    "streetName": "de Versailles",
    "postCode": "75016",
    "city": "Paris",
    "country": "France"
  }
},
"relatedEntity": [
  {
    "id": "789",
    "href": "https://host:port/troubleTicket/troubleTicket/789",
    "name": "Internet connexion problem to solve",
    "@referredType": "troubleTicket"
  },
  {
    "id": "465",
    "href": "https://host:port/workForceManagement/workOrder/465",
    "name": "Work order generated for technician intervention",
    "@referredType": "workOrder"
  }
],
```

Appointment API REST Specification

```

"note":[
  {
    "date":"2018-02-04T11:00:18.071Z",
    "author":"Ludovic Robert",
    "text":"Problem explained to the technician"
  },
  {
    "date":"2018-02-04T14:40:43.071Z",
    "author":"Ludovic Robert",
    "text":"Work order generated"
  }
]
}

```

We provide below the json representation of an “Appointment” resource object for UC2:

```

{
  "id":"22",
  "href":"https://host:port/appointment/appointment/22",
  "externalId":"432114",
  "category":"customerProblem",
  "description":"Customer bill problem explanation",
  "status": "accepted",
  "creationDate":"2018-02-01T14:40:43.071Z",
  "lastUpdate": "2018-02-04T14:40:43.071Z",
  "validFor" : {
    "startDateTime":"2018-02-15T14:00:00.071Z",
    "endDateTime":"2018-02-15T15:00:00.071Z"
  },
  "attachment":[
    {
      "id":"123456",
      "href":"https://host:port/customerBillManagement/customerBill/123456",
      "name":"CustomerBill123456",
      "mimeType":"application/pdf",
      "sizeUnit":"bytes",
      "size": 135,
      "url":"https://DocumentManager/123456/CustomerBill123456.pdf",
      "@type":"billDocument"
    }
  ],
  "calendarEvent": {
    "id":"33",
    "href":" https://host:port/timeManagement/calendarEvent/33",
    "description":"Outlook calendar event"
  },
  "relatedParty": [
    {

```

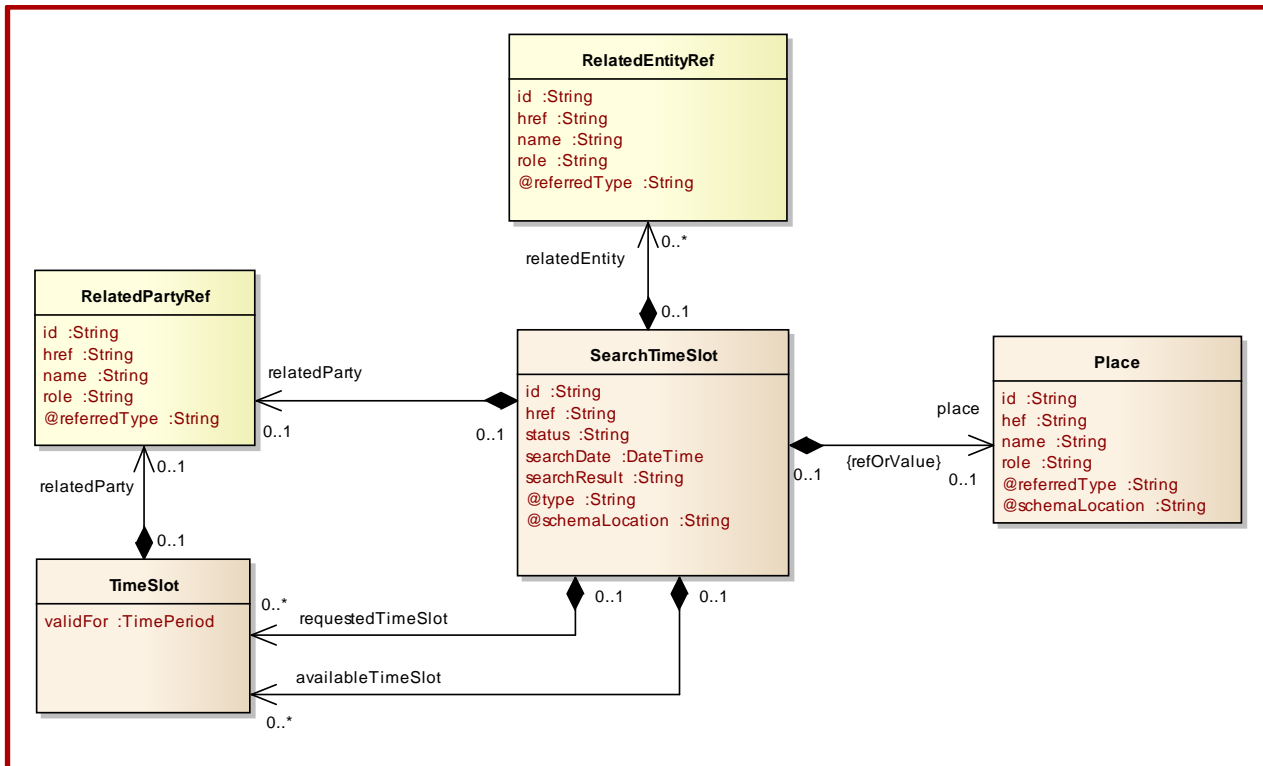


