

TM Forum Specification

Shopping Cart API REST Specification

TMF663
Release 18.0.0
June 2018

Latest Update: TM Forum Release 18.0.0	Member Evaluation
Version 2.0.1	IPR Mode: RAND

NOTICE

Copyright © TM Forum 2018. All Rights Reserved.

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

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

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

Direct inquiries to the TM Forum office:

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

TABLE OF CONTENTS

NOTICE.....	2
TABLE OF CONTENTS	3
LIST OF TABLES	4
INTRODUCTION	5
SAMPLE USE CASES.....	6
Support of polymorphism and extension patterns	10
RESOURCE MODEL.....	11
Managed Entity and Task Resource Models.....	11
Shopping Cart resource	11
Notification Resource Models	24
Shopping Cart Creation Notification.....	25
Shopping Cart Change Notification	26
API OPERATIONS.....	27
Operations on Shopping Cart	28
List shopping carts	28
Retrieve shopping cart	30
Create shopping cart.....	32
Patch shopping cart	36
Delete shopping cart.....	39
API NOTIFICATIONS	40
Register listener.....	40
Unregister listener.....	41
Publish Event to listener.....	41
ACKNOWLEDGEMENTS	43
Version History	43
Release History	43
Contributors to Document	44

LIST OF TABLES

N/A

INTRODUCTION

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

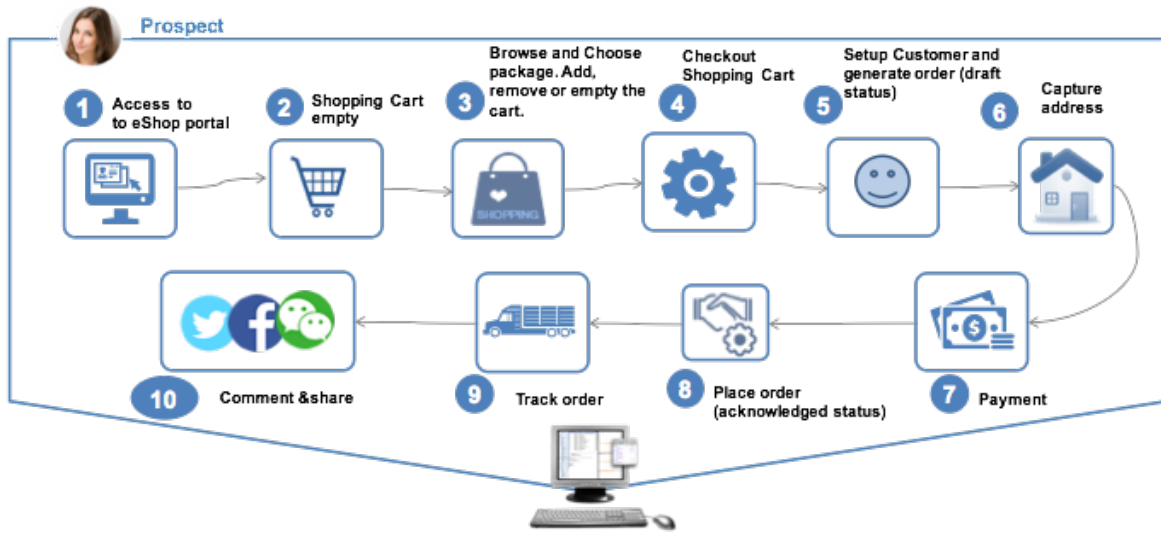
Shopping Cart is the necessary component in modern sales channel both for online channel of e-commerce and call center and physical stores. It allows visitors to the shopping site to select items for eventual purchase. It provides a means of capturing a user purchase requirement, to accumulate a list of items for purchase, described as “placing items in the shopping cart” or “adding to cart.” It typically calculates a total charge for the cart including chosen goods or service, shipping and handling charges and the associated taxes are typically calculated during the checkout. Shopping cart supports purchase of both tangible and intangible good and service (e.g. mobile device or family plan). The charge includes the one-off fee such as the fee for handset and the recurring fee. The customer will be informed about the fee information, but only pay the one-off fee after confirming the shopping cart. The recurring fee may be paid when reaching the bill cycle.

Shopping Cart Item is a contained sub resource. Cart item is the product offering which has been selected and added into the shopping cart with its price calculation. This item represents the customer’s request to order, but not finally decided.

The Shopping Cart API provides standardized mechanism for the management of shopping carts. Including creation, update, retrieval, deletion and notification of event. Shopping Cart entity is used for the temporarily selection and reservation of product offerings in e-commerce and retail purchase. Shopping cart supports purchase of both tangible and intangible goods and service (e.g. handset, telecom network service). The charge includes the one-off fee such as the fee for handset and the recurring fee such as the fee of a network service. Shopping Cart contains list of cart items, a reference to party or party role (e.g. customer) or contact medium in case of unknown customer, in addition the calculated total items price including promotions.

