

TM Forum Specification

Service Ordering Management API Conformance Profile

TMF641B
Release 18.5.0
January 2019

Release: Release 18.5.0	Status: Member Evaluation
Version: 3.0.0	IPR Mode: RAND

NOTICE

Copyright © TM Forum 2019. All Rights Reserved.

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

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

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

Direct inquiries to the TM Forum office:

4 Century Drive, Suite 100
Parsippany, NJ 07054, USA
Tel No. +1 973 944 5100
Fax No. +1 973 998 7916
TM Forum Web Page: www.tmforum.org

Table of Contents

NOTICE.....	2
Table of Contents.....	3
INTRODUCTION - API DESCRIPTION.....	4
RESOURCE MODEL CONFORMANCE.....	5
API MANDATORY AND OPTIONAL RESOURCES.....	5
ServiceOrder MANDATORY AND OPTIONAL ATTRIBUTES.....	5
API OPERATIONS CONFORMANCE.....	10
API MANDATORY AND OPTIONAL OPERATIONS.....	10
API GET FILTERING OPERATION CONFORMANCE.....	11
Filtering in ServiceOrder resource.....	11
/serviceOrderingManagement/serviceOrder/.....	13
/serviceOrderingManagement/serviceOrder/{ID}.....	13
API POST OPERATION CONFORMANCE.....	14
/serviceOrderingManagement/serviceOrder/.....	14
API PATCH OPERATION CONFORMANCE.....	17
API DELETE OPERATION CONFORMANCE.....	18
API CONFORMANCE TEST SCENARIOS.....	19
ServiceOrder resource TEST CASES.....	19
Document History.....	28
Release History.....	28
Version History.....	28
Contributors to Document.....	28

INTRODUCTION - API DESCRIPTION

This document is the REST API conformance profile for the Service Ordering Management REST API (TMF 641).

RESOURCE MODEL CONFORMANCE

API MANDATORY AND OPTIONAL RESOURCES

For the resources defined by the API fill the following table and indicate which ones are mandatory and which ones are optional.

Resource Name	Mandatory or Optional	Comments
serviceOrder	M	

ServiceOrder MANDATORY AND OPTIONAL ATTRIBUTES

The table below summarizes mandatory and optional attributes for resource "ServiceOrder"

Attribute Name	Mandatory or Optional	Comments
id	M (in response messages) O (otherwise)	Initially generated by the server and provided in the response upon resource creation. Not available in POST request
href	M (in response messages) O (otherwise)	Initially generated by the server and provided in the response upon resource creation Value in response must be the same as the one set in Location header provided upon entity creation Not available in POST request

Attribute Name	Mandatory or Optional	Comments
externalId	O	
priority	O	Implementation could defined a be default value
description	O	
category	O	
state	M (in response messages) O (otherwise)	Only required in response messages from the server when providing the ServiceOrdering entity contents (responses to create-POST and read-GET the entity)
orderDate	M (in response messages)	Not available in POST request
completionDate	O	Not available in POST request
requestedStartDate	O	
requestedCompletionDate	O	
expectedCompletionDate	O	Not available in POST request
startDate	O	Not available in POST request
notificationContact	O	
@baseType	O	
@type	O	

Attribute Name	Mandatory or Optional	Comments
@schemaLocation	O	
relatedParty	O	Array of structures
id	M except if href provided	
href	M except if id provided	
role	M	
name	O	
@referredType	O	
@schemaLocation	O	
@type	O	
@baseType	O	
orderRelationship	O	Array of structures
type	M	
id	M except if href provided	
href	M except if id provided	
@referredType	O	
@schemaLocation	O	
@type	O	
@baseType	O	
note	O	Array of note(s)

Attribute Name	Mandatory or Optional	Comments
date	O	
author	M	
system	O	
text	M	
@schemaLocation	O	
@type	O	
@baseType	O	
serviceOrderItem	M	Array of item - At least one must be provided
id	M	
action	M	
state	M (in response messages) O (otherwise)	Only required in response messages from the server when providing the ServiceOrdering entity contents (responses to create-POST and read-GET the entity)
@type	O	
@schemaLocation	O	
@baseType	O	
appointment	O	Array of appointment(s)
id	M except if href provided	
href	M except if id	