```
"id": "32",
"href": "https://host:port/partyManagement/individual/32",
"name": "Kate Smith",
"role": "customer"
},
{
  "id": "62",
  "href": "https://host:port/partyManagement/individual/62",
  "name": "Adam Smith",
  "role": "customerRepresentative"
}
],
"contactMedium": {
  "type": "phoneNumber",
  "characteristic": {
    "phoneNumber": "0901020304"
  }
},
"relatedEntity": [
  {
    "id": "789",
    "href": "https://host:port/troubleTicket/troubleTicket/789",
    "name": "Customer bill amount disputed by the customer",
    "@referredType": "troubleTicket"
  }
],
"note": [
  {
    "date": "2018-02-04T11:00:18.071Z",
    "author": "Ludovic Robert",
    "text": "Customer bill amount analyzed – reason found"
  }
]
}
```

SEARCH TIME SLOT RESOURCE

This resource is used to retrieve available time slots. One of this available time slot is after used to create or reschedule an appointment.

Resource model



Lifecycle

The search time slot lifecycle is tracked by mean of the status field. Typical lifecycle values are:

- rejected: task request is rejected
- inProgress: task execution has begun and it is in progress
- done: task has been done
- terminatedWithError: task has not been executed correctly and no response is provided

Note that an implementation of the specification may enrich this list of status.

Field descriptions

[SearchTimeSlot fields](#)

id	A string. Unique identifier of the search time slot request
href	A string. Reference to access the search time slot resource
status	A string. Status of the search time slot (rejected, inProgress, done or terminatedWithError)
searchDate	A date time (DateTime). Date when the search time slot is performed.

Appointment API REST Specification

searchResult	A string. Result of the search time slot (success or fail for example)
@type	A string. Indicates the type of resource.
@schemaLocation	A string. Link to schema describing this REST resource.
relatedParty	A party reference (RelatedPartyRef). Party requested for this appointment, owner of the calendar on which time slots are checked. Only in the request.
relatedEntity	A list of related entity references (RelatedEntity [*]). Other resources linked to the appointment. For example, it can be an order item to deliver for an order or a problem to solve for a trouble ticket. Only in the request.
place	A place reference or value (Place). Describes a place for the appointment (using either geographicAddress or GeographicLocation) and/or identify an existing place (either an existing geographicAddress or existing GeographicLocation or existing GeographicSite) through its id. Only in the request.
requestedTimeSlot	A list of requested time slots (TimeSlot[*]). Date time for appointment beginning date and end date given in the response. Only in the request.
availableTimeSlot	A list of available time slots (TimeSlot[*]). Date time for appointment beginning date and end date. Only in the response.

[Place sub-resource](#)

Describe a place for the appointment (using either geographicAddress or GeographicLocation) and/or identify an existing place (either an existing geographicAddress or existing GeographicLocation or existing GeographicSite) through its id.

id	A string. Unique identifier of a place - Must be a geographicAddress id or GeographicLocation id or GeographicSite id.
href	A string. Reference of a place.
name	A string. Short text describing the place.
role	A string. Role of the place - could be "installation place" for example.
@referredType	A string. Indicates the (class) type of place.

Appointment API REST Specification

@schemaLocation A string. A link to the schema describing this REST resource.

[RelatedEntityRef sub-resource](#)

Related entity reference. Related entity linked to the appointment.

id A string. Unique identifier of the related entity.

href A string. Reference of the related entity.

name A string. Name of the related entity.

role A string. Role played by the related entity.

@referredType A string. Indicates the (class) type of the entity.

[RelatedPartyRef sub-resource](#)

RelatedParty reference. A related party defines party or party role linked to the appointment.

id A string. Unique identifier of a related party.

href A string. Reference of the related party.

name A string. Name of the related party.

role A string. Role played by the related party.

@referredType A string. Indicates the (class) type of the entity.

[TimeSlot sub-resource](#)

Requested time slots and available time slots.

validFor A time period (TimePeriod). Beginning date time and end date time of the time slot.

relatedParty A party reference (RelatedPartyRef). Party available during this time slot (for example an intervention team).

Json representation sample

We provide below the json representation of the search time slot request for UC1 (we want to know the available slots in the calendar of the technician John Doe between the 15 and 18 of February to make an intervention at a place):

Appointment API REST Specification

```

{
  "requestedTimeSlot": [
    {
      "validFor": {
        "startDateTime": "2018-02-15T10:00:00.071Z",
        "endDateTime": "2018-02-18T00:00:00.071Z"
      }
    }
  ],
  "relatedParty": {
    "id": "56",
    "href": "https://host:port/partyManagement/individual/56",
    "name": "John Doe",
    "role": "technician"
  },
  "place": {
    "role": "interventionAddress",
    "@referredType": "geographicAddress",
    "geographicAddress": {
      "streetNr": "96",
      "streetType": "avenue",
      "streetName": "de Versailles",
      "postCode": "75016",
      "city": "Paris",
      "country": "France"
    }
  },
  "relatedEntity": [
    {
      "id": "789",
      "href": "https://host:port/troubleTicket/troubleTicket/789",
      "name": "Internet connexion problem to solve",
      "@referredType": "troubleTicket"
    }
  ]
}

```

We provide below the json representation of the search time slot response for UC1 (a list of two free slots are available for the requested technician John Doe and one is available for another suggested technician Adam Smith):