SAMPLE USE CASES

Shopping Cart - Prospect

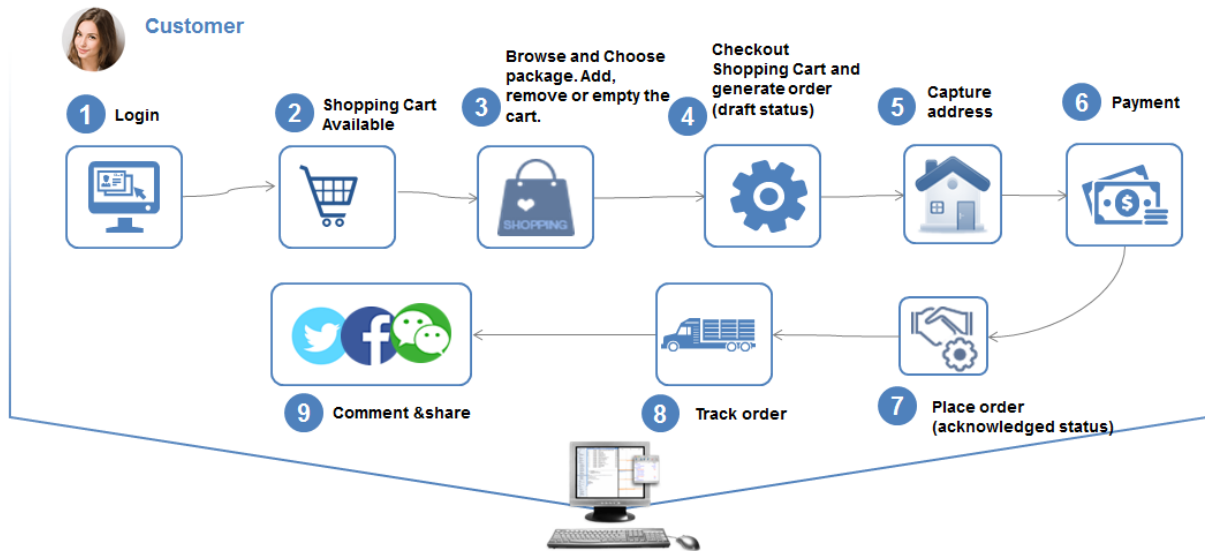


Use Case Id	UC_TMF_ShoppingCart_0001
Use Case Name	Prospect Purchase Product Offering using Shopping Cart
Summary	This case describes the prospect selects and purchases the offering using the shopping cart, and finally submits the request to generate the order.
Actor(s)	<ol style="list-style-type: none"> Prospect (person, not yet a customer) Online Shopping Portal (system)
Pre-Conditions	<ol style="list-style-type: none"> Online shopping portal is available, including the offerings and integration with external systems such as payment, logistics. Prospect browse the Online Shopping Portal
Begins When	When all pre-conditions have been met and the prospect starts adding the offering into the shopping cart.

Shopping Cart API REST Specification

Use Case Id	UC_TMF_ShoppingCart_0001
Description	<ol style="list-style-type: none">1. The prospect browses the online shopping portal.2. The prospect browses the offering and chooses what he/she wants to purchase, and add it into the shopping cart.3. The shopping cart is created for the prospect with the added items and visible for the prospect to operate on it.4. The customer can also remove the existing item in the cart, or empty it completely.5. The prospect checkout the shopping cart to generate the product order in draft status (Checkout Flow).6. The prospect required to enroll as customer, customer is generated in CRM system. The prospect become a customer7. The customer fills in the delivery address for the order.8. The customer finishes payment.9. The order is submitted and it can be tracked.10. The customer comments the purchase process or share the experience.

Shopping Cart - Customer



Use Case Id	UC_TMF_ShoppingCart_0002
Use Case Name	Customer Purchase Product Offering using Shopping Cart
Summary	This case describes the customer selects and purchases the offering using the shopping cart, and finally submits the request to generate the order.
Actor(s)	<ol style="list-style-type: none"> Customer (person) Online Shopping Portal (system)
Pre-Conditions	<ol style="list-style-type: none"> Online shopping portal is available, including the offerings and integration with external systems such as payment, logistics. Customer has logged into the online shopping portal.
Begins When	When all pre-conditions have been met and the customer starts adding the offering into the shopping cart.
Description	<ol style="list-style-type: none"> The customer logs into the online shopping portal. If the customer has already saved the shopping cart in the previous purchase process during last time of log-in, the shopping cart can be retrieved and reused.

Shopping Cart API REST Specification

Use Case Id	UC_TMF_ShoppingCart_0002
	<ol style="list-style-type: none">3. The customer browses the offering and chooses what he/she wants to purchase, and add it into the shopping cart.4. The shopping cart is created for the customer with the added items and visible for the customer to operate on it.5. The customer can also remove the existing item in the cart, or empty it completely.6. The customer submits the shopping cart to generate the product order in draft status (Checkout Flow)7. The customer fills in the delivery address for the order.8. The customer finishes payment.9. The order is submitted and it can be tracked.10. The customer comments the purchase process or share the experience.

SUPPORT OF POLYMORPHISM AND EXTENSION PATTERNS

Support of polymorphic collections and types and schema based extension is provided by means of a list of generic meta-attributes that we describe below. Polymorphism in collections occurs when entities inherit from base entities, for instance a `BillingAccount` and `SettlementAccount` inheriting properties from the abstract `Account` 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 `Account` instances some may be instances of `BillingAccount` where other could be instances of `SettlementAccount`. The `@type` gives this information. All resources and sub-resources of this API have a `@type` attributes that can be provided when this is useful.

The `@referredType` can be used within reference entities (like for instance an `AccountRef` 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 `BillingAccountRef` or `SettlementAccountRef`, 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

SHOPPING CART RESOURCE

Shopping Cart resource is used for the temporarily selection and reservation of product offerings in e-commerce, call center and retail purchase. Shopping cart supports purchase of both physical and digital goods and service (e.g. handset, telecom network service). Shopping Cart contain list of cart items, a reference to customer (partyRole) or contact medium in case customer not exist, and the total items price including promotions.

