TM Forum Specification

Service Ordering Management API User Guide

TMF641

Team Approved Date: 28-May-2020

Release Status: Pre-production	**Approval Status: Team Approved**
Version 4.0.0	**IPR Mode: RAND**

NOTICE

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

Direct inquiries to the TM Forum office:

4 Century Drive, Suite 100
Parsippany, NJ 07054, USA
Tel No. +1 973 944 5100
Fax No. +1 973 998 7196
TM Forum Web Page: www.tmforum.org

Table of Contents

Introduction

The following document is the specification of the REST API for Service Order Management. It includes the model definition as well as all available operations. Possible actions are creating, updating and retrieving Service Orders (including filtering). The specification covers also a task-based resource to request Service Order Cancellation.

The following Assumptions were considered in the development of this document:

- The Order Management system has access to a catalog system
- A service order will describe a list of service order items.
- A service order item references an action on an existing or future service.
- By service we designed Customer-Facing Service (CFS) as well as Resource Facing Service (RFS).

From a component perspective, a service order should be available

- from a Service Orchestration Component (and it could mix CFS and RFS)
- from an Infrastructure Control & Management component (and it would have only RFS)

Service ordering API performs the following operations

…on Service Order:

- Retrieval of a service order or a collection of service orders depending on filter criteria
- Partial update of a service order (including updating rules)
- Creation of a service order (including default values and creation rules)
- Deletion of service order (for administration purposes)
- Notification of events on service order
    - service order creation
    - service order deletion
    - service order state change
    - service order value change used to notify that any data in an order has just changed
    - service order information required used to notify that some data in the order need to be filled / are missing

…on CancelServiceOrder:

- Retrieval of a service order cancellation request or a collection of service order cancellation requests depending on filter criteria
- Creation of a service order cancel request
- Notification of events on service order
    - cancel service Order creation
    - cancel service Order state change
    - order information required used to notify that some data in the cancel service order need to be filled / are missing

SAMPLE USE CASES

We illustrate in this document a multi-items service order covering use-cases:

- Item 1: Creation of a very simplified CPE illustrating an object characteristic
- Item 2: Modification of an existing service to change a characteristic value and a supporting resource.
- Item 3: Creation of a service with feature
- Item 4 : Modification of a service order item to switch to inactive an existing service

The document also provide an example to a cancel service order request/

Support of polymorphism and extension

Support of polymorphic collections and types and schema based extension is provided by means of a list of generic meta-attributes that we describe below. Polymorphism in collections occurs when entities inherit from base entities, for instance a TypeAServiceOrder or TypeBServiceOrder inheriting properties from the base ServiceOrder entity.

Generic support of polymorphism and pattern extensions is described in the TMF API Guidelines v3.0 Part 2 document.

The @type attribute provides a way to represent the actual class type of an entity. For example, within a list of ServiceOrder instances some may be instances of TypeAServiceOrder where other could be instances of TypeBServiceOrder . The @type gives this information. All resources and sub-resources of this API have a @type attributes that can be provided when this is useful.

The @referredType can be used within reference entities (like for instance a RelatedParty object) to explicitly denote the actual entity type of the referred class. Notice that in reference entities the @type, when used, denotes the class type of the reference itself, such as RelatedParty, and not the class type of the referred object. However, since reference classes are rarely sub-classed, @type is generally not useful in reference objects.

The @schemaLocation property can be used in resources to allow specifying user-defined properties of an Entity or to specify the expected *characteristics* of an entity.

The @baseType attribute gives a way to provide explicitly the base of class of a given resource that has been extended