```

{
  "id": "99",
  "href": "https://host:port/appointment/searchTimeSlot/99",

```

Appointment API REST Specification

```
"status": "done",
"searchDate": "2018-02-01T14:40:43.071Z",
"searchResult": "success",
"availableSlot": [
  {
    "validFor": {
      "startDateTime": "2018-02-15T14:00:00.071Z",
      "endDateTime": "2018-02-15T16:00:00.071Z"
    },
    "relatedParty": {
      "id": "56",
      "href": "https://host:port/partyManagement/individual/56",
      "name": "John Doe",
      "role": "technician"
    }
  },
  {
    "validFor": {
      "startDateTime": "2018-02-16T16:30:00.071Z",
      "endDateTime": "2018-02-16T18:00:00.071Z"
    },
    "relatedParty": {
      "id": "56",
      "href": "https://host:port/partyManagement/individual/56",
      "name": "John Doe",
      "role": "technician"
    }
  },
  {
    "validFor": {
      "startDateTime": "2018-02-15T16:00:00.071Z",
      "endDateTime": "2018-02-15T17:00:00.071Z"
    },
    "relatedParty": {
      "id": "58",
      "href": "https://host:port/partyManagement/individual/58",
      "name": "Adam Smith",
      "role": "technician"
    }
  }
]
}
```

Notification Resource Models

Five notifications are defined for this API.

Appointment API REST Specification

Notifications related to Appointment:

- AppointmentStateChangeNotification
- AppointmentCreationNotification
- AppointmentAttributeValueChangeNotification
- AppointmentRemoveNotification

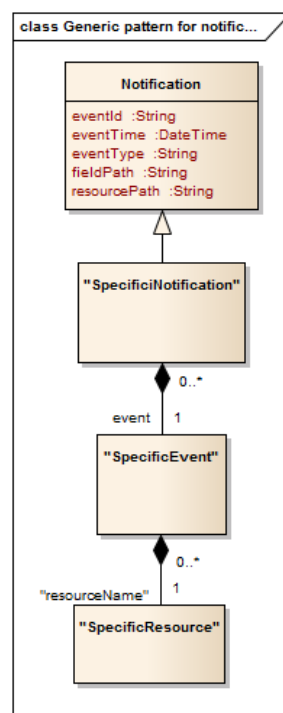
Notifications related to SearchTimeSlot:

- SearchTimeSlotStateChangeNotification
- SearchTimeSlotCreationNotification
- SearchTimeSlotRemoveNotification

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

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

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



APPOINTMENT STATE CHANGE NOTIFICATION

Notification sent when the state of an appointment is updated.

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2018-02-16T16:30:00.071Z",
  "eventType":"AppointmentStateChangeNotification",
  "event": {
    "appointment" :
      { -- SEE Appointment RESOURCE SAMPLE -- }
  }
}
```

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":"2018-02-16T16:30:00.071Z",
  "eventType":"AppointmentCreationNotification",
  "event": {
    "appointment" :
      { -- 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":"2018-02-16T16:30:00.071Z",
  "eventType":"AppointmentAttributeValueChangeNotification",
  "event": {
    "appointment" :
      { -- SEE Appointment RESOURCE SAMPLE -- }
  }
}
```

APPOINTMENT REMOVE NOTIFICATION

Notification sent when an appointment is removed.

Json representation samples

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

```
{
  "eventId":"00001",
  "eventTime":"2018-02-16T16:30:00.071Z",
  "eventType":"AppointmentRemoveNotification",
  "event": {
    "appointment" :
      { -- SEE Appointment RESOURCE SAMPLE -- }
  }
}
```

SEARCH TIME SLOT STATE CHANGE NOTIFICATION

Notification sent when the state of a search time slot is updated.

Json representation sample

We provide below the json representation of an example of a ‘SearchTimeSlotStateChangeNotification’ notification object

```
{
  "eventId":"00001",
  "eventTime":"2018-02-16T16:30:00.071Z",
  "eventType":"SearchTimeSlotStateChangeNotification",
  "event": {
    "searchTimeSlot" :
      { -- SEE SearchTimeSlot RESOURCE SAMPLE -- }
  }
}
```

SEARCH TIME SLOT CREATION NOTIFICATION

Notification sent when a new search time slot is created.

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2018-02-16T16:30:00.071Z",
  "eventType":"SearchTimeSlotCreationNotification",
  "event": {
    "searchTimeSlot" :
    { -- SEE SearchTimeSlot RESOURCE SAMPLE -- }
  }
}
```

SEARCH TIME SLOT STATE REMOVE NOTIFICATION

Notification sent when a search time slot is removed.