Resource model

Shopping Cart API REST Specification

href	A string. Hyperlink to access the shopping cart.
id	A string. Unique identifier created on provider side (e.g. Order Capture system).
validFor	A time period. The period for which the shopping cart is valid (e.g. 90 if no activity or 7 days if cart is empty).
contactMedium	A list of contact mediums (ContactMedium [*]). Indicates the contact medium that could be used to contact the party.
cartTotalPrice	A list of cart prices (CartPrice [*]). Total amount of the shopping cart, usually of money, that represents the actual price paid by the Customer for cart (considering only "Active" cart items).
cartItem	A list of cart items (CartItem [*]). An identified part of the shopping cart. A shopping cart is decomposed into one or more shopping cart item. Cart item represents a product offering or bundled product offering that user wish to purchase, as well as the pricing of the product offering, reference to product in case of configured characteristic or installation address. Cart items can be related to other cart item to related bundled offerings or reference cart Items to a shipping options.
relatedParty	A list of related party references (RelatedPartyRef [*]). A related party defines party or party role linked to a specific entity.

CartItem sub-resource

An identified part of the shopping cart. A shopping cart is decomposed into one or more shopping cart item. Cart item represents a product offering or bundled product offering that user wish to purchase, as well as the pricing of the product offering, reference to product in case of configured characteristic or installation address. Cart items can be related to other cart item to related bundled offerings or reference cart Items to a shipping options.

@baseType	A string. Generic attribute indicating the base class type of the extension class of the current object. Useful only when the class type of the current object is unknown to the implementation.
@schemaLocation	A string. Generic attribute containing the link to the schema that defines the structure of the class type of the current object.
@type	A string. Generic attribute containing the name of the resource class type.
action	A string. Can be "add" / "modify" / "no_change" / "delete".

Shopping Cart API REST Specification

id	A string. Identifier of the cart item (generally it is a sequence number 01, 02, 03, ...) in the shopping cart.
quantity	An integer. Quantity of cart items.
status	A string. status of cart item. e.g. "Active", "SavedForLater".
cartTerm	A list of cart terms (CartTerm [*]).
cartItem	A list of cart items (CartItem [*]). An identified part of the shopping cart. A shopping cart is decomposed into one or more shopping cart item. Cart item represents a product offering or bundled product offering that user wish to purchase, as well as the pricing of the product offering, reference to product in case of configured characteristic or installation address. Cart items can be related to other cart item to related bundled offerings or reference cart Items to a shipping options.
note	A list of notes (Note [*]). Extra information about a given entity.
itemTotalPrice	A list of cart prices (CartPrice [*]). An amount, usually of money, that represents the actual price paid by the customer for this item. May represent the total price of the shopping cart or the total of the cart item depending on the relation.
product	A product (Product). Product reference. Configure the product characteristics (only configurable characteristics and necessary only if a non default value is selected) and/or identify the product that needs to be modified/deleted.
itemPrice	A list of cart prices (CartPrice [*]). An amount, usually of money, that represents the actual price paid by the customer for this item. May represent the total price of the shopping cart or the total of the cart item depending on the relation.
productOffering	A product offering reference (ProductOfferingRef). A product offering represents entities that are orderable from the provider of the catalog, this resource includes pricing information.
cartItemRelationship	A list of cart item relationships (CartItemRelationship [*]). Relationship among cart items mainly other than hierarchical relationships such as "RelyOn", "DependentOn", "Shipping" etc.

CartItemRelationship sub-resource

Relationship among cart items mainly other than hierarchical relationships such as "RelyOn", "DependentOn", "Shipping" etc.

Shopping Cart API REST Specification

id	A string. Unique identifier of the referred cart item.
type	A string. Type of the cart item relationship.
cartItem	A list of cart item references (CartItemRef [*]). CartItem reference. A CartItem is an identified part of the shopping cart.

CartPrice sub-resource

An amount, usually of money, that represents the actual price paid by the customer for this item. May represent the total price of the shopping cart or the total of the cart item depending on the relation.

@baseType	A string. Generic attribute indicating the base class type of the extension class of the current object. Useful only when the class type of the current object is unknown to the implementation.
@schemaLocation	A string. Generic attribute containing the link to the schema that defines the structure of the class type of the current object.
@type	A string. Generic attribute containing the name of the resource class type.
description	A string. A narrative that explains in detail the semantics of this order item price.
name	A string. A short descriptive name such as "Subscription price".
priceType	A string. A category that describes the price, such as recurring, discount, allowance, penalty, and so forth.
recurringChargePeriod	A string. Could be month, week...
unitOfMeasure	A string. Could be minutes, GB...
price	A price (Price). Provides all amounts (tax included, duty free, tax rate), used currency and percentage to apply for Price Alteration.
priceAlteration	A list of price alterations (PriceAlteration [*]). Is an amount, usually of money, that modifies the price charged for an order item.

CartTerm sub-resource

Shopping Cart API REST Specification

@baseType	A string. Generic attribute indicating the base class type of the extension class of the current object. Useful only when the class type of the current object is unknown to the implementation.
@schemaLocation	A string. Generic attribute containing the link to the schema that defines the structure of the class type of the current object.
@type	A string. Generic attribute containing the name of the resource class type.
description	A string.
duration	A quantity (Quantity).
name	A string.