Attribute Name	Mandatory or Optional	Comments
	provided	
description	O	
@referredType	O	
@schemaLocation	O	
@type	O	
@baseType	O	
orderItemRelationship	O	Array of Item relationship
relationshipType	M	
id	M	
@schemaLocation	O	
@type	O	
@baseType	O	
service		0 or 1
id	M except if href provided	
href	M except if id provided	
category	O	
name	O	
serviceType	O	
state	O	
@type	O	

Attribute Name	Mandatory or Optional	Comments
@baseType	O	
@schemaLocation	O	
supportingService	O	Array of service(s)
id	M except if href provided	
href	M except if id provided	
@type	O	
@referredType	O	
@schemaLocation	O	
@baseType	O	
supportingResource	O	Array of resource(s)
id	M except if href provided	
href	M except if id provided	
@type	O	
@referredType	O	
@schemaLocation	O	
@baseType	O	
serviceSpecificationRef	O	0 or 1
id	M	
href	M	

Attribute Name	Mandatory or Optional	Comments
version	O	
name	O	
@type	O	
@referredType	O	
@schemaLocation	O	
@baseType	O	
targetServiceSchema		0 or 1
@type	M	
@baseType	O	
@schemaLocation	M	
place	O	Array of place(s)
id	M except if href provided	
href	M except if id provided	
name	O	
role	M	
@rType	M	Because place could be a geographicAddress, geographicLocation or a geographicSite, a type must be provided.
@baseType	O	
@schemaLocation	O	

Attribute Name	Mandatory or Optional	Comments
characteristic		Array of product characteristic(s)
name	M	
valueType	O	If not filled, string is defaulted
value	M	1 - one value must be provided for each serviceCharacteristic
@type	O	
@baseType	O	
@schemaLocation	O	
serviceRelationship		Array of service relationship(s)
relationshipType	M	
@schemaLocation	O	
@type	O	
@baseType	O	
service	M	1 - this is a ref to the targeted service
id	M except if href provided	
href	M except if id provided	
@baseType	O	
@schemaLocation	O	

	Attribute Name	Mandatory or Optional	Comments
	@type	0	
	@referredType	0	

API OPERATIONS CONFORMANCE

For every single resource use the following templates and define what operations are optional and what operations are mandatory.

API MANDATORY AND OPTIONAL OPERATIONS

The following table indicates which ones are mandatory and which ones are optional for each one of the resources in the API (default is for all resources).

Uniform API Operation	Mandatory/Optional	Comments
GET	M	GET must be used to retrieve a representation of a resource
POST	M	POST must be used to create a new resource
PUT	O	PUT must be used to completely update a resource identified by its resource URI
PATCH (JSON-MERGE)	O	PATCH must be used to partially update a resource
DELETE	O	DELETE must be used to remove a resource

API GET FILTERING OPERATION CONFORMANCE

For every single resource use the following template to specify the mandatory and optional features supported by the GET filtering operation.

Definitions

Filtered Search: : A filtered search can be applied using query parameters in order to obtain only the resource entities that meet the criteria defined by the filtering parameters included in the query request. Several elements can be applied to the filtered search. In that case logic, a logical AND is applied to combine the criteria (e.g.:?category=<value> &status=<value>)

Attribute selection (Filtered Response Data): In order to apply a filter and limit the number of attributes included in the response, the GET request can include the “?fields=” query parameter. Several elements can be applied to the filter. In that case, a logical AND is applied to combine the values (e.g.:?fields=category,state) will provide in the response only the values assigned to attributes category and state. Attribute selection capabilities are the same for collections retrieval and individual resource queries

Filtering in ServiceOrder resource

Attribute name	Filtered search First Level	Filtered search N Level	Attribute Selection First Level	Attribute Selection N Level
id	M	M	M	M
href	NA	NA	O	O
externalId	M	O	M	O
priority	O	O	O	O
description	O	NA	M	NA
category	O	NA	O	NA
state	M	NA	M	NA
orderDate	M	NA	O	O
completionDate	M	NA	O	O
requestedStartDate	O	NA	M	NA
requestedCompletion Date	O	NA	O	NA
expectedCompletionDate	O	NA	M	NA
startDate	O	NA	M	NA