Json representation sample

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

```
{
  "eventId":"00001",
  "eventTime":"2018-02-16T16:30:00.071Z",
  "eventType":"SearchTimeSlotRemoveNotification",
  "event": {
    "searchTimeSlot" :
    { -- SEE SearchTimeSlot 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
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

CREATE AN APPOINTMENT

POST /appointment

Description

This operation is used to create an appointment.
A prerequisite is to choose an available time slot before.

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. Notice that it is up to an implementer to add additional mandatory attributes.

Mandatory Attributes	Rule
validFor	A validFor.startDateTime and validFor.endDateTime must be provided validFor.startDateTime must not be in the past validFor.endDateTime must not be in the past validFor.endDateTime must be superior to validFor.startDateTime

Non Mandatory Attributes	Rule
externalId	
description	
status	only filled on server side
category	
creationDate	only filled on server side
lastUpdate	only filled on server side
calendarEvent	
relatedParty	At least one party is provided (for which the appointment is booked) Can be optional if relatedParty is not known and can be assigned later on. Anonymous appointments where never a related party will be assigned can also be possible.
relatedEntity	
attachment	

Appointment API REST Specification

Non Mandatory Attributes	Rule
place	
contactMedium	
note	

Additional Rules

The following table provides additional rules indicating mandatory fields in sub-resources or relationships when creating an Appointment resource.

Context	Mandatory Sub-Attributes
relatedParty	role + id OR href
relatedEntity	@referredType and id OR href
calendarEvent	id OR href

Default Values Summary

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

Attributes	Default Value
status	initialized

Usage Samples

Request
<pre> POST /appointment/appointment Content-Type: application/json { "externalId":"432113", "category":"intervention", "description":"Fix an internet connexion problem for a customer", "validFor" : { "startDateTime":"2018-02-15T14:00:00.071Z", "endDateTime":"2018-02-15T16:00:00.071Z" }, "attachment":[{ "id":"4444", "href":"https://host:port/documentManagement/attachment/4444", "name":"ContractForCustomer32", "mimeType":"application/pdf", </pre>

Appointment API REST Specification

```

    "sizeUnit":"bytes",
    "size": 135,
    "url":"https://DocumentManager/4444/CustomerContract32.pdf",
    "@type":"customerContract"
  }
],
"calendarEvent": {
  "id":"33",
  "href":" https://host:port/timeManagement/calendarEvent/33",
  "description":"Outlook calendar event"
},
"relatedParty": [
  {
    "id":"32",
    "href":"https://host:port/partyManagement/individual/32",
    "name":"Kate Smith",
    "role":"customer"
  },
  {
    "id":"56",
    "href":"https://host:port/partyManagement/individual/56",
    "name":"John Doe",
    "role":"technician"
  }
],
"place":{
  "role":"interventionAddress",
  "@referredType" : "geographicAddress",
  "geographicAddress":{
    "streetNr":"96",
    "streetType": "avenue",
    "streetName": "de Versailles",
    "postCode": "75016",
    "city": "Paris",
    "country": "France"
  }
},
"relatedEntity":[
  {
    "id":"789",
    "href": "https://host:port/troubleTicket/troubleTicket/789",
    "name": "Internet connexion problem to solve",
    "@referredType": "troubleTicket"
  },
  {
    "id":"465",
    "href": "https://host:port/workForceManagement/workOrder/465",
    "name": "Work order generated for technician intervention",
    "@referredType": "workOrder"
  }
],

```

Appointment API REST Specification

```

"note":[
  {
    "date":"2018-02-04T11:00:18.071Z",
    "author":"Ludovic Robert",
    "text":"Problem explained to the technician"
  },
  {
    "date":"2018-02-04T14:40:43.071Z",
    "author":"Ludovic Robert",
    "text":"Work order generated"
  }
]
}

```

Response

```

201
Content-Type: application/json
{
  "id":"21",
  "href":"https://host:port/appointment/appointment/21",
  "externalId":"432113",
  "category":"intervention",
  "description":"Fix an internet connexion problem for a customer",
  "status": "accepted",
  "creationDate":"2018-02-01T14:40:43.071Z",
  "lastUpdate": "2018-02-04T14:40:43.071Z",
  "validFor" : {
    "startDateTime":"2018-02-15T14:00:00.071Z",
    "endDateTime":"2018-02-15T16:00:00.071Z"
  },
  "attachment":[
    {
      "id":"4444",
      "href":"https://host:port/documentManagement/attachment/4444",
      "name":"ContractForCustomer32",
      "mimeType":"application/pdf",
      "sizeUnit":"bytes",
      "size": 135,
      "url":"https://DocumentManager/4444/CustomerContract32.pdf",
      "@type":"customerContract"
    }
  ],
  "calendarEvent": {
    "id":"33",
    "href":" https://host:port/timeManagement/calendarEvent/33",
    "description":"Outlook calendar event"
  },
  "relatedParty": [

```

Appointment API REST Specification