ContactMedium sub-resource

Indicates the contact medium that could be used to contact the party.

@baseType	A string. Generic attribute indicating the base class type of the extension class of the current object. Useful only when the class type of the current object is unknown to the implementation.
@schemaLocation	A string. Generic attribute containing the link to the schema that defines the structure of the class type of the current object.
@type	A string. Generic attribute containing the name of the resource class type.
preferred	A boolean. If true, indicates that is the preferred contact medium.
type	A string. Type of the contact medium, such as: email address, telephone number, postal address.
validFor	A time period. The time period that the contact medium is valid for.
characteristic	A medium characteristic (MediumCharacteristic). Describes the contact medium characteristics that could be used to contact a party (an individual or an organization).

MediumCharacteristic sub-resource

Describes the contact medium characteristics that could be used to contact a party (an individual or an organization).

city	A string. The city.
country	A string. The country.

Shopping Cart API REST Specification

emailAddress	A string. Full email address in standard format.
faxNumber	A string. The fax number of the contact.
phoneNumber	A string. The primary phone number of the contact.
postCode	A string. Postcode.
stateOrProvince	A string. State or province.
street1	A string. Describes the street.
street2	A string. Complementary street description.
type	A string. Type of medium (fax, mobile phone...).

Money sub-resource

A base / value business entity used to represent money.

unit	A string. Currency (ISO4217 norm uses 3 letters to define the currency).
value	A float. A positive floating point number.

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.
text	A string. Text of the note.

Place sub-resource

Place defines a location generically. Could be a geographic address a geographic address, a geographic location.

@baseType	A string. Generic attribute indicating the base class type of the extension class of the current object. Useful only when the class type of the current object is unknown to the implementation.
@referredType	A string. Indicates the type of the referred object. This attribute is to be used when the object is representing a reference to an existing object instead of the of the object itself.
@schemaLocation	A string. Generic attribute containing the link to the schema that defines the structure of the class type of the current object.

Shopping Cart API REST Specification

@type	A string. Generic attribute containing the name of the resource class type.
href	A string. Unique reference of the place.
id	A string. Unique identifier of the place.
name	A string. A user-friendly name for the place, such as "Paris Store", "London Store", "Main Home".
role	A string. Role of the place (for instance: 'home delivery', 'shop retrieval').

Price sub-resource

Provides all amounts (tax included, duty free, tax rate), used currency and percentage to apply for Price Alteration.

@baseType	A string. Generic attribute indicating the base class type of the extension class of the current object. Useful only when the class type of the current object is unknown to the implementation.
@schemaLocation	A string. Generic attribute containing the link to the schema that defines the structure of the class type of the current object.
@type	A string. Generic attribute containing the name of the resource class type.
dutyFreeAmount	A money (Money). All taxes excluded amount (expressed in the given currency).
percentage	A float. Percentage to apply for ProdOfferPriceAlteration.
taxIncludedAmount	A money (Money). All taxes included amount (expressed in the given currency).
taxRate	A float. Tax rate.

PriceAlteration sub-resource

Is an amount, usually of money, that modifies the price charged for an order item.

@baseType	A string. Generic attribute indicating the base class type of the extension class of the current object. Useful only when the class type of the current object is unknown to the implementation.
@schemaLocation	A string. Generic attribute containing the link to the schema that defines the structure of the class type of the current object.
@type	A string. Generic attribute containing the name of the resource class type.

Shopping Cart API REST Specification

applicationDuration	An integer. Duration during which the alteration applies on the order item price (for instance 2 months free of charge for the recurring charge).
description	A string. A narrative that explains in detail the semantics of this order item price alteration.
name	A string. A short descriptive name such as "Monthly discount".
priceType	A string. A category that describes the price such as recurring, one time and usage.
priority	An integer. Priority level for applying this alteration among all the defined alterations on the order item price.
recurringChargePeriod	A string. Could be month, week...
unitOfMeasure	A string. Could be minutes, GB...
price	A price (Price). Provides all amounts (tax included, duty free, tax rate), used currency and percentage to apply for Price Alteration.

Product sub-resource

Product reference. Configure the product characteristics (only configurable characteristics and necessary only if a non default value is selected) and/or identify the product that needs to be modified/deleted.

@baseType	A string. Generic attribute indicating the base class type of the extension class of the current object. Useful only when the class type of the current object is unknown to the implementation.
@referredType	A string. Indicates the type of the referred object. This attribute is to be used when the object is representing a reference to an existing object instead of the of the object itself.
@schemaLocation	A string. Generic attribute containing the link to the schema that defines the structure of the class type of the current object.
@type	A string. Generic attribute containing the name of the resource class type.
href	A string. Reference of the product.
id	A string. Unique identifier of the product.
name	A string. Name of the product.
productRelationship	A list of product relationships (ProductRelationship [*]). Linked products to the one instantiate, it can be:

Shopping Cart API REST Specification

	<ul style="list-style-type: none"> - "bundled" if the product is a bundle and you want to describe the "bundled" products inside this bundle - "reliesOn" if the product needs another already owned product to rely on (e.g. an option on an already owned mobile access product) <p>"targets" or "isTargeted" (depending on the way of expressing the link) for any other kind of links that may be useful.</p>
place	A list of places (Place [*]). Place defines a location generically. Could be a geographic address a geographic address, a geographic location.
characteristic	A list of product characteristics (ProductCharacteristic [*]). Characteristics of the product to instantiate or to modify.
relatedParty	A list of related party references (RelatedPartyRef [*]). A related party defines party or party role linked to a specific entity.
productSpecification	A product specification reference (ProductSpecificationRef). A ProductSpecification is a detailed description of a tangible or intangible object made available externally in the form of a ProductOffering to customers or other parties playing a party role.

