

TM Forum Specification

Promotion API REST Specification

TMF671

Release 17.0.1

November 2017

Latest Update: TM Forum Release 17	TM Forum Approved
Version 2.0.1	IPR Mode: RAND

NOTICE

Copyright © TM Forum 2017. 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.

TM FORUM invites any TM FORUM Member or any other party that believes it has patent claims that would necessarily be infringed by implementations of this TM Forum Standards Final Deliverable, to notify the TM FORUM Team Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this deliverable.

The TM FORUM invites any party to contact the TM FORUM Team Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this TM FORUM Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the TM FORUM Collaboration Project Team that produced this TM FORUM Standards Final Deliverable. TM FORUM may include such claims on its website, but disclaims any obligation to do so.

TM FORUM takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this TM FORUM Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on TM FORUM's procedures with respect to rights in any document or deliverable produced by a TM FORUM Collaboration Project Team can be found on the TM FORUM website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this TM FORUM Standards Final Deliverable, can be obtained from the TM FORUM Team Administrator. TM FORUM makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

Direct inquiries to the TM Forum office:

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	4
List of Tables	5
Introduction	6
SAMPLE USE CASES.....	11
RESOURCE MODEL.....	13
Managed Entity and Task Resource Models.....	13
Promotion Resource	13
Notification Resource Models.....	20
API OPERATION.....	22
List Promotions.....	22
Retrieve Promotion.....	23
Create Promotion	24
Patch Promotion	28
Delete Promotion.....	29
API NOTIFICATION.....	31
REGISTER LISTENER.....	31
UNREGISTER LISTENER	31
Publish Event to Listener	32
AcknowledgmEnts	33
Release History	33
Contributors to Document.....	33

LIST OF TABLES

N/A

INTRODUCTION

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

It provides a standardized mechanism for Promotion management such as creation, update, retrieval, deletion and notification of events.

Promotion API manages the following data resources:

- **Promotion** (also called product offering promotion, according to the name in Business Process Framework)
 - o Promotion is widely used in e-commerce domain. It is used to provide the additional discount, voucher, bonus or gift to the customer who meets the pre-defined criteria. Using promotion, the enterprise is able to attract the users and encourage more consumption, especially continuous purchases. Normally Promotion is not regarded as one type of product or product offering. It is often applied when the customer buys the product offerings with the price or amount surpassing the certain limit. The typical example is the “buy 1 and get 1 free” or “buy \$100 and get \$10 bonus”.



The example of promotion in Amazon can be browsed online. It shows the typical promotion in the e-commerce world.

Promotion Type	Qualifying Action	Benefit Options	Example
Money Off (Purchase Discount)	Customer reaches a minimum threshold, either by spending a dollar amount you've specified, or by buying a certain quantity of a product or products.	<ul style="list-style-type: none"> • Buy \$X or more, and get \$Y off purchase of qualifying products • Buy \$X or more, and get Y% off qualifying products • Buy n items or more, and get \$Y off purchase of qualifying products • Buy n items or more, and get Y% off qualifying products • Buy every n items, and get \$Y off purchase of qualifying products • Buy every n items, and get Y% off qualifying products • Buy every n items at a fixed reduced price 	<ul style="list-style-type: none"> • Spend \$25 or more and receive \$5 off • Save 30% when you purchase one or more qualifying items • For every 5 pairs of socks purchased, save 20%

Buy One Get One (Free Product)		<ul style="list-style-type: none"> • Buy \$X or more, and get n items for free on qualifying products • Buy N items or more, and get n items for free on qualifying products • Buy every N items, and get n items for free on qualifying products 	<ul style="list-style-type: none"> • Buy 2 lipsticks, and get 1 free • Buy 1 muffin pan, get 1 box of muffin cups free • For every 5 pencils purchased, get a pencil sharpener for free
External Benefit		<ul style="list-style-type: none"> • Buy \$X or more, and get a post-order benefit • Buy n items or more, and get a post-order benefit 	Receive free instructional video with purchase

Note: Promotion API in this document is about the configuration and maintenance of this business object, not the execution of such promotion. This API follows the example of Product Catalog Management API which focuses on the “configuration phase” instead of the “use phase”. The execution of promotion is the responsibility of e-commerce platform which has to apply the discount (or gift, voucher, etc) based on the information retrieved from Promotion API.