```
{
  "id": "32",
  "href": "https://host:port/partyManagement/individual/32",
  "name": "Kate Smith",
  "role": "customer"
},
{
  "id": "56",
  "href": "https://host:port/partyManagement/individual/56",
  "name": "John Doe",
  "role": "technician"
}
],
"place": {
  "role": "interventionAddress",
  "@referredType": "geographicAddress",
  "geographicAddress": {
    "streetNr": "96",
    "streetType": "avenue",
    "streetName": "de Versailles",
    "postCode": "75016",
    "city": "Paris",
    "country": "France"
  }
},
"relatedEntity": [
  {
    "id": "789",
    "href": "https://host:port/troubleTicket/troubleTicket/789",
    "name": "Internet connexion problem to solve",
    "@referredType": "troubleTicket"
  },
  {
    "id": "465",
    "href": "https://host:port/workForceManagement/workOrder/465",
    "name": "Work order generated for technician intervention",
    "@referredType": "workOrder"
  }
],
"note": [
  {
    "date": "2018-02-04T11:00:18.071Z",
    "author": "Ludovic Robert",
    "text": "Problem explained to the technician"
  },
  {
    "date": "2018-02-04T14:40:43.071Z",
    "author": "Ludovic Robert",
    "text": "Work order generated"
  }
]
]
```


}

PATCH AN APPOINTMENT

PATCH /appointment/{id}

Description

This operation allows partial updates of an appointment entity. It could be used to cancel an appointment (change of the status value) or to reschedule an appointment (change of the validFor and/or relatedParty).

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 not patchable attributes, including constraint rules on their usage.

Patchable attributes	Rule
category	
description	
status	To manage the appointment process (cf. appointment lifecycle)
validFor	
attachment	
relatedParty	Only when the party is not the customer
relatedEntity	
note	
place	
contactMedium	
calendarEvent	

Non Patchable attributes	Rule
id	
href	
externalId	

Appointment API REST Specification

Non Patchable attributes	Rule
creationDate	
lastUpdate	

Usage Samples

Sample for the cancellation of an existing appointment :

Request
PATCH /appointment/appointment/21 Content-Type: application/json <pre>{ "status": "cancelled" }</pre>
Response
200 Content-Type: application/json <pre>{ "id": "21", "href": "https://host:port/appointment/appointment/21", "externalId": "432113", "category": "intervention", "description": "Fix an internet connexion problem for a customer", "status": "cancelled", "creationDate": "2018-02-01T14:40:43.071Z", "lastUpdate": "2018-02-04T14:40:43.071Z", "validFor" : { "startDateTime": "2018-02-15T14:00:00.071Z", "endDateTime": "2018-02-15T16:00:00.071Z" }, "attachment": [{ "id": "4444", "href": "https://host:port/documentManagement/attachment/4444", "name": "ContractForCustomer32", "mimeType": "application/pdf", "sizeUnit": "bytes", "size": 135, "url": "https://DocumentManager/4444/CustomerContract32.pdf", "@type": "customerContract" }], }</pre>

Appointment API REST Specification

```
"calendarEvent": {
  "id": "33",
  "href": "https://host:port/timeManagement/calendarEvent/33",
  "description": "Outlook calendar event"
},
"relatedParty": [
  {
    "id": "32",
    "href": "https://host:port/partyManagement/individual/32",
    "name": "Kate Smith",
    "role": "customer"
  },
  {
    "id": "56",
    "href": "https://host:port/partyManagement/individual/56",
    "name": "John Doe",
    "role": "technician"
  }
],
"place": {
  "role": "interventionAddress",
  "@referredType": "geographicAddress",
  "geographicAddress": {
    "streetNr": "96",
    "streetType": "avenue",
    "streetName": "de Versailles",
    "postCode": "75016",
    "city": "Paris",
    "country": "France"
  }
},
"relatedEntity": [
  {
    "id": "789",
    "href": "https://host:port/troubleTicket/troubleTicket/789",
    "name": "Internet connexion problem to solve",
    "@referredType": "troubleTicket"
  },
  {
    "id": "465",
    "href": "https://host:port/workForceManagement/workOrder/465",
    "name": "Work order generated for technician intervention",
    "@referredType": "workOrder"
  }
],
"note": [
  {
    "date": "2018-02-04T11:00:18.071Z",
    "author": "Ludovic Robert",
    "text": "Problem explained to the technician"
  }
],
```

Appointment API REST Specification

```

{
  "date":"2018-02-04T14:40:43.071Z",
  "author":"Ludovic Robert",
  "text":"Work order generated"
}
]
}

```

Sample for the rescheduling of an existing appointment :

Request

```

PATCH /appointment/appointment/21
Content-Type: application/json
{
  "validFor" : {
    "startDateTime":"2018-02-16T14:00:00.071Z",
    "endDateTime":"2018-02-16T16:00:00.071Z"
  },
  "relatedParty": [
    {
      "id":"58",
      "href":"https://host:port/partyManagement/individual/58",
      "name":"Adam Smith",
      "role":"technician"
    }
  ]
}

```

Response

```

200
Content-Type: application/json
{
  "id":"21",
  "href":"https://host:port/appointment/appointment/21",
  "externalId":"432113",
  "category":"intervention",
  "description":"Fix an internet connexion problem for a customer",
  "status": "accepted",
  "creationDate":"2018-02-01T14:40:43.071Z",
  "lastUpdate": "2018-02-04T14:40:43.071Z",
  "validFor" : {
    "startDateTime":"2018-02-16T14:00:00.071Z",

```

Appointment API REST Specification