Attribute name	Filtered search First Level	Filtered search N Level	Attribute Selection First Level	Attribute Selection N Level
notificationContact	O	NA	O	NA
orderRelationship				
id	O	NA	O	O
type	O	NA	O	NA
href	NA	NA	O	NA
relatedParty				
id	O	NA	O	O
href	NA	NA	NA	NA
role	O	NA	O	O
name	O	NA	O	NA
serviceOrderItem				
id	NA	NA	M	O
state	O	NA	M	O
action	O	NA	O	O
orderItemRelationship				
id	NA	NA	NA	NA
relationshipType	NA	NA	NA	NA
serviceSpecification				
id	O	NA	O	NA
href	NA	NA	NA	NA
version	O	NA	O	NA
name	O	NA	O	NA
supportingService				
id	O	NA	O	NA
href	NA	NA	NA	NA
supportingResource				
id	O	NA	O	NA
href	NA	NA	NA	NA
service				
id	O	NA	O	NA
href	NA	NA	NA	NA
category	O	NA	O	NA
serviceType	O	NA	O	NA

Attribute name	Filtered search First Level	Filtered search N Level	Attribute Selection First Level	Attribute Selection N Level
name	O	NA	O	NA
state	O	NA	O	NA
place				
id	O	NA	O	NA
href	NA	NA	NA	NA
role	O	NA	O	NA
name	O	NA	O	NA
appointment				
id	O	NA	O	O
description	O	NA	O	NA
href	NA	NA	O	NA
note				
date	NA	NA	NA	NA
author	NA	NA	NA	NA
text	NA	NA	NA	NA
system	NA	NA	NA	NA

/serviceOrderingManagement/serviceOrder/

Filtered Search: A filtered search can be applied using the following filtering criteria

Providing filtering criteria is mandatory to avoid too many answers retrieved. See example below:

- externalId
- description
- orderDate
- state + orderDate interval (orderDate > 11/4/2018 & orderDate > 11/9/2017)

Filtered Data: A filtered response can be requested for the following attributes using the “?fields=” query parameter

- Any of the attributes in the first level of ServiceOrdering resource definition

/serviceOrderingManagement/serviceOrder/{ID}

Filtered Search: A filtered response can be requested for the following attributes using the “?fields=” query parameter

- Any of the attributes in the first level of ServiceOrder resource definition

API POST OPERATION CONFORMANCE

/serviceOrderingManagement/serviceOrder/

This Uniform Contract operation is used to create a ServiceOrder resource in the server.

The response to this operation must include a Location header set to /serviceOrderingManagement /serviceOrder/{ID} where {ID} indicates the identifier assigned by the server to the new ServiceOrder resource created

POST	M	
Response Status Code 201	M	
Other Status Codes	NA	

The following table indicates attributes that are required to be sent when creating a new ServiceOrder as well as attributes with special considerations. All other attributes defining the resource are not required to be sent as part of the BODY of the POST request message:

Attribute name	Mandator	Default	Rule
id	NA		Must not be in POST request
href	NA		Must not be in POST request
externalId	O		
priority	O	"4"	Could be implementation-based defined
description	O		
category	O		
state	NA		Must not be in POST request
orderDate	NA		Must not be in POST request
completionDate	NA		Must not be in POST request
requestedStartDate	O		
requestedCompletionDate	O		
expectedCompletionDate	NA		Must not be in POST request
startDate	NA		Must not be in POST request
NotificationContact	O		

Attribute name	Mandatory	Default	Rule
note	O		If note present, author and text are mandatory. Date is optional and values by default with today date.
orderRelationship	O		If orderRelationship present, type and (id AND/OR href) are mandatory
serviceOrderItem	M		At least one item must be provided.
orderItem.id	M		
orderItem.action	M		
orderItem.state	NA		Must not be in POST request
orderItemRelationship	O		if provided, type & id are mandatory
appointment	O		href or/and id must be provided
relatedParty	O		href or/and id must be provided role is mandatory
place	O		href or/and id must be provided Role is mandatory
serviceRelationship	O		type and service are mandatory
serviceSpecification	O		href or/and id must be provided
targetServiceSchema	O		@type & @schemaLocation must be provided
service.characteristic	O		name and value must be provided . ValueType is by default 'string' if not provided

The response from the server must include a BODY with the contents of the new resource created, filled with at least the same information elements that were included in the request and are supported by the server. Notice that the value stored by the server may be different than the one set in the request (e.g.: serviceSpecification.Name)

If the POST request includes optional parameters (as per the model resource definition) that are not supported by the server, then the server must reject the request (replying with a 4xx error response) indicating the parameter not supported.