ProductCharacteristic sub-resource

Characteristics of the product to instantiate or to modify.

name	A string. Name of the characteristic.
value	A string. Value of the characteristic.

ProductRelationship sub-resource

Linked products to the one instantiate, it can be:

- "bundled" if the product is a bundle and you want to describe the "bundled" products inside this bundle
- "reliesOn" if the product needs another already owned product to rely on (e.g. an option on an already owned mobile access product)

"targets" or "isTargeted" (depending on the way of expressing the link) for any other kind of links that may be useful.

type	A string. Type of the product relationship. It can be:
------	--------------------------------------------------------

- "bundled" if the product is a bundle and you want to describe the

Shopping Cart API REST Specification

"bundled" products inside this bundle

- "reliesOn" if the product needs another already owned product to rely on (e.g. an option on an already owned mobile access product)

"targets" or "isTargeted" (depending on the way of expressing the link) for any other kind of links that may be useful.

product A product (Product). Product reference. Configure the product characteristics (only configurable characteristics and necessary only if a non default value is selected) and/or identify the product that needs to be modified/deleted.

Quantity sub-resource

amount A float.

units A string.

CartItemRef relationship

CartItem reference. A CartItem is an identified part of the shopping cart.

@referredType A string. Generic attribute indicating the name of the class type of the referred resource entity.

id A string. Unique identifier of the cart item.

ProductOfferingRef relationship

ProductOffering reference. A product offering represents entities that are orderable from the provider of the catalog, this resource includes pricing information.

@referredType A string. Generic attribute indicating the name of the class type of the referred resource entity.

href A string. Reference of the product offering.

id A string. Unique identifier of the product offering.

name A string. Name of the product offering.

ProductSpecificationRef relationship

Product specification reference: A ProductSpecification is a detailed description of a tangible or intangible object made available externally in the form of a ProductOffering to customers or other parties playing a party role.

Shopping Cart API REST Specification

@referredType	A string. Generic attribute indicating the name of the class type of the referred resource entity.
href	A string. Reference of the product specification.
id	A string. Unique identifier of the product specification.
name	A string. Name of the product specification.
version	A string. Version of the product specification.
describing	A product specification schema reference (ProductSpecificationSchemaRef). ProductSpecification description could be done with schema use. This information allows to retrieve schema type and location.

RelatedPartyRef relationship

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

@referredType	A string. Generic attribute indicating the name of the class type of the referred resource entity.
href	A string. Reference of the related party, could be a party reference or a party role reference.
id	A string. Unique identifier of a related party.
name	A string. Name of the related party.
role	A string. Role of the related party.

Json representation sample

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

```
{
  "href": "https://host:port/shoppingCart/v2/shoppingCart/1203",
  "id": "1203",
  "@type": "ShoppingCart",
  "validFor": {
    "startDateTime": "2017-03-27T00:00",
    "endDateTime": "2017-10-24T00:00"
  },
  "contactMedium": [
    {
      "type": "email",
      "characteristic": {
        "emailAddress": "JackSmith@mail.com"
      }
    }
  ]
}
```

Shopping Cart API REST Specification

```
}
],
"cartTotalPrice": [
  {
    "description": "Total Recurring Price.",
    "name": "Monthly Price",
    "priceType": "recurring",
    "recurringChargePeriod": "monthly",
    "price": {
      "dutyFreeAmount": {
        "money": {
          "unit": "EUR",
          "value": 29
        }
      },
      "taxIncludedAmount": {
        "money": {
          "unit": "EUR",
          "value": 31.9
        }
      },
      "taxRate": 10
    }
  },
  {
    "applicationDuration": 3,
    "description": "First 3 month get 50% off",
    "name": "3 month for half",
    "priceCondition": "new customer",
    "priceType": "recurring",
    "recurringChargePeriod": "monthly",
    "validFor": {
      "startDateTime": "2017-03-26T00:00",
      "endDateTime": "2017-10-24T00:00"
    },
    "price": {
      "percentage": 50
    }
  }
]
},
"cartItem": [
  {
    "action": "add",
    "id": "8307",
    "quantity": 1,
    "status": "Active",
    "note": {
      "author": "Mr Smith",
      "date": "2017-03-28T00:00",
      "text": "Please wrap with double bag"
    }
  },

```

Shopping Cart API REST Specification

```
"product": {
  "name": "Talk Simple 25",
  "place": [
    {
      "href": "https://host:port/placeManagement/place/7396",
      "id": "7396",
      "name": "Main Store",
      "role": "default delivery"
    }
  ]
},
"productOffering": {
  "href": "https://host:port/productCatalog/productOffering/14277",
  "id": "14277",
  "name": "Talk Simple 25"
}
]
```

Notification Resource Models

2 notifications are defined for this API

Notifications related to ShoppingCart:

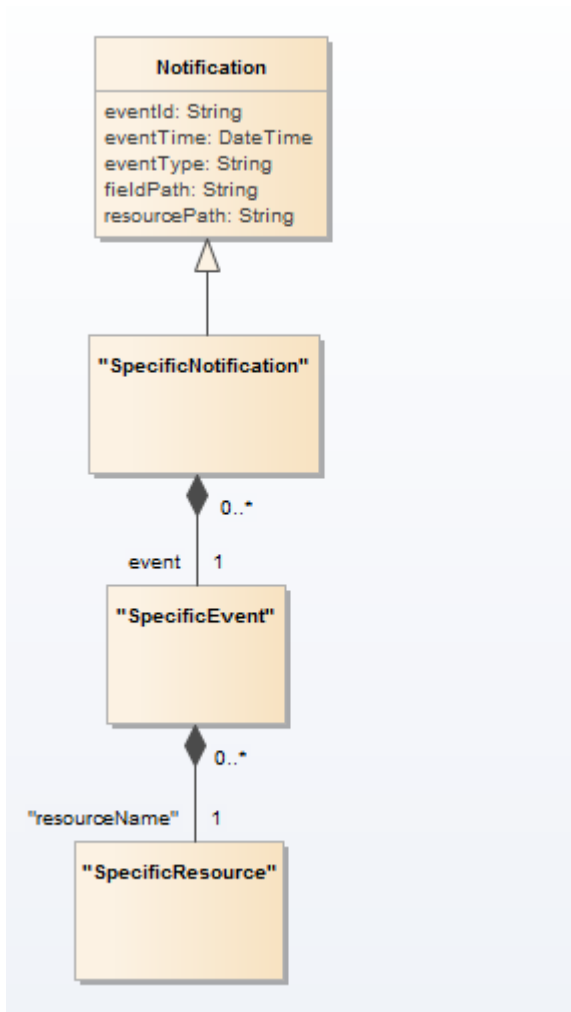
- ShoppingCartCreationNotification
- ShoppingCartChangeNotification

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

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

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

Shopping Cart API REST Specification



SHOPPING CART CREATION NOTIFICATION

Notification sent when a new ShoppingCart resource is created.

Json representation sample

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

```

{
  "eventId": "00001",
  "eventTime": "2015-11-16T16:42:25-04:00",
  "eventType": "ShoppingCartCreationNotification",
  "event": {
    "shoppingCart": {
      "-- SEE ShoppingCart RESOURCE SAMPLE --"
    }
  }
}

```

SHOPPING CART CHANGE NOTIFICATION

Notification sent when changing a ShoppingCart resource.

Json representation sample

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

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

API OPERATIONS

Remember the following Uniform Contract:

Operation on Entities	Uniform API Operation	Description
Query Entities	GET Resource	GET must be used to retrieve a representation of a resource.
Create Entity	POST Resource	POST must be used to create a new resource
Partial Update of an Entity	PATCH Resource	PATCH must be used to partially update a resource
Complete Update of an Entity	PUT Resource	PUT must be used to completely update a resource identified by its resource URI
Remove an Entity	DELETE Resource	DELETE must be used to remove a resource
Execute an Action on an Entity	POST on TASK Resource	POST must be used to execute Task Resources
Other Request Methods	POST on TASK Resource	GET and POST must not be used to tunnel other request methods.

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

Notifications are also described in a subsequent section.

OPERATIONS ON SHOPPING CART

LIST SHOPPING CARTS

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

Description

This operation list shopping cart 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 ShoppingCart resources.

Request
GET {apiRoot}/shoppingCart Accept: application/json
Response
200 [{ "href": "https://host:port/shoppingCart/v2/shoppingCart/1203", "id": "1203", "@type": "ShoppingCart", "validFor": { "startDateTime": "2017-03-27T00:00", "endDateTime": "2017-10-24T00:00" }, "contactMedium": [{ "type": "email", "characteristic": { "emailAddress": "JackSmith@mail.com" } }], "cartTotalPrice": [{

Shopping Cart API REST Specification

```

"description": "Total Recurring Price.",
"name": "Monthly Price",
"priceType": "recurring",
"recurringChargePeriod": "monthly",
"price": {
  "dutyFreeAmount": {
    "money": {
      "unit": "EUR",
      "value": 29
    }
  },
  "taxIncludedAmount": {
    "money": {
      "unit": "EUR",
      "value": 31.9
    },
    "taxRate": 10
  }
},
"priceAlteration": [
  {
    "applicationDuration": 3,
    "description": "First 3 month get 50% off",
    "name": "3 month for half",
    "priceCondition": "new customer",
    "priceType": "recurring",
    "recurringChargePeriod": "monthly",
    "validFor": {
      "startDateTime": "2017-03-26T00:00",
      "endDateTime": "2017-10-24T00:00"
    },
    "price": {
      "percentage": 50
    }
  }
]
],
"cartItem": [
  {
    "action": "add",
    "id": "8307",
    "quantity": 1,
    "status": "Active",
    "note": {
      "author": "Mr Smith",
      "date": "2017-03-28T00:00",
      "text": "Please wrap with double bag"
    },
    "product": {
      "name": "Talk Simple 25",
      "place": [
        {

```

```

    "href": "https://host:port/placeManagement/place/7396",
    "id": "7396",
    "name": "Main Store",
    "role": "default delivery"
  }
]
},
"productOffering": {
  "href": "https://host:port/productCatalog/productOffering/14277",
  "id": "14277",
  "name": "Talk Simple 25"
}
}
]
}
]

```

RETRIEVE SHOPPING CART

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

Description

This operation retrieves a shopping cart entity.