```

    "endTime": "2018-02-16T16:00:00.071Z"
  },
  "attachment": [
    {
      "id": "4444",
      "href": "https://host:port/documentManagement/attachment/4444",
      "name": "ContractForCustomer32",
      "mimeType": "application/pdf",
      "sizeUnit": "bytes",
      "size": 135,
      "url": "https://DocumentManager/4444/CustomerContract32.pdf",
      "@type": "customerContract"
    }
  ],
  "calendarEvent": {
    "id": "33",
    "href": "https://host:port/timeManagement/calendarEvent/33",
    "description": "Outlook calendar event"
  },
  "relatedParty": [
    {
      "id": "32",
      "href": "https://host:port/partyManagement/individual/32",
      "name": "Kate Smith",
      "role": "customer"
    },
    {
      "id": "99",
      "href": "https://host:port/partyManagement/individual/99",
      "name": "Adam Smith",
      "role": "technician"
    }
  ],
  "place": {
    "role": "interventionAddress",
    "@referredType": "geographicAddress",
    "geographicAddress": {
      "streetNr": "96",
      "streetType": "avenue",
      "streetName": "de Versailles",
      "postCode": "75016",
      "city": "Paris",
      "country": "France"
    }
  },
  "relatedEntity": [
    {
      "id": "789",
      "href": "https://host:port/troubleTicket/troubleTicket/789",
      "name": "Internet connexion problem to solve",
      "@referredType": "troubleTicket"
    }
  ]
}

```

```

    },
    {
      "id": "465",
      "href": "https://host:port/workForceManagement/workOrder/465",
      "name": "Work order generated for technician intervention",
      "@referredType": "workOrder"
    }
  ],
  "note": [
    {
      "date": "2018-02-04T11:00:18.071Z",
      "author": "Ludovic Robert",
      "text": "Problem explained to the technician"
    },
    {
      "date": "2018-02-04T14:40:43.071Z",
      "author": "Ludovic Robert",
      "text": "Work order generated"
    }
  ]
}

```

RETRIEVE AN APPOINTMENT

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

Description

This operation retrieves an appointment 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

Request
GET /appointment/appointment/21 Accept : application/json
Response
200

Appointment API REST Specification

```

Content-Type: application/json
{
  "id": "21",
  "href": "https://host:port/appointment/appointment/21",
  "externalId": "432113",
  "category": "intervention",
  "description": "Fix an internet connexion problem for a customer",
  "status": "accepted",
  "creationDate": "2018-02-01T14:40:43.071Z",
  "lastUpdate": "2018-02-04T14:40:43.071Z",
  "validFor": {
    "startDateTime": "2018-02-15T14:00:00.071Z",
    "endDateTime": "2018-02-15T16:00:00.071Z"
  },
  "attachment": [
    {
      "id": "4444",
      "href": "https://host:port/documentManagement/attachment/4444",
      "name": "ContractForCustomer32",
      "mimeType": "application/pdf",
      "sizeUnit": "bytes",
      "size": 135,
      "url": "https://DocumentManager/4444/CustomerContract32.pdf",
      "@type": "customerContract"
    }
  ],
  "calendarEvent": {
    "id": "33",
    "href": "https://host:port/timeManagement/calendarEvent/33",
    "description": "Outlook calendar event"
  },
  "relatedParty": [
    {
      "id": "32",
      "href": "https://host:port/partyManagement/individual/32",
      "name": "Kate Smith",
      "role": "customer"
    },
    {
      "id": "56",
      "href": "https://host:port/partyManagement/individual/56",
      "name": "John Doe",
      "role": "technician"
    }
  ],
  "place": {
    "role": "interventionAddress",
    "@referredType": "geographicAddress",
    "geographicAddress": {
      "streetNr": "96",
      "streetType": "avenue",

```

```

    "streetName": "de Versailles",
    "postCode": "75016",
    "city": "Paris",
    "country": "France"
  }
},
"relatedEntity":[
  {
    "id":"789",
    "href": "https://host:port/troubleTicket/troubleTicket/789",
    "name": "Internet connexion problem to solve",
    "@referredType": "troubleTicket"
  },
  {
    "id":"465",
    "href": "https://host:port/workForceManagement/workOrder/465",
    "name": "Work order generated for technician intervention",
    "@referredType": "workOrder"
  }
],
"note":[
  {
    "date":"2018-02-04T11:00:18.071Z",
    "author":"Ludovic Robert",
    "text":"Problem explained to the technician"
  },
  {
    "date":"2018-02-04T14:40:43.071Z",
    "author":"Ludovic Robert",
    "text":"Work order generated"
  }
]
}

```

LIST APPOINTMENTS

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

Description

This operation list appointment 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

Request
<pre>GET /appointment/appointment?relatedParty.id=32&relatedParty.role=customer&fields=id,category,status,validFor Accept: application/json</pre>
Response
<pre>200 Content-Type : application/json [{ "id": "21", "category": "intervention", "status": "accepted", "validFor" : { "startDateTime": "2018-02-15T14:00:00.071Z", "endDateTime": "2018-02-15T16:00:00.071Z" } }, { "id": "22", "category": "customerProblem", "status": "accepted", "validFor" : { "startDateTime": "2018-02-15T14:00:00.071Z", "endDateTime": "2018-02-15T15:00:00.071Z" } }]</pre>

DELETE AN APPOINTMENT

DELETE /appointment/{id}

Note : this operation is available only to ADMIN API users.

Description

This operation deletes an appointment entity.

Usage Samples

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

Request
DELETE /appointment/appointment/21
Response
204

CREATE A SEARCH TIME SLOT

POST /searchTimeSlot**Description**

This operation creates a search time slot entity. It is used to retrieve, according to a set of criteria, available time slots used after to book or reschedule an appointment on.

The expected API behavior is the following :

- 1 – API requester gives a list a requested time slots, for a related party (optional) and some information to define the context of the appointment like the place or the reference of other resources linked to the appointment.
- 2 – A search time slot is performed based on this given information. Two outputs are possible:
 - a – Available time slots correspond to the search criteria. The searchResult attribute value is “success”. The list of available time slots with for each the related party concerned is given in the response in the availableTimeSlot structure.
 - b – None available time slots correspond to the search criteria. The searchResult attribute value is “fail”. Nothing is provided in the response in the availableTimeSlot structure.

Appointment API REST Specification

Note that these two outputs (2a and 2b) trigger a “searchTimeSlot.statusChange” to “done” and not to “terminatedWithError”. This last value is only used if the server (API Provider) is not able to perform the search request.

Mandatory and Non Mandatory Attributes

The following tables provides the list of mandatory and non mandatory including any possible rule conditions and applicable default values. Notice that it is up to an implementer to add additional mandatory attributes.

Mandatory Attribute	Rule
requestedTimeSlot	At least one period with a begin and an end date to restrict the search validFor.startDateTime must not be in the past validFor.endDateTime must not be in the past validFor.endDateTime must be superior to validFor.startDateTime

Non Mandatory Attributes	Rule
relatedParty	
relatedEntity	
place	

Additional Rules

The following table provides additional rules indicating mandatory fields in sub-resources or relationships when creating a resource.

Context	Mandatory Sub-Attributes
relatedParty	role + id OR href
relatedEntity	id OR href

Usage Samples

Request
<pre> POST /appointment/searchTimeSlot Content-type: application/json { "requestedTimeSlot": [{ "validFor": { "startDateTime":"2018-02-15T100:00:00.071Z", "endDateTime":"2018-02-18T00:00:00.071Z" } }] } </pre>

Appointment API REST Specification