The BODY of the response from the server must include attribute “href” set to the same value as the one in the Location header.

The server must include in the BODY of the response, even if they are not included in the request, the following attributes that are mandatory in the definition of a ServiceOrder as per the resource model defined

- id
- href
- state
- orderDate
- orderItem.state

We must note that some attributes must not be send in the POST request and could be not valued in the POST answer:

- expectedCompletionDate
- completionDate
- startDate

The BODY of the response from the server could include attribute “href” (or “reference”) under each one of the entities within the ServiceOrder model that can be addressed individually and were included in the response. This applies to any of the following entities

- relatedParty
- serviceSpecification
- place
- service
- supportingService
- supportingResource

API PATCH OPERATION CONFORMANCE

This section defines which attributes are patchable.

Since PATCH operation is optional and not included in the basic CONNECT certification this is not applicable in this conformance document.

API DELETE OPERATION CONFORMANCE

This section defines what operations can be used to delete a ServiceOrder resource.

Since DELETE operation is optional and not included in the basic CONNECT certification this is not applicable in this conformance document

API CONFORMANCE TEST SCENARIOS

This section describes the test scenarios required for the basic CONNECT certification of ServiceOrdering API.

Test Cases must be executed in the order defined for each resource because the result from one of the scenarios will be input for the next one.

Requests must be addressed to the endpoint provided for certification, specifically they must be addressed to the URI defined by the concatenation of the {apiRoot} and the specific resource, where the {apiRoot} is defined as

{serverRoot}/serviceOrderingManagement/v1, being {serverRoot} defines the certification endpoint

ServiceOrder resource TEST CASES

Nominal Scenarios

TC_ServiceOrder_N1 - Create new ServiceOrder with minimum required information

- Send a POST message to {apiRoot}/serviceOrder/ with the following contents in the BODY

```
{
  "externalId": "OrangeBSS748",
  "priority": "1",
  "description": "Service order description",
  "category": "CloudServiceOrdering",
  "requestedStartDate": "2018-01-15T09:37:40.508Z",
  "requestedCompletionDate": "2018-01-15T09:37:40.508Z",
  "orderItem": [
    {
      "id": "1",
      "action": "add",
      "service": {
        "state": "active",
        "serviceType": "CFS",
        "serviceCharacteristic": [
          {
            "name": "VCPE_IP",
            "valueType": "String",
            "value": "193.218.236.21"
          }
        ]
      },
      "serviceSpecification": {
        "id": "12",
        "href": "http://serverurl/productCatalogManagement/v1/productSpecification/12",

```



```

        "name": "VCPE",
        "version": "1"
    }
}
]
}

```

- Wait for a response from the server with the following characteristics
 - Response Code 201-Created
 - Include a location header in the body set to `/{apiRoot}/serviceOrder/{IDSO1}` where `{IDSO1}` indicates the identifier assigned by the server to the new ServiceOrder resource
 - The response message includes all mandatory parameters (including `id`, `state`, `orderItem.state` and `orderDate` that were not sent in the original request)
 - The body of the response matches the values set in the original request
- Send a GET message to `/{apiRoot}/serviceOrder/{IDSO1}`
- Wait for a response from the server with the following characteristics
 - Response Code 200-OK
 - The body of the response includes one ServiceOrder resource with ID set to `{IDSO1}`, the same identifier as assigned by the server to the new resource created
 - The response message includes all mandatory parameters
 - The body of the response for the resource with identifier `{IDSO1}` matches the values set in the original request

TC_ServiceOrder_N2 - Create new ServiceOrder with minimum set of parameters supported by server

- Send a POST message to `{apiRoot}/serviceOrder/` with the following contents in the BODY

```

{
  "externalId": "TMFS0741",
  "priority": "2",
  "description": "Service order description",
  "category": "CloudServiceOrdering",