Attribute selection is enabled for all first level attributes.

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

Usage Samples

Here's an example of a request for retrieving a ShoppingCart resource.

Request
GET {apiRoot}/shoppingCart/1203 Accept: application/json
Response
200 { "href": "https://host:port/shoppingCart/v2/shoppingCart/1203",

Shopping Cart API REST Specification

```
"id": "1203",
"@type": "ShoppingCart",
"validFor": {
  "startDateTime": "2017-03-27T00:00",
  "endDateTime": "2017-10-24T00:00"
},
"contactMedium": [
  {
    "type": "email",
    "characteristic": {
      "emailAddress": "JackSmith@mail.com"
    }
  }
],
"cartTotalPrice": [
  {
    "description": "Total Recurring Price.",
    "name": "Monthly Price",
    "priceType": "recurring",
    "recurringChargePeriod": "monthly",
    "price": {
      "dutyFreeAmount": {
        "money": {
          "unit": "EUR",
          "value": 29
        }
      },
      "taxIncludedAmount": {
        "money": {
          "unit": "EUR",
          "value": 31.9
        },
        "taxRate": 10
      }
    }
  },
  {
    "priceAlteration": [
      {
        "applicationDuration": 3,
        "description": "First 3 month get 50% off",
        "name": "3 month for half",
        "priceCondition": "new customer",
        "priceType": "recurring",
        "recurringChargePeriod": "monthly",
        "validFor": {
          "startDateTime": "2017-03-26T00:00",
          "endDateTime": "2017-10-24T00:00"
        },
        "price": {
          "percentage": 50
        }
      }
    ]
  }
],
```

```

"cartItem": [
  {
    "action": "add",
    "id": "8307",
    "quantity": 1,
    "status": "Active",
    "note": {
      "author": "Mr Smith",
      "date": "2017-03-28T00:00",
      "text": "Please wrap with double bag"
    },
    "product": {
      "name": "Talk Simple 25",
      "place": [
        {
          "href": "https://host:port/placeManagement/place/7396",
          "id": "7396",
          "name": "Main Store",
          "role": "default delivery"
        }
      ]
    },
    "productOffering": {
      "href": "https://host:port/productCatalog/productOffering/14277",
      "id": "14277",
      "name": "Talk Simple 25"
    }
  }
]
}

```

CREATE SHOPPING CART

POST /shoppingCart

Description

This operation creates a shopping cart entity.

Mandatory and Non Mandatory Attributes

The following tables provides the list of mandatory and non mandatory attributes when creating a ShoppingCart, 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
cartItem	

Shopping Cart API REST Specification

Non Mandatory Attributes	Default Value	Rule
@baseType		
@schemaLocation		
@type		
validFor		
contactMedium		
cartTotalPrice		
relatedParty		

Additional Rules

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

Context	Mandatory Sub-Attributes
cartItem	id, action
cartItem.cartItem	id, action
cartItem.note	text
relatedParty	id OR href OR name
cartItem.product.place	role, href
cartItem.productOffering	id OR href
cartItem.product	id OR href

Default Values Summary

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

Attributes	Default Value
cartItem.status	Active

Usage Samples

Here's an example of a request for creating a ShoppingCart resource. In this example the request only passes mandatory attributes.

Request
POST {apiRoot}/shoppingCart Content-Type: application/json { "cartItem": [{

Shopping Cart API REST Specification

```

"action": "add",
"id": "8307",
"quantity": 1,
"status": "Active",
"note": {
  "author": "Mr Smith",
  "date": "2017-03-28T00:00",
  "text": "Please wrap with double bag"
},
"product": {
  "name": "Talk Simple 25",
  "place": [
    {
      "href": "https://host:port/placeManagement/place/7396",
      "id": "7396",
      "name": "Main Store",
      "role": "default delivery"
    }
  ]
},
"productOffering": {
  "href": "https://host:port/productCatalog/productOffering/14277",
  "id": "14277",
  "name": "Talk Simple 25"
}
}
]
}

```

Response

201

```

{
  "href": "https://host:port/shoppingCart/v2/shoppingCart/1203",
  "id": "1203",
  "@type": "ShoppingCart",
  "validFor": {
    "startDateTime": "2017-03-27T00:00",
    "endDateTime": "2017-10-24T00:00"
  },
  "cartTotalPrice": [
    {
      "description": "Total Recurring Price.",
      "name": "Monthly Price",
      "priceType": "recurring",
      "recurringChargePeriod": "monthly",
      "price": {
        "dutyFreeAmount": {
          "money": {
            "unit": "EUR",

```

Shopping Cart API REST Specification

```
    "value": 29
  }
},
"taxIncludedAmount": {
  "money": {
    "unit": "EUR",
    "value": 31.9
  },
  "taxRate": 10
}
},
"priceAlteration": [
  {
    "applicationDuration": 3,
    "description": "First 3 month get 50% off",
    "name": "3 month for half",
    "priceCondition": "new customer",
    "priceType": "recurring",
    "recurringChargePeriod": "monthly",
    "validFor": {
      "startDateTime": "2017-03-26T00:00",
      "endDateTime": "2017-10-24T00:00"
    },
    "price": {
      "percentage": 50
    }
  }
]
},
"cartItem": [
  {
    "action": "add",
    "id": "8307",
    "quantity": 1,
    "status": "Active",
    "note": {
      "author": "Mr Smith",
      "date": "2017-03-28T00:00",
      "text": "Please wrap with double bag"
    },
    "product": {
      "name": "Talk Simple 25",
      "place": [
        {
          "href": "https://host:port/placeManagement/place/7396",
          "id": "7396",
          "name": "Main Store",
          "role": "default delivery"
        }
      ]
    }
  }
],
"productOffering": {
```

```

    "href": "https://host:port/productCatalog/productOffering/14277",
    "id": "14277",
    "name": "Talk Simple 25"
  }
}
]
}