```

    ],
    "relatedParty": {
      "id": "56",
      "href": "https://host:port/partyManagement/individual/56",
      "name": "John Doe",
      "role": "technician"
    },
    },
    "place": {
      "role": "interventionAddress",
      "@referredType": "geographicAddress",
      "geographicAddress": {
        "streetNr": "96",
        "streetType": "avenue",
        "streetName": "de Versailles",
        "postCode": "75016",
        "city": "Paris",
        "country": "France"
      }
    }
  },
  "relatedEntity": [
    {
      "id": "789",
      "href": "https://host:port/troubleTicket/troubleTicket/789",
      "name": "Internet connexion problem to solve",
      "@referredType": "troubleTicket"
    }
  ]
}

```

Response

201

Content-Type : application/json

```

{
  "id": "99",
  "href": "https://host:port/appointment/searchTimeSlot/99",
  "status": "done",
  "searchDate": "2018-02-01T14:40:43.071Z",
  "searchResult": "success",
  "availableTimeSlot": [
    {
      "validFor": {
        "startDateTime": "2018-02-15T14:00:00.071Z",
        "endDateTime": "2018-02-15T16:00:00.071Z"
      },
    },
    "relatedParty": {
      "id": "56",
      "href": "https://host:port/partyManagement/individual/56",
    }
  ]
}

```

```
"name":"John Doe",
"role":"technician"
}
},
{
"validFor" : {
"startDateTime":"2018-02-16T16:30:00.071Z",
"endDateTime":"2018-02-16T18:00:00.071Z"
},
"relatedParty":{
"id":"56",
"href":"https://host:port/partyManagement/individual/56",
"name":"John Doe",
"role":"technician"
}
},
{
"validFor" : {
"startDateTime":"2018-02-15T16:00:00.071Z",
"endDateTime":"2018-02-15T17:00:00.071Z"
},
"relatedParty":{
"id":"58",
"href":"https://host:port/partyManagement/individual/58",
"name":"Adam Smith",
"role":"technician"
}
}
}
}
```

LIST SEARCH TIME SLOTS

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

Description

This operation list search time slots entities.

Attribute selection is enabled for all first level attributes.

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

Note: A specific implementation of the Appointment Management API can choose whether or not to implement GET, PATCH, and DELETE for the SearchTimeSlot task resource, according to business considerations (such as the need for accounting or the need to support asynchronous search).

Usage Samples

Appointment API REST Specification

Request
<p>GET</p> <p>/appointment/searchTimeSlot?relatedParty.id=56&relatedParty.role=technician&fields=id,status,searchResult,availableTimeSlot</p> <p>Accept: application/json</p>
<p>200</p> <p>Content-Type : application/json</p> <pre>[{ "id": "99", "status": "done", "searchResult": "success", "availableTimeSlot": [{ "validFor": { "startDateTime": "2018-02-15T14:00:00.071Z", "endDateTime": "2018-02-15T16:00:00.071Z" }, "relatedParty": { "id": "56", "href": "https://host:port/partyManagement/individual/56", "name": "John Doe", "role": "technician" } }, { "validFor": { "startDateTime": "2018-02-16T16:30:00.071Z", "endDateTime": "2018-02-16T18:00:00.071Z" }, "relatedParty": { "id": "56", "href": "https://host:port/partyManagement/individual/56", "name": "John Doe", "role": "technician" } }] }]</pre>

RETRIEVE A SEARCH TIME SLOT

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

Description

This operation retrieves a search time slot 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.

Note: A specific implementation of the Appointment Management API can choose whether or not to implement GET, PATCH, and DELETE for the SearchTimeSlot task resource, according to business considerations (such as the need for accounting or the need to support asynchronous search).

Usage Samples

Request
<pre>GET /appointment/searchTimeSlot/99 Accept: application/json</pre>
<pre>200 Content-Type : application/json { "id": "99", "href": "https://host:port/appointment/searchTimeSlot/99", "status": "done", "searchDate": "2018-02-01T14:40:43.071Z", "searchResult": "success", "availableTimeSlot": [{ "validFor" : { "startDateTime": "2018-02-15T14:00:00.071Z", "endDateTime": "2018-02-15T16:00:00.071Z" }, "relatedParty": { "id": "56", "href": "https://host:port/partyManagement/individual/56", "name": "John Doe", "role": "technician" } }], }</pre>