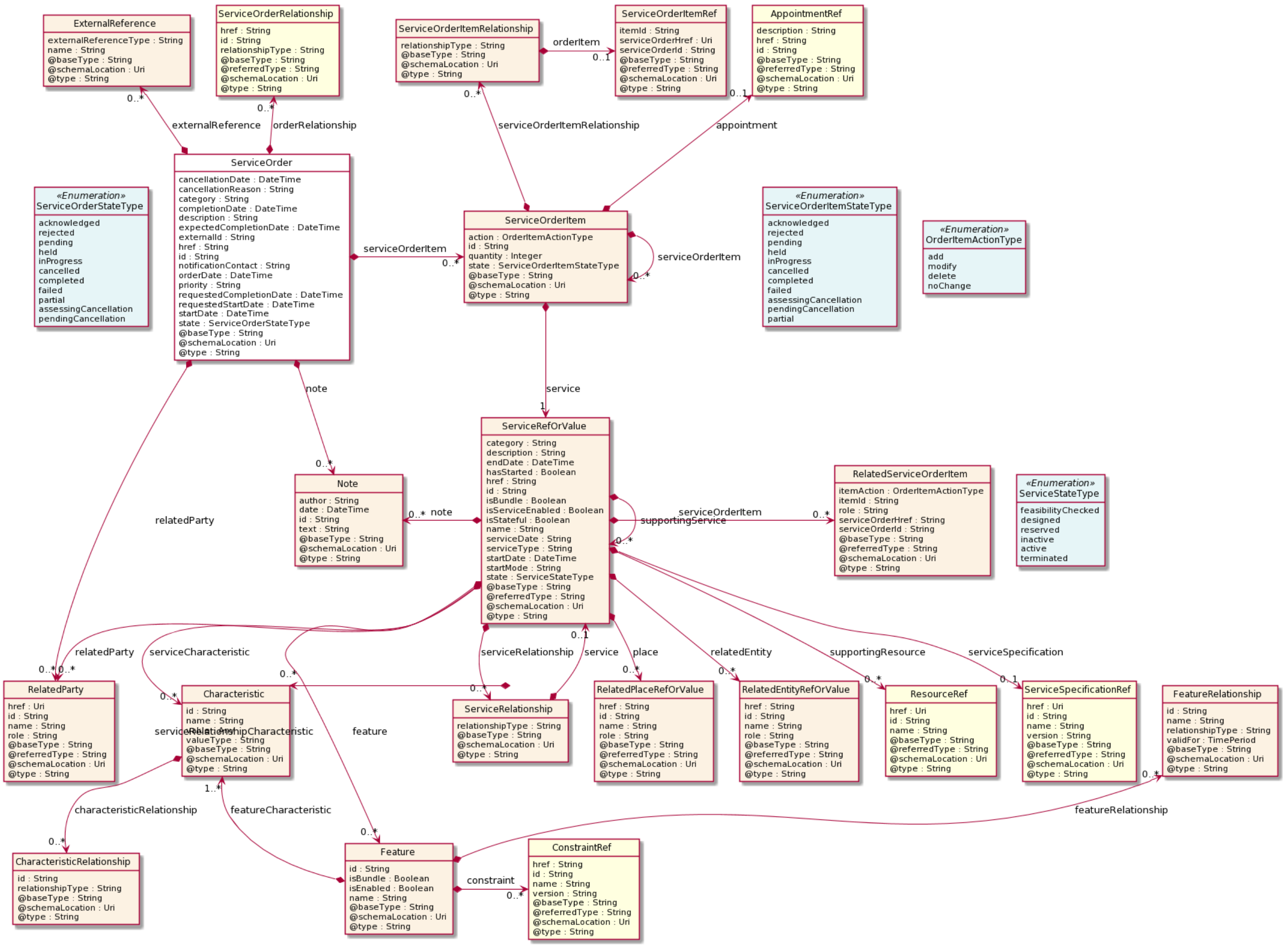
RESOURCE MODEL

Managed Entity and Task Resource Models

Service Order resource

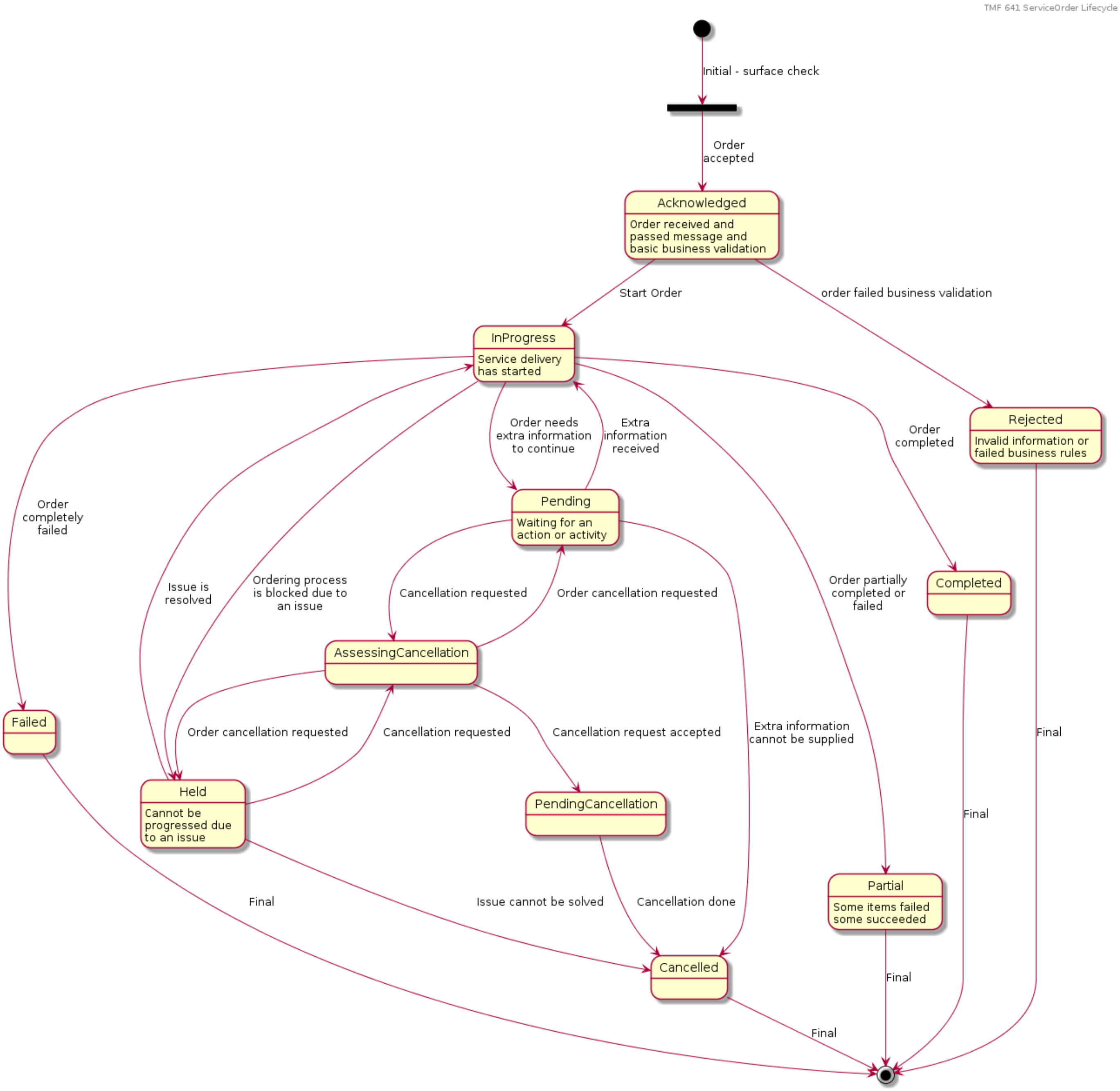
A Service Order is a type of order which can be used to describe a group of operations on service – one service order item per service. An action at the level of the service order item describe the operation to be done on a service (add, changer, terminate for example). The service order is triggered from the BSS system in charge of the product order management to the SOM (Service order Management) system that will manage the service fulfillment.

Resource model



Here is the state machine diagram for a Service order. Each order state is described in the table below.

Note Consistence rules are defined between order state & order item state.


TMF 641 ServiceOrder Lifecycle
Initial - surface check
Order accepted
Acknowledged
Order received and passed message and basic business validation
Start Order
order failed business validation
InProgress
Service delivery has started
Order needs extra information to continue
Extra information received
Order completed
Rejected
Invalid information or failed business rules
Order completely failed
Pending
Waiting for an action or activity
Issue is resolved
Ordering process is blocked due to an issue
Cancellation requested
Order cancellation requested
Order partially completed or failed
Completed
AssessingCancellation
Failed
Order cancellation requested
Cancellation requested
Cancellation request accepted
Extra information cannot be supplied
Final
Held
Cannot be progressed due to an issue
PendingCancellation
Final
Final
Issue cannot be solved
Cancellation done
Partial
Some items failed some succeeded
Cancelled
Final
Final

ServiceOrder & ServiceOrderItem State definition:

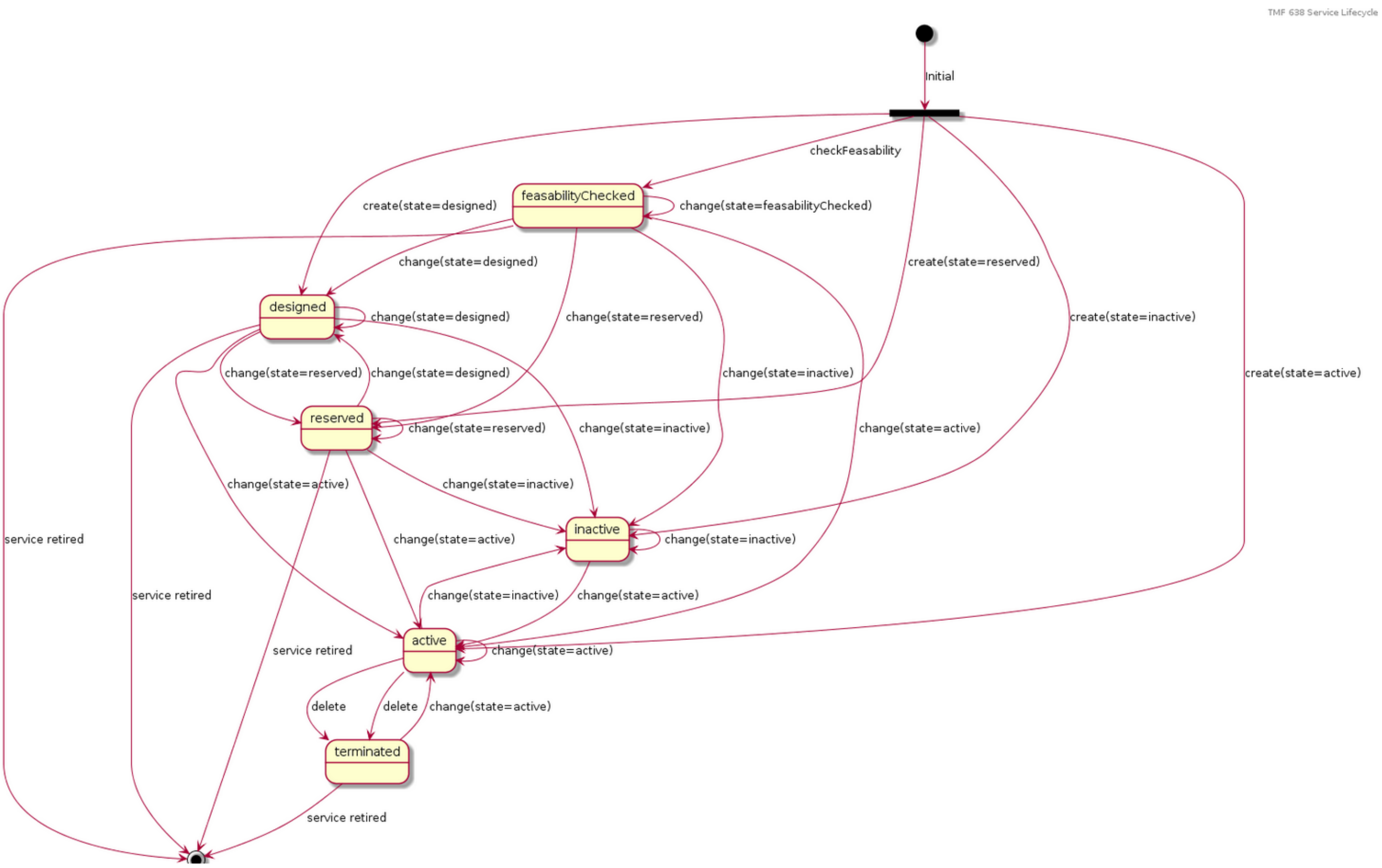
The following table provides service order state and service order item state description:

State	Definition
Acknowledged	The Acknowledged state is when an order/item has been received and has passed surface message validation.