```

PATCH SHOPPING CART

PATCH /shoppingCart/{id}

Description

This operation allows partial updates of a shopping cart 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
@baseType	
@schemaLocation	
@type	
contactMedium	
cartItem	
relatedParty	

Non Patchable Attributes	Rule
id	
href	
validFor	
cartTotalPrice	

Usage Samples

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

Request

PATCH {apiRoot}/shoppingCart/7616
Content-Type: application/merge-patch+json

```
{
  "contactMedium": [
    {
      "type": "email",
      "characteristic": {
        "emailAddress": "peter_henry@newmail.com"
      }
    }
  ]
}
```

Response

201

```
{
  "href": "https://host:port/shoppingCart/v1/shoppingCart/1203",
  "id": "1203",
  "validFor": {
    "startDateTime": "2017-03-27T00:00",
    "endDateTime": "2017-10-24T00:00"
  },
  "contactMedium": [
    {
      "type": "email",
      "characteristic": {
        "emailAddress": "peter_henry@newmail.com "
      }
    }
  ],
  "cartTotalPrice": [
    {
      "description": "Total Recurring Price.",
      "name": "Monthly Price",
      "priceType": "recurring",
      "recurringChargePeriod": "monthly",
      "price": {
        "dutyFreeAmount": {
          "money": {
            "unit": "EUR",
            "value": 29
          }
        }
      },
      "taxIncludedAmount": {
        "money": {
```

Shopping Cart API REST Specification

```
    "unit": "EUR",
    "value": 31.9
  },
  "taxRate": 10
}
},
"priceAlteration": [
{
  "applicationDuration": 3,
  "description": "First 3 month get 50% off",
  "name": "3 month for half",
  "priceCondition": "new customer",
  "priceType": "recurring",
  "recurringChargePeriod": "monthly",
  "validFor": {
    "startDateTime": "2017-03-26T00:00",
    "endDateTime": "2017-10-24T00:00"
  },
  "price": {
    "percentage": 50
  }
}
]
},
"cartItem": [
{
  "action": "add",
  "id": "8307",
  "quantity": 1,
  "status": "Active",
  "note": {
    "author": "Mr Smith",
    "date": "2017-03-28T00:00",
    "text": "Please wrap with double bag"
  },
  "product": {
    "name": "Talk Simple 25",
    "place": [
      {
        "href": "https://host:port/placeManagement/place/7396",
        "id": "7396",
        "name": "Main Store",
        "role": "default delivery"
      }
    ]
  },
  "productOffering": {
    "href": "https://host:port/productCatalog/productOffering/14277",
    "id": "14277",
    "name": "Talk Simple 25"
  }
}
]
```

```
]
}
```

DELETE SHOPPING CART

DELETE /shoppingCart/{id}

Note: this operation is available only to ADMIN API users

Description

This operation deletes a shopping cart entity.

Usage Samples

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

Request
DELETE {apiRoot}/shoppingCart/42
Response
204

API NOTIFICATIONS

For every single of operation on the entities use the following templates and provide sample REST notification POST calls.

It is assumed that the Pub/Sub uses the Register and UnRegister mechanisms described in the REST Guidelines reproduced below.

REGISTER LISTENER

POST /hub

Description

Sets the communication endpoint address the service instance must use to deliver information about its health state, execution state, failures and metrics. Subsequent POST calls will be rejected by the service if it does not support multiple listeners. In this case DELETE /api/hub/{id} must be called before an endpoint can be created again.

Behavior

Returns HTTP/1.1 status code 204 if the request was successful.

Returns HTTP/1.1 status code 409 if request is not successful.

Usage Samples

Here's an example of a request for registering a listener.

Request
<pre>POST /api/hub Accept: application/json {"callback": "http://in.listener.com"}</pre>
Response
<pre>201 Content-Type: application/json Location: /api/hub/42 {"id": "42", "callback": "http://in.listener.com", "query": null}</pre>

UNREGISTER LISTENER

DELETE /hub/{id}

Description

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

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

Shopping Cart API REST Specification

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**

Version Number	Date	Release led by:	Description
Version 1.0	04/15/2017	Pierre Gauthier TM Forum pgauthier@tmforum.org	First Release of the Document.
Version 2.0	11-Jun-2018	Pierre Gauthier	Alignment with Guidelines 3.0
Version 2.0.1	28-Jun-2018	Adrienne Walcott	Formatting/style edits prior to R18 publishing.

RELEASE HISTORY

Release Number	Date	Release led by:	Description
Release 18.0.0	25-Jun-2018	Pierre Gauthier TM Forum pgauthier@tmforum.org	Initial Release

CONTRIBUTORS TO DOCUMENT

<ul style="list-style-type: none">• Jacob Avraham, Amdocs jacoba@amdocs.com• ROBERT Ludovic, Orange ludovic.robert@orange.com• Mariano Belaunde, Orange mariano.belaunde@orange.com• MaXu, Huawei maxu@huawei.com	Initial version
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------