```

"validFor" : {
  "startDateTime":"2018-02-16T16:30:00.071Z",
  "endDateTime":"2018-02-16T18:00:00.071Z"
},
"relatedParty":{
  "id":"56",
  "href":"https://host:port/partyManagement/individual/56",
  "name":"John Doe",
  "role":"technician"
}
},
{
  "validFor" : {
    "startDateTime":"2018-02-15T16:00:00.071Z",
    "endDateTime":"2018-02-15T17:00:00.071Z"
  },
  "relatedParty":{
    "id":"58",
    "href":"https://host:port/partyManagement/individual/58",
    "name":"Adam Smith",
    "role":"technician"
  }
}
]
}

```

PATCH A SEARCH TIME SLOT

PATCH /searchTimeSlot/{id}

Note: this operation is available only to ADMIN API users.

Description

This operation allows partial updates of a search time slot 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.

Note: A specific implementation of the Appointment Management API can choose whether or not to implement GET, PATCH, and DELETE for the SearchTimeSlot task resource, according to business considerations (such as the need for accounting or the need to support asynchronous search).

Appointment API REST Specification

Patchable and Non Patchable Attributes

The tables below provide the list of patchable and not patchable attributes, including constraint rules on their usage. Note that patching is possible only for ADMIN API users.

Patchable attributes	Rule
status	
searchResult	
availableTimeSlot	

Non Patchable attributes	Rule
id	
href	
searchDate	

Additional Rules

The following pre-condition allowed for this operation.

Pre-condition
PATCH allowed if status is still inProgress

Usage samples

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

Request
PATCH /appointment/searchTimeSlot/99 Content-Type:application/merge-patch+json { "status": "done" }
Response
200 Content-Type : application/json { "id": "99",

```

"href":"https://host:port/appointment/searchTimeSlot/99",
"status": "done",
"searchDate":"2018-02-01T14:40:43.071Z",
"searchResult": "success",
"availableTimeSlot": [
  {
    "validFor" : {
      "startDateTime":"2018-02-15T14:00:00.071Z",
      "endDateTime":"2018-02-15T16:00:00.071Z"
    },
    "relatedParty":{
      "id":"56",
      "href":"https://host:port/partyManagement/individual/56",
      "name":"John Doe",
      "role":"technician"
    }
  },
  {
    "validFor" : {
      "startDateTime":"2018-02-16T16:30:00.071Z",
      "endDateTime":"2018-02-16T18:00:00.071Z"
    },
    "relatedParty":{
      "id":"56",
      "href":"https://host:port/partyManagement/individual/56",
      "name":"John Doe",
      "role":"technician"
    }
  },
  {
    "validFor" : {
      "startDateTime":"2018-02-15T16:00:00.071Z",
      "endDateTime":"2018-02-15T17:00:00.071Z"
    },
    "relatedParty":{
      "id":"58",
      "href":"https://host:port/partyManagement/individual/58",
      "name":"Adam Smith",
      "role":"technician"
    }
  }
]
}

```

DELETE A SEARCH TIME SLOT

DELETE /searchTimeSlot/{id}

Note: this operation is available only to ADMIN API users.

Appointment API REST Specification

Description

This operation deletes a search time slot entity.

Note: A specific implementation of the Appointment Management API can choose whether or not to implement GET, PATCH, and DELETE for the SearchTimeSlot task resource, according to business considerations (such as the need for accounting or the need to support asynchronous search).

Usage samples

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

Request
DELETE /appointment/searchTimeSlot/99
Response
204 Content-Type : application/json

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.

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..

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.

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

Appointment API REST Specification

<pre>{ "event": { EVENT BODY }, "eventType": "EVENT_TYPE" }</pre>
Response
201

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

ACKNOWLEDGEMENTS

DOCUMENT HISTORY

VERSION HISTORY

Version Number	Date	Release led by:	Description
Release 1.0	04/15/2013	Pierre Gauthier TM Forum pgauthier@tmforum.org Orange Team	First Release of Draft Version of the Document.
2.0	10/07/2016	Orange Team	Second release taking into account remarks
2.1	14/12/2016	Orange Team	Update taking into account validation and approbation remarks from tmforum members for 16.5 release
3.0	10/04/2018	Orange with contribution from amdocs	Update for release 18.0: new features from guidelines 3.0, remarks from SID mapping, remarks from NTS Retail API implementation
3.0.1	27-Jun-2018	Adrienne Walcott	Formatting/style edits prior to R18 publishing

RELEASE HISTORY

Release Number	Date	Release led by:	Description
Release 18.0.0	24-Jun-2018	Pierre Gauthier	Initial Release