Rejected	The Rejected state is where: • Invalid information is provided through the order request (not checked synchronously – like check that a service id provided exist in the inventory and relation service order item action is valid) • The order request fails to meet business rules for ordering Same rules applied for order item. If an order item is rejected the complete service order is rejected.
Pending	The Pending state is used when an order/item is currently in a waiting stage for an action/activity to be completed before the order/item can progress further, pending order amend or cancel assessment. In situations where Access Seeker action is required, an “information required” notification will be issued on transition into this state. A pending stage can lead into auto cancellation of an order/item, if no action is taken.
InProgress	The In Progress state is where an order Item has passed the Order check successfully and service delivery has started. For the order at least one order item is inProgress
Held	The Held state is used when an order/item cannot be progressed due to an issue. Service provided has temporarily delayed completing an order/item to resolve an infrastructure shortfall to facilitate supply of order. Upon resolution of the issue, the order/item will continue to progress.
AssessingCancellation	Following a cancel request, the SP is assessing if cancel can be done for the order/item (or if the Service Order has reached PONR). If cancellation request is not accepted after assessment the order will return in Held or Pending or InProgress state.
PendingCancellation	Once a cancel order has been accepted by SP, it could in some UC take time to effectively cancel the order/item. During this time when accepted cancellation is effectively processed, we use Pending Cancellation.
Cancelled	The Cancelled state is where an In-Flight Order/item has been successfully cancelled.
Completed	The Completed state is where an item has complete provision and the service is now active. For an order if **all** order items are completed
Failed	Order item as not a successful delivery completion. The service is not delivered, and the order item failed. **All** Order items have failed which results in the entire Order has Failed.
Partial	Some Order items have failed, and some have succeeded so the entire Order is in a Partial state. This provides support for partial Failure of an Order (or a service order item with ‘embedded’ service order item)

Lifecycle of Service

Here is the state machine diagram for a Service:



The following table provided service state explanation:

A service is in state…	when the Service Component Activation Interface shall support the following operations
feasabilityChecked	The service component OS is requested to determine whether the necessary resources are available and sufficient for the installation of a given service. It should be noted that this is just an initial feasibility check and no RFS is created.
designed	The Service Component OS is requested to design the service. This could be used to allocate resources, but nothing is supposed to be reserved.
reserved	The Service Component OS is requested to reserve a resource, or a set of resources, required for a service. This situation allows for the requesting OS to determine whether the underlying resources are available and reserve them in order to support a service (an RFS). As a result of the reservation request, a service is instantiated & reserved.
inactive	The service is deactivated - The Resource Facing service is deactivated and thus is no longer available for service. It remains allocated to the CFS that is managed by the high level service activation OS
active	The service component OS is requested to activate a given resource facing service such that the component is fully available and active as part of the CFS. When this request is complete, all RFS component shall be in active state
terminated	The service is 'logically deleted'. The resource facing service is deleted and thus allocated from the CFS. All associated resources are freed and made available for service to other users.

Note: when action 'no change' is used we did not expect any state change.

Field descriptions

ServiceOrder fields

cancellationDate	A date time (DateTime). Date when the order is cancelled. This is used when order is cancelled.
cancellationReason	A string. Reason why the order is cancelled. This is used when order is cancelled.
category	A string. Used to categorize the order, useful for the OM system, such as: Broadband, TVOption.
completionDate	A date time (DateTime). Effective delivery date amended by the provider.
description	A string. A free-text description of the service order.
expectedCompletionDate	A date time (DateTime). Expected delivery date amended by the provider.
externalId	A string. ID given by the consumer to facilitate searches.
externalReference	A list of external references (ExternalReference [*]). External reference of the individual or reference in other system.
href	A string. Hyperlink to access the order.
id	A string. ID created on repository side.
note	A list of notes (Note [*]). Extra-information about the order, e.g. useful to add extra delivery information that could be useful for a human process.
notificationContact	A string. Contact attached to the order to send back information regarding this order.
orderDate	A date time (DateTime).
orderRelationship	A list of service order relationships (ServiceOrderRelationship [*]). A list of service orders related to this order (e.g. prerequisite, dependent on).
priority	A string. Can be used by consumers to prioritize orders in a Service Order Management system.
relatedParty	A list of related parties (RelatedParty [*]). A list of parties which are involved in this order and the role they are playing.
requestedCompletionDate	A date time (DateTime). Requested delivery date from the requestor's perspective.
requestedStartDate	A date time (DateTime). Order start date wished by the requestor.

serviceOrderItem	A list of service order items (ServiceOrderItem [*]). A list of service order items to be processed by this order.
startDate	A date time (DateTime). Date when the order was started for processing.
state	A service order state type (ServiceOrderStateType). State of the order: described in the state-machine diagram.

Characteristic sub-resource

Describes a given characteristic of an object or entity through a name/value pair.

@baseType	A string. When sub-classing, this defines the super-class.
@schemaLocation	An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.
@type	A string. When sub-classing, this defines the sub-class Extensible name.
characteristicRelationship	A list of characteristic relationships (CharacteristicRelationship [*]). Another Characteristic that is related to the current Characteristic;.
id	A string. Unique identifier of the characteristic.
name	A string. Name of the characteristic.
value	An any (Any). The value of the characteristic.
valueType	A string. Data type of the value of the characteristic.

CharacteristicRelationship sub-resource

Another Characteristic that is related to the current Characteristic;.

id	A string. Unique identifier of the characteristic.
relationshipType	A string. The type of relationship.

ExternalReference sub-resource

External reference of the individual or reference in other system.

externalReferenceType	A string. Type of the external reference.
name	A string. External reference name.

Feature sub-resource

Configuration feature.

constraint	A list of constraint references (ConstraintRef [*]). This is a list of feature constraints.

featureCharacteristic A list of characteristics (Characteristic [1..*]). This is a list of Characteristics for a particular feature.

featureRelationship A list of feature relationships (FeatureRelationship [*]). Configuration feature.

id A string. Unique identifier of the feature.

isBundle A boolean. True if this is a feature group. Default is false.

isEnabled A boolean. True if this feature is enabled. Default is true.

name A string. This is the name for the feature.

FeatureRelationship sub-resource

Configuration feature.

id A string. Unique identifier of the target feature.

name A string. This is the name of the target feature.

relationshipType A string. This is the type of the feature relationship.

validFor A time period. The period for which this feature relationship is valid.

Note sub-resource

Extra information about a given entity.

author A string. Author of the note.

date A date time (DateTime). Date of the note.

id A string. Identifier of the note within its containing entity (may or may not be globally unique, depending on provider implementation).

text A string. Text of the note.

RelatedEntityRefOrValue sub-resource

A reference to an entity, where the type of the entity is not known in advance. A related entity defines an entity described by reference or by value linked to a specific entity. The polymorphic attributes @type, @schemaLocation & @referredType are related to the Entity and not the RelatedEntityRefOrValue class itself.

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

name A string. Name of the related entity.

href A string. Reference of the related entity.

id A string. Unique identifier of a related entity.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type A string. When sub-classing, this defines the sub-class Extensible name.

role A string.

RelatedParty sub-resource

Related Entity reference. A related party defines party or party role linked to a specific entity.

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

name A string. Name of the related entity.

href An uri (Uri). Hyperlink reference.

id A string. unique identifier.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type A string. When sub-classing, this defines the sub-class Extensible name.

role A string. Role played by the related party.

RelatedPlaceRefOrValue sub-resource

Related Entity reference. A related place defines a place described by reference or by value linked to a specific entity. The polymorphic attributes @type, @schemaLocation & @referredType are related to the place entity and not the RelatedPlaceRefOrValue class itself.

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

name A string. A user-friendly name for the place, such as [Paris Store], [London Store], [Main Home].

href A string. Unique reference of the place.

id A string. Unique identifier of the place.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type A string. When sub-classing, this defines the sub-class Extensible name.

role A string.

RelatedServiceOrderItem sub-resource

RelatedServiceOrderItem (a ServiceOrder item) .The service order item which triggered service creation/change/termination.

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

itemAction An order item action type (OrderItemActionType). Action of the order item for this service.

itemId A string. Identifier of the order item where the service was managed.

role A string. role of the service order item for this service.

serviceOrderHref A string. Reference of the related entity.

serviceOrderId A string. Unique identifier of a related entity.

ServiceOrderItem sub-resource

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type A string. When sub-classing, this defines the sub-class Extensible name.

action An order item action type (OrderItemActionType). The action to be carried out on the Service. Can be: add, modify, delete, noChange.

appointment An appointment reference (AppointmentRef). An appointment that was set up with a related party for this order item.

id A string. Identifier of the individual line item.

quantity An integer. Quantity ordered.

service A service ref or value (ServiceRefOrValue). The Service to be acted on by the order item.

serviceOrderItem A list of service order items (ServiceOrderItem [*]). A list of order items embedded to this order item.

serviceOrderItemRelationship A list of service order item relationships (ServiceOrderItemRelationship [*]). A list of order items related to this order item.

state A service order item state type (ServiceOrderItemStateType). State of the order item: described in the state machine diagram. This is the requested state.

ServiceOrderItemRelationship sub-resource

Linked service order item to the one containing this attribute.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type A string. When sub-classing, this defines the sub-class Extensible name.

orderItem A service order item reference (ServiceOrderItemRef). A service order item in relationship with this order item.

relationshipType A string. The type of related order item, can be: dependency if the order item needs to be not started until another order item is complete.

ServiceOrderRelationship sub-resource

Linked service order to the one containing this attribute.

@referredType A string. The entity type of the related order.

href A string. A hyperlink to the related order.

id A string. The id of the related order.

relationshipType A string. The type of related order, such as: [dependency] if the order needs to be [not started] until another order item is complete (a service order in this case) or [cross-ref] to keep track of the source order (a productOrder).