```

```

"requestedStartDate": "2019-03-15T09:37:40.508Z",
"requestedCompletionDate": "2019-03-15T09:37:40.508Z",
"orderItem": [
  {
    "id": "1",
    "action": "add",
    "service": {
      "state": "active",
      "serviceType": "RFS",
      "serviceCharacteristic": [
        {
          "name": "ProtectionLevel",
          "valueType": "String",
          "value": "BulletProof"
        },
        {
          "name": "active",
          "valueType": "boolean",
          "value": true
        }
      ],
      "serviceSpecification": {
        "id": "5697-hg45",
        "href": "http://serverurl/productCatalogManagement/v1/productSpec/5697-
hg45",
        "name": "NiceGuyFirewal",
        "version": "1"
      }
    }
  }
]
}

```

- Wait for a response from the server with the following characteristics
 - Response Code 201-Created
 - Include a location header in the body set to `/{apiRoot}/serviceOrder/{IDS02}` where `{IDS02}` indicates the identifier assigned by the server to the new ServiceOrder resource
 - The response message includes all mandatory parameters (including orderDate and state that were not sent in the original request)
 - The body of the response matches the values set in the original request
- Send a GET message to `/{apiRoot}/serviceOrder/{IDS02}`
- Wait for a response from the server with the following characteristics

- Response Code 200-OK
- The body of the response includes one ServiceOrder resource with ID set to {IDS02}, the same identifier as assigned by the server to the new resource created
- The response message includes all mandatory parameters
- The body of the response for the resource with identifier {IDS02} matches the values set in the original request

TC_ServiceOrder_N3 - Search for ServiceOrder with specific characteristics

- Send a GET message to /{apiRoot}/serviceOrder?category=cloudServiceOrdering
- Wait for a response from the server with the following characteristics
 - Response Code 200-OK
 - The body of the response includes at least two ServiceOrder resources referring to {IDS01} and {IDS02}
 - The body of the response for the resource with each identifier matches the values in the corresponding original request
- Send a GET message to /{apiRoot}/serviceOrder?priority=1&category=cloudServiceOrdering
- Wait for a response from the server with the following characteristics
 - Response Code 200-OK
 - The body of the response includes one ServiceOrder resource referring to {IDS01} and there is no reference to ServiceOrder resource {IDS02}
 - The response message includes all mandatory parameters
 - The body of the response for the resource with identifier {IDS01} matches the values in the original request
- Send a GET message to /{apiRoot}/serviceOrder?externalId= TMFSO741
- Wait for a response from the server with the following characteristics

- Response Code 200-OK
- The body of the response includes one ServiceOrder resource referring to {IDS02} and there is no reference to ServiceOrder resource {IDS01}
- The response message includes all mandatory parameters
- The body of the response for the resource with identifier {IDS02} matches the values in the original request

TC_ServiceOrder_N4 - Filtered retrieval of ServiceOrder

- Send a GET message to `/{apiRoot}/serviceOrder/{IDS02}?fields=id,href,externalId,priority,state`
- Wait for a response from the server with the following characteristics
 - Response Code 200-OK
 - The body of the response includes one ServiceOrder resource referring to {IDS02} and including only attributes id,href,externalId, priority and state.
- Send a GET message to `/{apiRoot}/serviceOrder/{IDS01}?fields=id,state,orderItem.id,orderItem.state,orderItem.action`
- Wait for a response from the server with the following characteristics
 - Response Code 200-OK
 - The body of the response includes one ServiceOrder resource referring to {IDS01} and including only attributes id, state, and an array of order item with attributes: id, state, and action matching the values in the original request.