ServiceRefOrValue sub-resource

A Service to be created defined by value or existing defined by reference. The polymorphic attributes @type, @schemaLocation & @referredType are related to the Service entity and not the RelatedServiceRefOrValue class itself.

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

name A string. Name of the service.

href A string. Reference of the service.

id A string. Unique identifier of the service.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type	A string. When sub-classing, this defines the sub-class Extensible name.
category	A string. Is it a customer facing or resource facing service.
description	A string. Free-text description of the service.
endDate	A date time (DateTime). Date when the service ends.
hasStarted	A boolean. If TRUE, this Service has already been started.
isBundle	A boolean. If true, the service is a ServiceBundle which regroup a service hierachy. If false, the service is an 'atomic' service (hierachy leaf).
isServiceEnabled	A boolean. If FALSE and hasStarted is FALSE, this particular Service has NOT been enabled for use - if FALSE and hasStarted is TRUE then the service has failed.
isStateful	A boolean. If TRUE, this Service can be changed without affecting any other services.
serviceDate	A string. Date when the service was created (whatever its status).
serviceType	A string. Business type of the service.
startDate	A date time (DateTime). Date when the service starts.
startMode	A string. This attribute is an enumerated integer that indicates how the Service is started, such as: 0: Unknown; 1: Automatically by the managed environment; 2: Automatically by the owning device; 3: Manually by the Provider of the Service; 4: Manually by a Customer of the Provider; 5: Any of the above.
feature	A list of features (Feature [*]). A list of features associated with this service.
note	A list of notes (Note [*]). A list of notes made on this service.
place	A list of related place ref or values (RelatedPlaceRefOrValue [*]). A list of places (Place [*]). Used to define a place useful for the service (for example a geographical place whre the service is installed).
relatedEntity	A list of related entity ref or values (RelatedEntityRefOrValue [*]). A list of related entity in relationship with this service.
relatedParty	A list of related parties (RelatedParty [*]). A list of related party references (RelatedParty [*]). A related party defines party or party role linked to a specific entity.
serviceCharacteristic	A list of characteristics (Characteristic [*]). A list of characteristics that characterize this service (ServiceCharacteristic [*]).
serviceOrderItem	A list of related service order items (RelatedServiceOrderItem [*]). A list of service order items related to this service.

serviceRelationship A list of service relationships (ServiceRelationship [*]). A list of service relationships (ServiceRelationship [*]). Describes links with other service(s) in the inventory.

serviceSpecification A service specification reference (ServiceSpecificationRef). The specification from which this service was instantiated.

state A service state type (ServiceStateType). The life cycle state of the service, such as designed, reserved, active, etc...

supportingResource A list of resource references (ResourceRef [*]). A list of supporting resources (SupportingResource [*]).Note: only Service of type RFS can be associated with Resources.

supportingService A list of service ref or values (ServiceRefOrValue [*]). A list of supporting services (SupportingService [*]). A collection of services that support this service (bundling, link CFS to RFS).

ServiceRelationship sub-resource

relationshipType A string.

service A service ref or value (ServiceRefOrValue). A Service to be created defined by value or existing defined by reference. The polymorphic attributes @type, @schemaLocation & @referredType are related to the Service entity and not the RelatedServiceRefOrValue class itself.

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

AppointmentRef relationship

Refers an appointment, such as a Customer presentation or internal meeting or site visit.

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

description A string. An explanatory text regarding the appointment made with a party.

href A string. The reference of the appointment.

id A string. The identifier of the referred appointment.

ConstraintRef relationship

Constraint reference. The Constraint resource represents a policy/rule applied to an entity or entity spec.

@referredType A string. The (class) type of the referred constraint.

href A string. Hyperlink reference to the target constraint.

id A string. reference id to the target constraint.

name A string. Name given to the constraint.

version A string. constraint version.

ResourceRef relationship

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

name A string. Name of the related entity.

href An uri (Uri). Hyperlink reference.

id A string. unique identifier.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type A string. When sub-classing, this defines the sub-class Extensible name.

ServiceOrderItemRef relationship

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type A string. When sub-classing, this defines the sub-class Extensible name.

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

itemId A string. Identifier of the line item.

serviceOrderHref An uri (Uri). Link to the order to which this item belongs to.

serviceOrderId A string. Identifier of the order that this item belongs to.

ServiceSpecificationRef relationship

Service specification reference: ServiceSpecification(s) required to realize a ProductSpecification.

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

name A string. Name of the related entity.

href An uri (Uri). Hyperlink reference.

id A string. unique identifier.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type A string. When sub-classing, this defines the sub-class Extensible name.

version A string. Service specification version.

Json representation sample

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

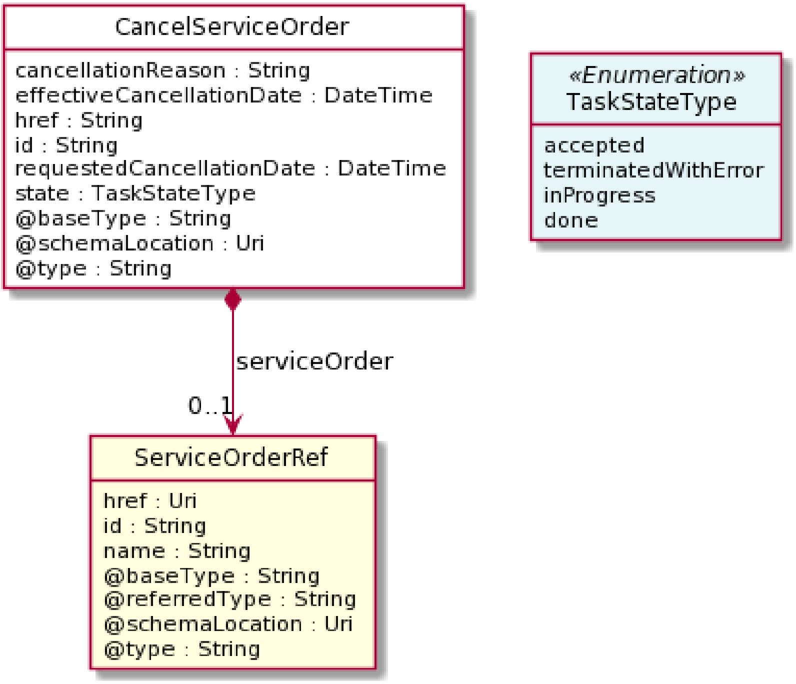
```
{
    "id": "42",
    "href": "http://serverlocation:port/serviceOrdering/v4/serviceOrder/42",
    "externalId": "BSS747",
    "priority": "1",
    "description": "Service order description",
    "category": "TMF resource illustration",
    "state": "acknowledged",
    "orderDate": "2018-01-12T09:37:40.508Z",
    "completionDate": "",
    "requestedStartDate": "2018-01-15T09:37:40.508Z",
    "requestedCompletionDate": "2018-01-15T09:37:40.508Z",
    "expectedCompletionDate": "2018-01-15T09:37:40.508Z",
    "startDate": "2018-01-12T09:37:40.508Z",
    "@type": "ServiceOrder",
    "note": [
        {
            "date": "2018-01-15T09:37:40.508Z",
            "author": "Harvey Poupon",
            "text": "Some text"
        }
    ],
    "relatedParty": [
        {
            "id": "456",
            "href": "http://serverlocation:port/partyManagement/v4/party/456",
            "role": "requester",
            "name": "Jean Pontus",
            "@referredType": "Individual"
        }
    ],
    "serviceOrderItem": [
        {
            "id": "1",
            "action": "add",
            "state": "acknowledged",
            "service": {
                "@type": "Service",
        "state": "active",
        "serviceType": "CFS",
        "serviceCharacteristic": [
          {
            "id": "452-gh6",
            "name": "vCPE",
            "valueType": "object",
            "value": {
              "@type": "vCPE",
              "@schemaLocation": "http://nbi/api/v4/serviceSpecification/ONAPvCPE_Spec/specificationInputSchema",
              "vCPE_IP": "193.218.236.21",
              "MaxTxRate": 300,
              "TransmitPower": "11 dBm",
              "maxTream": "OFF"
            }
          }
        ],
        "serviceSpecification": {
          "@type": "ONAPServiceSpecificationRef",
          "@schemaLocation": "http://my.schema/ONAPServiceSpecificationRef.json",
          "id": "ONAPvCPE_Spec",
          "href": "http://../serviceSpecification/ONAPvCPE_Spec",
          "name": "vCPE",
          "version": "1",
          "invariantUUID": "456-852-357",
          "toscaModelURL": "http//..."
        }
      },
      "@type": "ServiceOrderItem"
    },
    {
      "id": "2",
      "action": "modify",
      "state": "acknowledged",
      "service": {
        "id": "456",
        "@type": "Service",
        "href": "http://serverlocation:port/serviceInventoryManagement/v4/service/456",
        "state": "active",
        "serviceType": "CFS",
        "serviceCharacteristic": [
          {
            "id": "452-gh6",
            "valueType": "string",
            "name": "InputPort",
            "value": "456/7"
          }
        ],
        "supportingResource": [
          {
            "id": "3456_DFG5-H690",
            "href": "http://...",
            "@referredType": "CloudResource"
          }
        ]
      },
      "@type": "ServiceOrderItem"
    },
    {
      "id": "3",
      "action": "add",
      "state": "acknowledged",
      "service": {
        "state": "active",
        "serviceType": "CFS",
        "@type": "Service",
        "serviceRelationship": [
          {
            "relationshipType": "reliesOn",
            "service": {
              "href": "https://.../serviceManagement/v4/service/45",
              "id": "45"
            }
          }
        ],
        "serviceSpecification": {
          "id": "48",
          "href": "http//..../48",
          "name": "genericDataService48",
          "version": "2"
        },
        "feature": [
          {
            "id": "Feat1",
            "isEnabled": true,
            "name": "DataDownload",
            "@type": "Feature",
            "featureCharacteritic": [
              {
                "name": "VolumeCapped",
                "id": "444",
                "value": true,
                "valueType": "boolean"
              }
            ]
          }
        ]
      },
      "@type": "ServiceOrderItem"
    },
    {
      "id": "4",
      "action": "modify",
      "state": "acknowledged",
      "service": {
        "id": "12",
        "href": "http://serverlocation:port/serviceInventoryManagement/v4/service/12",
        "state": "inactive"
      },
      "@type": "ServiceOrderItem"
    }
  ]
```

```
}
```

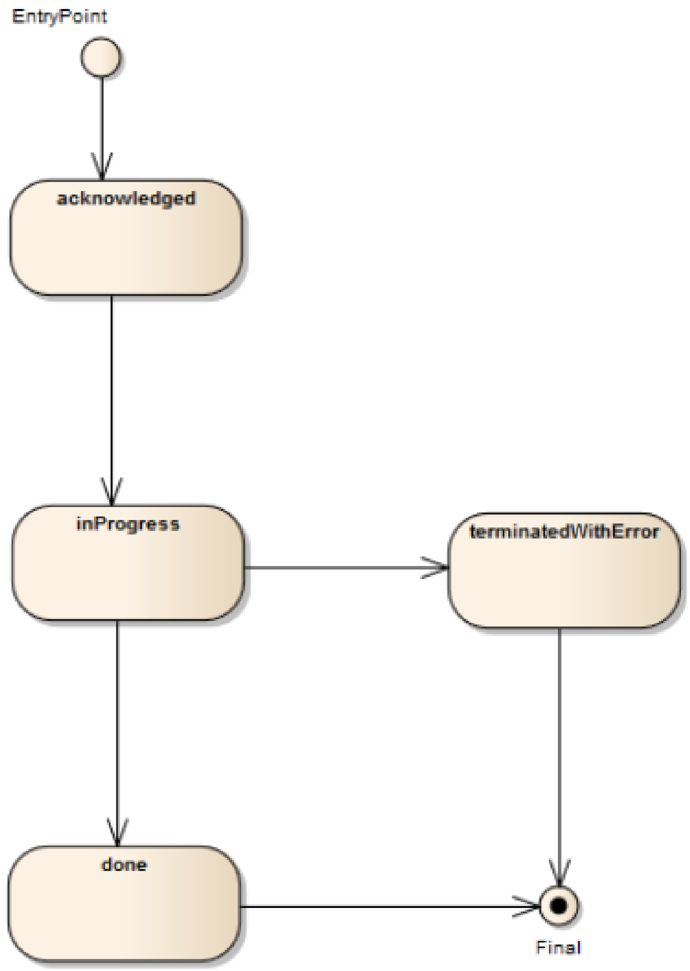
Cancel Service Order resource

Request for cancellation an existing Service order.

Resource model



Lifecycle



CancelOrder State definition:

State	Definition
InProgress	The *In Progress* state is when the CancelOrder is currently in the hands of server side to assess cancellation request
Acknowledged	The acknowledged state is where a CancelOrder request has been received and has passed message and basic business validations. Everything should be updatable in this state
TerminatedWith Error	The CancelOrder could not be correctly completed by Service Order component. It did not mean that the cancel request is denied – it means that the assessment was not be done
Done	CancelOrder assessment has been done. The Order has been cancelled or keep is as it if cancel request denied.

Field descriptions

CancelServiceOrder fields

cancellationReason A string. Reason why the order is cancelled.

effectiveCancellationDate A date time (DateTime). Date when the order is cancelled.

href A string. Hyperlink to access the cancellation request.

id A string. id of the cancellation request (this is not an order id).

requestedCancellationDate A date time (DateTime). Date when the submitter wants the order to be cancelled.

serviceOrder A service order reference (ServiceOrderRef). Useful to understand the which was the Service order through which the service was instantiated in the service inventory.

state A task state type (TaskStateType). Tracks the lifecycle status of the cancellation request, such as Acknowledged, Rejected, InProgress, Pending and so on.

ServiceOrderRef relationship

Service Order reference. Useful to understand the which was the Service order through which the service was instantiated in the service inventory.

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

name A string. Name of the related service order.

href An uri (Uri). Hyperlink reference.

id A string. unique identifier.

@baseType A string. When sub-classing, this defines the super-class.

@schemaLocation An uri (Uri). A URI to a JSON-Schema file that defines additional attributes and relationships.

@type A string. When sub-classing, this defines the sub-class Extensible name.

Json representation sample

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

```
{
  "id": "789-fsds5-kjp",
  "href": "https://host:port/serviceOrdering/v4/cancelServiceOrder/789-fsds5-kjp",
  "cancellationReason": "Duplicate service order",
  "effectiveCancellationDate": "2019-04-30T12:58:35.489Z",
  "requestedCancellationDate": "2019-04-30T12:56:21.931Z",
  "serviceOrder": {
    "id": "45f-98f-ss45",
    "href": "https://host:port/serviceOrdering/v4/serviceOrder/45f-98f-ss45",
    "@referredType": "ServiceOrder"
  },
  "state": "done",
  "@type": "CancelServiceOrder"
}
```

Notification Resource Models

8 notifications are defined for this API

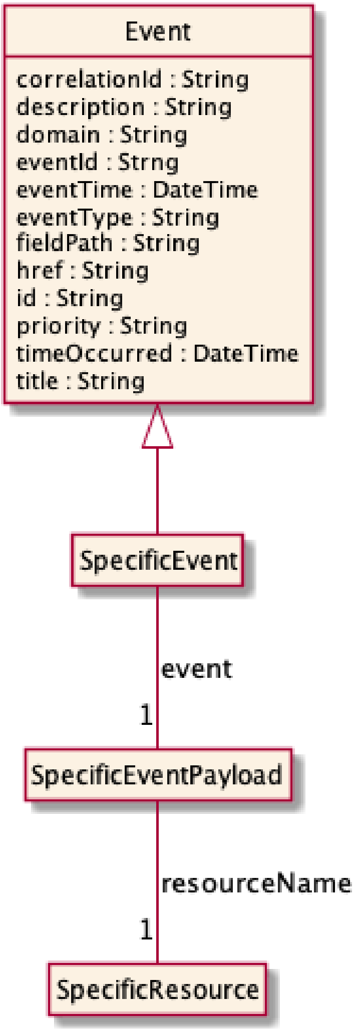
Notifications related to ServiceOrder:

- ServiceOrderCreateEvent
- ServiceOrderAttributeValueChangeEvent
- ServiceOrderStateChangeEvent
- ServiceOrderDeleteEvent
- ServiceOrderInformationRequiredEvent

Notifications related to CancelServiceOrder:

- CancelServiceOrderCreateEvent
- CancelServiceOrderStateChangeEvent
- CancelServiceOrderInformationRequiredEvent

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



Service Order Create Event

Notification ServiceOrderCreateEvent case for resource ServiceOrder

Json representation sample

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

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

Service Order Attribute Value Change Event

Notification ServiceOrderAttributeValueChangeEvent case for resource ServiceOrder

Json representation sample

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

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

Service Order State Change Event

Notification ServiceOrderStateChangeEvent case for resource ServiceOrder

Json representation sample

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

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

Service Order Delete Event

Notification ServiceOrderDeleteEvent case for resource ServiceOrder

Json representation sample

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

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

Service Order Information Required Event

Notification ServiceOrderInformationRequiredEvent case for resource ServiceOrder

Json representation sample

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

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

Cancel Service Order Create Event

Notification CancelServiceOrderCreateEvent case for resource CancelServiceOrder

Json representation sample

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

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

Cancel Service Order State Change Event

Notification CancelServiceOrderStateChangeEvent case for resource CancelServiceOrder

Json representation sample

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

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

Cancel Service Order Information Required Event

Notification CancelServiceOrderInformationRequiredEvent case for resource CancelServiceOrder

Json representation sample

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

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

API OPERATIONS

Remember the following Uniform Contract:

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

List service orders

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

Description

This operation list service order entities.
Attribute selection is enabled for all first level attributes.
Filtering may be available depending on the compliance level supported by an implementation.