Notice that this test case is using parameters "state", "id", "orderItem.id", "orderItem.state" and "orderItem.action" to filter the data included in the response but any other parameter could be used

TC_ServiceOrder_N5 - Filtered Search and Filtered data response

- Send a GET message to `{apiRoot}/serviceOrder?externalId=OrangeBSS748&fields=id,state,category,description`
- Wait for a response from the server with the following characteristics
 - Response Code 200-OK
 - The body of the response includes one ServiceOrder resource referring to {IDSO1} and there is no reference to ServiceOrder resource {IDSO2}
 - The body of the response for the resource includes only attributes id, state and category matching the values in the corresponding original request

Notice that this test case is using the parameters id,state,category, and description to filter the data included in the response but any other parameter could be used

Error Scenarios**TC_ServiceOrder_E1 - Unknown Trouble ServiceOrder identifier**

- Send a GET message to `{apiRoot}/serviceOrder/{IDSO3}`, where {IDSO3} does not match any of the identifiers previously created in the server
- Wait for a response from the server with the following characteristics
 - Response Code 404-Not Found

TC_ServiceOrder_E2 - Invalid Request - unexpected parameter

- Send a POST message to `{apiRoot}/ServiceOrder/` with the following contents in the BODY.

```
{
  "externalId": "TMFS0789",
  "priority": "3",
  "description": "Service order description",
  "category": "CloudServiceOrdering",
  "requestedStartDate": "2019-03-15T09:37:40.508Z",
  "state": "acknowledged",
  "requestedCompletionDate": "2019-03-15T09:37:40.508Z",
  "expectedCompletionDate": "2000-01-23T04:56:07.000+00:00",
  "note": [
```

```

    {
      "date": "2018-01-15T09:37:40.508Z",
      "author": "Jean Pontus",
      "text": "bla bla bla"
    }
  ],
  "orderItem": [
    {
      "id": "1",
      "action": "add",
      "state": "acknowledged",
      "service": {
        "state": "active",
        "serviceType": "RFS",
        "serviceCharacteristic": [
          {
            "name": "ProtectionLevel",
            "valueType": "String",
            "value": "Light"
          },
          {
            "name": "active",
            "valueType": "boolean",
            "value": true
          }
        ]
      },
      "serviceSpecification": {
        "id": "5697-hg45",
        "href":
"http://serverurl/productCatalogManagement/v1/productSpec/5697-hg45",
        "name": "NiceGuyFirewal",
        "version": "1"
      }
    }
  ]
}

```

Notice that this request features unaccepting parameters as state, orderItem.state and expectedCompletionDate.

- Wait for an error response from the server indicating the unexpected parameters in the request

TC_ServiceOrder_E3 - Invalid Request - Missing parameter mandatory in context

- Send a POST message to {apiRoot}/serviceOrder/ with the following contents in the BODY.

```

{
  "externalId": "TMFS1245",
  "priority": "1",

```

```

"description": "Service order description",
"category": "CloudServiceOrdering",
"requestedStartDate": "2019-03-15T09:37:40.508Z",
"state": "acknowledged",
"requestedCompletionDate": "2019-03-15T09:37:40.508Z",
"expectedCompletionDate": "2000-01-23T04:56:07.000+00:00",
  "note": [
    {
      "date": "2018-01-15T09:37:40.508Z",
      "author": "Jean Pontus",
      "text": "bla bla bla"
    }
  ],
"orderItem": [
  {
    "id": "1",
    "action": "add",
    "state": "acknowledged",
    "service": {
      "state": "active",
      "serviceType": "RFS",
      "serviceCharacteristic": [
        {
          "name": "ProtectionLevel",
          "valueType": "String",
          "value": "BulletProof"
        },
        {
          "name": "active",
          "valueType": "boolean",
          "value": true
        }
      ],
      "serviceSpecification": {
        "name": "NiceGuyFirewal",
        "version": "1"
      }
    }
  }
]
}

```

Notice that this request is missing a service specification id and href in the order item described (action is add so it must be specified) but any other mandatory parameter could be used

- Wait for an error response from the server indicating the mandatory parameter is missing in the request

Document History

Release History

Release Number	Date	Release led by:	Description
Release 18.0.0	12-Jan-2018	Ludovic Robert	First Release of the Document.
Release 18.5.0	14-Jan-2019	Henrique Rodrigues (TM Forum) hrodrigues@tmforum.org Ludovic Robert (Orange)	Version updated from the Release 1.0 following the specifications on the 18.5 Open API specification

Version History

Release Number	Date	Release led by:	Description
Version 1.0.0	14-Jan-2019	Henrique Rodrigues (TM Forum) hrodrigues@tmforum.org	First Release of Draft Version of the Document

Contributors to Document

Ludovic Robert	Orange
Pierre Gauthier	TM Forum
Henrique Rodrigues	TMForum