Usage Samples

Here's an example of a request for retrieving all the service orders for a given customer that were completed before a specified date

Request

```
GET /tmf-api/serviceOrdering/v4/serviceOrder?fields=id,href,expectedCompletionDate&state=completed&category=TMFresourceillustration&completionDate.gt=2018-01-14T00:00:00.000Z
Accept: application/json
```

Response

```
200

[
  {
    "id": "6987",
    "href": ".../serviceOrdering/v4/serviceOrder/6987",
    "expectedCompletionDate": "2019-04-11T14:52:21.823Z"
  },
  {
    "id": "7412",
    "href": ".../serviceOrdering/v4/serviceOrder/7412",
    "expectedCompletionDate": "2019-04-18T14:21:31.325Z"
  },
  {
    "id": "3214",
    "href": ".../serviceOrdering/v4/serviceOrder/3214",
    "expectedCompletionDate": "2019-04-21T18:08:31.325Z"
  },
  {
    "id": "6547",
    "href": ".../serviceOrdering/v4/serviceOrder/6547",
    "expectedCompletionDate": "2019-04-29T10:48:28.325Z"
  }
]
```

Retrieve service order

```
GET /serviceOrder/{id}?fields=...&{filtering}
```

Description

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

Usage Samples

Here's a sample of a request for retrieving a ServiceOrder resource based on its id

Request

```
GET /tmf-api/serviceOrdering/v4/serviceOrder/43
Accept: application/json
```

Response

```
200

{
  "id": "43",
  "href": "http://serverlocation:port/serviceOrdering/v4/serviceOrder/43",
  "externalId": "BSS747",
  "priority": "1",
  "description": "Service order description",
  "category": "TMF resource illustration",
  "state": "acknowledged",
  "orderDate": "2018-01-12T09:37:40.508Z",
  "requestedStartDate": "2018-01-15T09:37:40.508Z",
  "requestedCompletionDate": "2018-01-15T09:37:40.508Z",
  "expectedCompletionDate": "2018-01-15T09:37:40.508Z",
  "startDate": "2018-01-12T09:37:40.508Z",
  "@type": "ServiceOrder",
  "note": [
    {
      "date": "2018-01-15T09:37:40.508Z",
      "id": "1",
      "author": "Harvey Poupon",
      "text": "Some text"
    }
  ],
  "relatedParty": [
    {
      "id": "456",
      "href": "http://serverlocation:port/partyManagement/v4/party/456",
      "role": "requester",
      "name": "Jean Pontus",
      "@referredType": "Individual"
    }
  ],
  "serviceOrderItem": [
    {
      "id": "1",
      "action": "add",
      "state": "acknowledged",
      "service": {
        "@type": "Service",
        "state": "active",
        "serviceType": "CFS",
        "serviceCharacteristic": [
          {
            "id": "452-gh6",
            "name": "vCPE",
            "valueType": "object",
            "value": {
              "@type": "JSONSpecification",
              "@schemaLocation": "http://my.schemas/vCPE.schema.json",
              "vCPE_IP": "193.218.236.21",
              "MaxTxRate": 300,
              "TransmitPower": "11 dBm",
              "maxTream": "OFF"
            }
          }
        ],
        "serviceSpecification": {
          "@type": "ONAPServiceSpecification",
          "@baseType": "ServiceSpecificationRef",
          "@schemaLocation": "http://my.schema/ONAPServiceSpecification.json",
          "id": "12",
          "href": "http://.../serviceSpecification/12",
          "name": "vCPE",
          "version": "1",
          "invariantUUID": "456-852-357",
          "toscaModelURL": "http//...",
          "targetServiceSchema": {
            "@type": "onapJsonScheam",
            "@schemaLocation": "http://nbi/api/v4/serviceSpecification/456-852-357/specificationInputSchema"
          }
        }
      },
      "@type": "ServiceOrderItem"
    },
    {
      "id": "2",
      "action": "modify",
      "state": "acknowledged",
      "service": {
        "id": "456",
        "@type": "Service",
        "href": "http://serverlocation:port/serviceInventoryManagement/v4/service/456",
        "state": "active",
        "serviceType": "CFS",
        "serviceCharacteristic": [
          {
            "id": "452-gh6",
            "name": "InputPort",
            "valueType": "string",
            "value": "456/7"
          }
        ],
        "supportingResource": [
          {
            "id": "3456_DFG5-H690",
            "href": "http://serverlocation:port/resourceInventoryManagement/v4/resource/3456_DFG5-H690",
            "@referredType": "CloudResource"
          }
        ]
      },
      "@type": "ServiceOrderItem"
    },
    {
      "id": "3",
      "action": "add",
      "state": "acknowledged",
      "service": {
        "state": "active",
        "serviceType": "CFS",
        "@type": "Service",
        "serviceRelationship": [
          {
            "relationshipType": "reliesOn",
            "service": {
              "href": "https://serverlocation:port/serviceManagement/v4/service/45",
              "id": "45"
            }
          }
        ],
        "serviceSpecification": {
          "id": "48",
          "href": "http//serverlocation:port/serviceCatalogManagement/v4/serviceSpecification/48",
          "name": "genericDataService48",
          "version": "2"
        },
        "feature": [
          {
            "id": "Feat1",
            "isEnabled": true,
            "name": "DataDownload",
            "@type": "Feature",
            "featureCharacteristic": [
              {
                "name": "VolumeCapped",
                "value": true,
                "id": "444",
                "valueType": "boolean"
              }
            ]
          }
        ]
      },
      "@type": "ServiceOrderItem"
      },
      {
        "id": "4",
        "action": "modify",
        "state": "acknowledged",
        "service": {
          "id": "12",
          "href": "http://serverlocation:port/serviceInventoryManagement/v4/service/12",
          "state": "inactive"
        },
        "@type": "ServiceOrderItem"
      }
    ]
}
```

Create service order

POST /serviceOrder

Description

This operation creates a service order entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a ServiceOrder, 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
serviceOrderItem	

Non Mandatory Attributes	Rule
cancellationDate	
cancellationReason	
category	
completionDate	
description	
expectedCompletionDate	
externalId	
externalReference	
note	
notificationContact	
orderDate	
orderRelationship	
priority	
relatedParty	
requestedCompletionDate	
requestedStartDate	

Non Mandatory Attributes	Rule
startDate	
state	

Additional Rules

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

Context	Mandatory Sub-Attributes
serviceOrderItem	id, action, service
relatedParty	id, @referredType
note	text, id
orderRelationship	id, relationshipType
serviceOrderItem.service.supportingResource	id
serviceOrderItem.service.serviceCharacteristic	name
serviceOrderItem.serviceOrderItemRelationship	orderItem
serviceOrderItem.serviceOrderItemRelationship.orderItem	itemId
appointment	id
externalReference	name

The following pre-conditions apply for this operation.

Pre-conditions
When creating a service order (post) an order must not have the state, cancellationDate, cancellationReason, completionDate, orderDate fields valued
When creating a service order (post) an order item must not have the state field valued
In relatedParty, the role is mandatory if the id identified a party - optional if it is a party role.

Usage Samples

Here's a sample of a request for creating a ServiceOrder resource. We use the ‘fields’ parameter to restrict the numbers of attributes returned in the response.

Request

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

{
  "externalId": "BSS748",
  "priority": "1",
  "description": "Service order description",
  "category": "TMF resource illustration",
  "requestedStartDate": "2018-01-15T09:37:40.508Z",
  "requestedCompletionDate": "2018-01-15T09:37:40.508Z",
  "@type": "ServiceOrder",
  "serviceOrderItem": [
    {
      "id": "1",
      "action": "add",
      "@type": "ServiceOrderItem",
      "service": {
        "serviceState": "active",
        "type": "CFS",
        "serviceCharacteristic": [
          {
            "name": "vCPE",
            "valueType": "object",
            "value": {
              "@type": "JSONSpecification",
              "@schemaLocation": "http://nbi/api/v4/serviceSpecification/ONAPvCPE_Spec/specificationInputSchema",
              "vCPE_IP": "193.218.236.21",
              "MaxTxRate": 300,
              "TransmitPower": "11 dBm",
              "maxTream": "OFF"
            }
          }
        ],
        "serviceSpecification": {
          "id": "ONAPvCPE_Spec",
          "href": "http://...:serviceSpecification/ONAPvCPE_Spec",
          "name": "vCPE",
          "version": "1",
          "@type": "vCPE"
        }
      }
    }
  ]
}
```

Response

```
201

{
  "id": "5351",
  "href": "https://host:port/serviceOrder/v4/serviceOrder/5351"
}
```

Patch service order

PATCH /serviceOrder/{id}

Description

This operation allows partial updates of a service order entity. Support of json/merge (https://tools.ietf.org/html/rfc7386) is mandatory, support of json/patch (http://tools.ietf.org/html/rfc5789) is optional.

Note: If the update operation yields to the creation of sub-resources or relationships, the same rules concerning mandatory sub-resource attributes and default value settings in the POST operation applies to the PATCH operation. Hence these tables are not repeated here.

Patchable and Non Patchable Attributes

The tables below provide the list of patchable and non patchable attributes, including constraint rules on their usage.

Patchable Attributes	Rule
cancellationDate	
cancellationReason	
category	
completionDate	
description	
expectedCompletionDate	
externalId	
externalReference	
note	
notificationContact	
orderRelationship	
priority	
relatedParty	
requestedCompletionDate	
requestedStartDate	
serviceOrderItem	
startDate	
state	

Non Patchable Attributes	Rule
id	
href	
orderDate	

Additional Rules

The following pre-conditions apply for this operation.

Pre-conditions
When creating a service order (post) an order must not have the state, cancellationDate, cancellationReason, completionDate, orderDate fields valued

When creating a service order (post) an order item must not have the state field valued
In relatedParty, the role is mandatory if the id identified a party - optional if it is a party role.

Usage Samples

Here's an example of a request for patching a ServiceOrder resource (created via previous POST) with change on following attributes: description, requestedStartDate, requestedCompletionDate, vCPE IP (characteristic) value

Request

```
PATCH /tmf-api/serviceOrdering/v4/serviceOrder/42
Content-Type: application/merge-patch+json

{
  "description": "Service order new description updated v2",
  "requestedStartDate": "2018-01-16T09:37:40.508Z",
  "requestedCompletionDate": "2018-01-17T09:37:40.508Z",
  "orderItem": [
    {
      "id": "1",
      "action": "add",
      "service": {
        "serviceCharacteristic": [
          {
            "name": "vCPE",
            "valueType": "object",
            "value": {
              "@type": "JSONSpecification",
              "@schemaLocation": "http://nbi/api/v4/serviceSpecification/ONAPvCPE_Spec/specificationInputSchema",
              "vCPE_IP": "193.218.236.26",
              "MaxTxRate": 300,
              "TransmitPower": "11 dBm",
              "maxTream": "OFF"
            }
          }
        ]
      }
    }
  ]
}
```

Response

```
200

{
  "id": "5351",
  "href": "http://serverlocation:port/serviceOrdering/v4/serviceOrder/5351",
  "externalId": "BSS747",
  "priority": "1",
  "category": "TMF resource illustration",
  "state": "acknowledged",
  "orderDate": "2018-01-12T09:37:40.508Z",
  "completionDate": "",
  "description": "Service order new description updated v2",
  "requestedStartDate": "2018-01-16T09:37:40.508Z",
  "requestedCompletionDate": "2018-01-17T09:37:40.508Z",
  "expectedCompletionDate": "2018-01-15T09:37:40.508Z",
  "startDate": "2018-01-12T09:37:40.508Z",
  "@type": "ServiceOrder",
  "note": [
    {
      "date": "2018-01-15T09:37:40.508Z",
      "author": "Harvey Poupon",
      "text": "Some text"
    }
  ],
  "relatedParty": [
    {
      "id": "456",
      "href": "http://serverlocation:port/partyManagement/v4/party/456",
      "role": "requester",
      "name": "Jean Pontus",
      "@referredType": "Individual"
    }
  ],
  "orderItem": [
    {
      "id": "1",
      "action": "add",
      "state": "acknowledged",
      "service": {
        "@type": "Service",
        "state": "active",
        "serviceType": "CFS",
        "serviceCharacteristic": [
          {
            "id": "452-gh6",
            "name": "vCPE",
            "valueType": "object",
            "value": {
              "@type": "vCPE",
              "@schemaLocation": "http://nbi/api/v4/serviceSpecification/ONAPvCPE_Spec/specificationInputSchema",
              "vCPE_IP": "193.218.236.26",
              "MaxTxRate": 300,
              "TransmitPower": "11 dBm",
              "maxTream": "OFF"
            }
          }
        ],
        "serviceSpecification": {
          "@type": "ONAPServiceSpecificationRef",
          "@schemaLocation": "http://my.schema/ONAPServiceSpecificationRef.json",
          "id": "ONAPvCPE_Spec",
          "href": "http://../serviceSpecification/ONAPvCPE_Spec",
          "name": "vCPE",
          "version": "1",
          "invariantUUID": "456-852-357",
          "toscaModelURL": "http//..."
        }
      },
      "@type": "ServiceOrderItem"
    }
  ]
}
```

Here's an example of a request for patching a ServiceOrder resource (created via previous POST) with change on following attributes: description, requestedStartDate, requestedCompletionDate, vCPE IP (characteristic) value

Request

```
PATCH /tmf-api/serviceOrdering/v4/serviceOrder/42
Content-Type: application/json-patch+json

[
  {
    "op": "replace",
    "path": "requestedStartDate",
    "value": "2018-01-16T09:37:40.508Z"
  },
  {
    "op": "replace",
    "path": "description",
    "value": "Service order new description updated v2"
  },
  {
    "op": "replace",
    "path": "requestedCompletionDate",
    "value": "2018-01-17T09:37:40.508Z"
  },
  {
    "op": "replace",
    "path": "orderItem/1/service/serviceCharacteristic/value/",
    "value": {
      "@type": "JSONSpecification",
      "@schemaLocation": "http://nbi/api/v4/serviceSpecification/ONAPvCPE_Spec/specificationInputSchema",
      "vCPE_IP": "193.218.236.26",
      "MaxTxRate": 300,
      "TransmitPower": "11 dBm",
      "maxTream": "OFF"
    }
  }
]
```

Response

```
200

{
   "id": "5351",
   "href": "http://serverlocation:port/serviceOrdering/v4/serviceOrder/5351",
   "externalId": "BSS747",
   "priority": "1",
   "category": "TMF resource illustration",
   "state": "acknowledged",
   "orderDate": "2018-01-12T09:37:40.508Z",
   "completionDate": "",
   "description": "Service order new description updated v2",
   "requestedStartDate": "2018-01-16T09:37:40.508Z",
   "requestedCompletionDate": "2018-01-17T09:37:40.508Z",
   "expectedCompletionDate": "2018-01-15T09:37:40.508Z",
   "startDate": "2018-01-12T09:37:40.508Z",
   "@type": "ServiceOrder",
   "note": [
      {
         "date": "2018-01-15T09:37:40.508Z",
         "author": "Harvey Poupon",
         "text": "Some text"
      }
   ],
   "relatedParty": [
      {
         "id": "456",
         "href": "http://serverlocation:port/partyManagement/v4/party/456",
         "role": "requester",
         "name": "Jean Pontus",
         "@referredType": "Individual"
      }
   ],
   "orderItem": [
      {
         "id": "1",
         "action": "add",
         "state": "acknowledged",
         "service": {
            "@type": "Service",
            "state": "active",
            "serviceType": "CFS",
            "serviceCharacteristic": [
               {
                  "id": "452-gh6",
                  "name": "vCPE",
                  "valueType": "object",
                  "value": {
                     "@type": "vCPE",
                     "@schemaLocation": "http://nbi/api/v4/serviceSpecification/ONAPvCPE_Spec/specificationInputSchema",
                     "vCPE_IP": "193.218.236.26",
                     "MaxTxRate": 300,
                     "TransmitPower": "11 dBm",
                     "maxTream": "OFF"
```

```
                    }
                }
            ],
            "serviceSpecification": {
                "@type": "ONAPServiceSpecificationRef",
                "@schemaLocation": "http://my.schema/ONAPServiceSpecificationRef.json",
                "id": "ONAPvCPE_Spec",
                "href": "http://../serviceSpecification/ONAPvCPE_Spec",
                "name": "vCPE",
                "version": "1",
                "invariantUUID": "456-852-357",
                "toscaModelURL": "http//..."
            }
        },
        "@type": "ServiceOrderItem"
    }
  ]
}
```

Delete service order

DELETE /serviceOrder/{id}

Description

This operation deletes a service order entity.

Usage Samples

Here's an example of a request for deleting a Service order.

Request
DELETE /tmf-api/serviceOrdering/v4/serviceOrder/47
Response
204

Operations on Cancel Service Order

List cancel service orders

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

Description

This operation list cancel service order entities.
Attribute selection is enabled for all first level attributes.
Filtering may be available depending on the compliance level supported by an implementation.

Usage Samples

Here's an example of a request for retrieving a list of Service order cancel request(s). The given criteria is the cancel request state (done).

Request

```
GET /tmf-api/serviceOrdering/v4/cancelServiceOrder?fields=id,ServiceOrder.id,ServiceOrder.href&state=done
Accept: application/json
```

Response

```
200

[
  {
    "id": "6987",
    "serviceOrder": {
      "id": "45f-98f-ss45",
      "href": "https://host:port/serviceOrdering/v4/serviceOrder/45f-98f-ss45"
    }
  },
  {
    "id": "7412",
    "serviceOrder": {
      "id": "45f-789-sdre",
      "href": "https://host:port/serviceOrdering/v4/serviceOrder/45f-789-sdre"
    }
  },
  {
    "id": "3214",
    "serviceOrder": {
      "id": "45f-as4-os3f",
      "href": "https://host:port/serviceOrdering/v4/serviceOrder/45f-as4-os3f"
    }
  }
]
```

Retrieve cancel service order

```
GET /cancelServiceOrder/{id}?fields=...&{filtering}
```

Description

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

Usage Samples

Here's an example of a cancel Service order request retrieval. The given criteria is the Service order request cancel id 789-fsds5-kjp.

Request

```
GET /tmf-api/serviceOrdering/v4/cancelServiceOrder/789-fsds5-kjp
Accept: application/json
```

Response

```
200

{
   "id": "789-fsds5-kjp",
   "href": "https://host:port/serviceOrdering/v4/cancelserviceOrder/789-fsds5-kjp",
   "cancellationReason": "Duplicate service order",
   "effectiveCancellationDate": "2019-04-30T12:58:35.489Z",
   "requestedCancellationDate": "2019-04-30T12:56:21.931Z",
   "serviceOrder": {
      "id": "45f-98f-ss45",
      "href": "https://host:port/serviceOrdering/v4/serviceOrder/45f-98f-ss45",
      "@referredType": "ServiceOrder"
   },
   "state": "done",
   "@type": "CancelServiceOrder"
}
```

Create cancel service order

POST /cancelServiceOrder

Description

This operation creates a cancel service order entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a CancelServiceOrder, 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
ServiceOrder	

Non Mandatory Attributes	Rule
cancellationReason	
effectiveCancellationDate	
requestedCancellationDate	
serviceOrder	
state	

Additional Rules

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

Context	Mandatory Sub-Attributes
ServiceOrder	id

The following pre-conditions apply for this operation.

Pre-conditions
service order id must exist
When creating a cancel service order (post) the id, href, effectiveCancellationDate and state fields must not be valued in the request

Usage Samples

Here's an example of a request for cancel a Service order - the resulting Service order cancellation request is 789-fsds5-kjp.

Request

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

{
   "cancellationReason": "Duplicate service order",
   "requestedCancellationDate": "2019-04-30T12:56:21.931Z",
   "serviceOrder": {
      "id": "45f-98f-ss45",
        "@referredType": "ServiceOrder"
    },
    "@type": "CancelServiceOrder"
}
```

Response

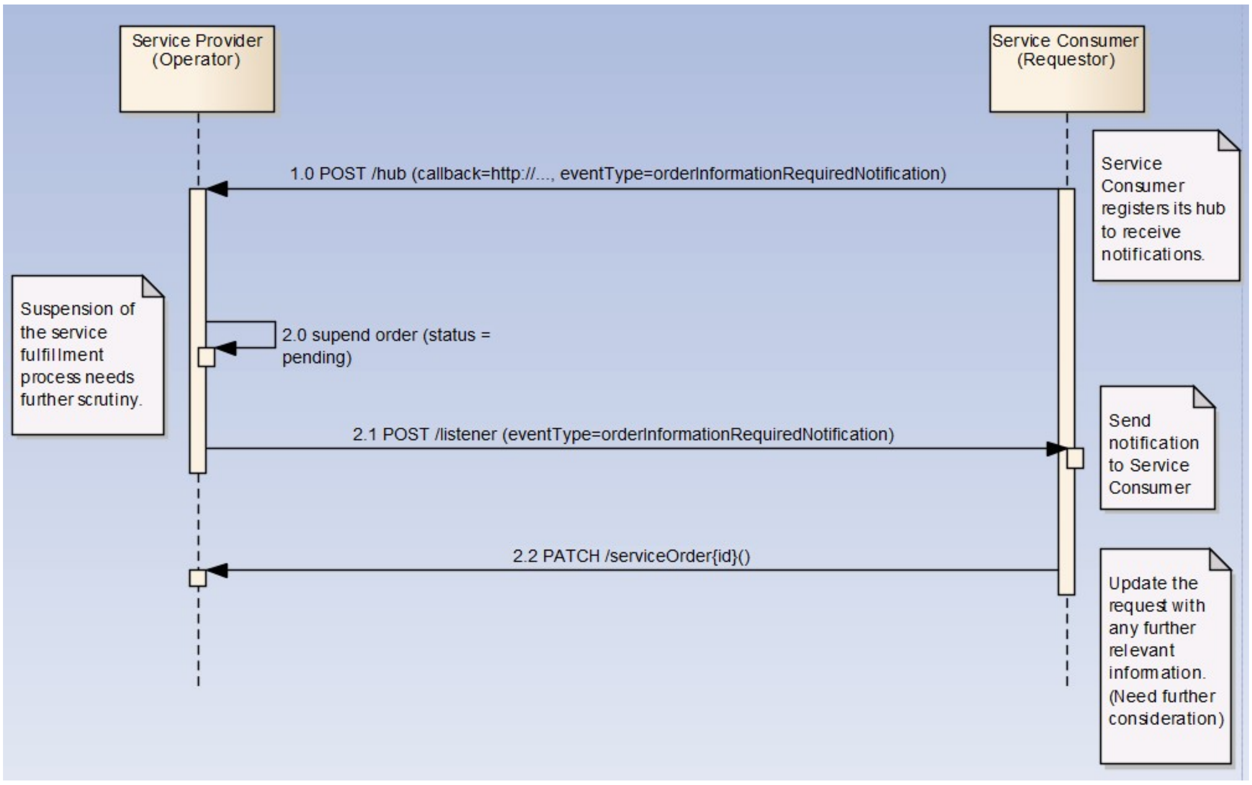
```
201

{
    "id": "789-fsds5-kjp",
    "href": "https://host:port/serviceOrdering/v4/cancelserviceOrder/789-fsds5-kjp",
    "cancellationReason": "Duplicate service order",
    "requestedCancellationDate": "2019-04-30T12:56:21.931Z",
    "serviceOrder": {
        "id": "45f-98f-ss45",
        "href": "https://host:port/serviceOrdering/v4/serviceOrder/45f-98f-ss45",
        "@referredType": "ServiceOrder"
    },
    "state": "done",
    "@type": "CancelServiceOrder"
}
```

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.


Service Provider (Operator)
Service Consumer (Requestor)
1.0 POST /hub (callback=http://..., eventType=orderInformationRequiredNotification)
Service Consumer registers its hub to receive notifications.
Suspension of the service fulfillment process needs further scrutiny.
2.0 supend order (status = pending)
2.1 POST /listener (eventType=orderInformationRequiredNotification)
Send notification to Service Consumer
2.2 PATCH /serviceOrder{id}()
Update the request with any further relevant information. (Need further consideration)

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/42 Accept: application/json
Response
204

Publish Event to listener

POST /client/listener

Description

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

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

Behavior

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

Usage Samples

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

Request

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

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

Response

```
201
```

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

Acknowledgements

Version History

Release Number	Date	Release led by:	Description
1.0.0		Pierre Gauthier TM Forum pgauthier@tmforum.org Andrew Forth aforth@amdocs.com August-Wilhelm Jagau August-wilhelm.jagau@ericsson.com	First Release of the Document.
2.0.0	24/10/2016	Pierre Gauthier TM Forum pgauthier@tmforum.org Ludovic Robert Orange ludovic.robert@orange.com Nicoleta Stoica Vofadone nicoleta.stoica@vodafone.com Jean-Luc Tymen Orange jeanluc.tymen@orange.com	Alignment with Guidelines 3.0
3.0	Dec 2018	Pierre Gauthier TM Forum pgauthier@tmforum.org Ludovic Robert Orange ludovic.robert@orange.com Mariano Belaunde Orange mariano.belaunde@orange.com	
4.0.0	28-May-2020	Pierre Gauthier TM Forum pgauthier@tmforum.org Ludovic Robert Orange ludovic.robert@orange.com	

Release History

Release Number	Date	Release led by:	Description
Pre-production	28-May-2020	Pierre Gauthier TM Forum Ludovic Robert Orange	

Contributors to Document