Promotion API performs the following operation on the resource of Promotion

- Retrieval of an existing promotion depending on filter criteria
- Partial update of an existing promotion
- Creation (configuration) of a new promotion
- Deletion of an existing promotion
- Notification of events:
 - o promotion creation
 - o promotion updating
 - o promotion deletion

Mapping with SID ABE

Promotion is mapped to “*Product Offering ABE::Product Promotion ABE*” in TMF Information Framework (SID).

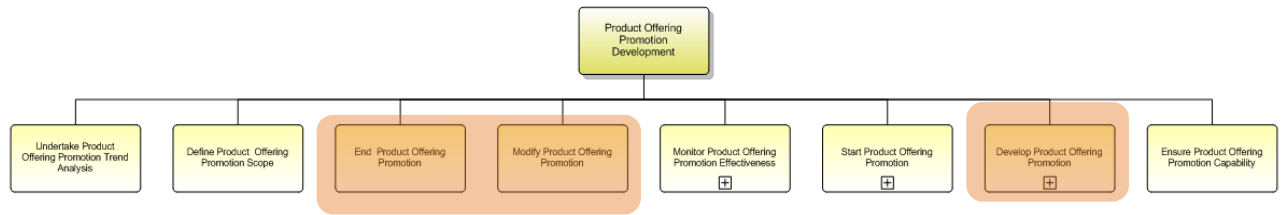
Mapping with Business Process Framework (eTOM)

In Business Process Framework there is the description for promotion:

- Level 2 Process: 1.2.7 Product Specification & Offering Development & Retirement
- Level 3 Process: 1.2.7.2 Product Offering Development & Retirement
- Level 4 Process : 1.2.7.2.6 Product Offering Promotion Development

Brief Description of “Product Offering Promotion Development” is to develop **a promotion** for one or more product offerings delivered using different mediums (television, radio, print, online) that advertise the promotion to reach the market, customers and channels.

The level 4 process of “Product Offering Promotion Development” is illustrated as a group of Level 5 processes.



In these processes, there are several of them requiring the API to support the design and development of the promotion, including:

- ✓ End Product Offering Promotion
- ✓ Modify Product Offering Promotion
- ✓ Develop Product Offering Promotion

Difference between Promotion API and other existing TMF APIs

Here the differences between promotion API and other existing published TMF APIs are explained to clarify why this separate API is not covered simply with those APIs.

✓ **Difference with Product Catalog API**

Promotion and Product Offering in the Product Catalog has some similarities. The product offering and promotion has the relationship of mutual complement.

Since Product Catalog API focuses on the configuration of product offering, it is necessary to distinguish the relationship between Product Offering and Promotion. The key comparison points are listed as the table below, especially the yellow background color.

Feature	<i>Product Offering</i>	<i>Promotion</i>
Definition	Formal sellable entity on the shelf for sales.	Indirectly acquired benefits to promote the sales of offering.
Role	Product Manager	Marketing or sales team
Target customer	Most customer, sometimes for specific customers	For specific customer, and only available for all customer in short term
Time Span	Long Term	Certain period
Relation with product	Depending on product	No direct relation with product
Tariff and Charge	Often including tariff plan	None
Discount & rebate	Sometimes included	Included
Awarding	None	Awarding the customer with gift, code, or loyalty points

Criteria	Conditions consist of place, channel, term	More flexible condition including the combination of factors (customer attribute, offering attribute, resource elements, channel)
Acquisition	By purchasing	Only available by awarding for free, or by joining the activity. Promotion cannot be sold directly.
Relationship	Multiple type: dependency, mutual exclusive, replacement, inheritance	Mutual Exclusive and Dependency
Terms & Condition	Necessary	Contract restraint is optional
Penalty	Sometimes with penalty	None
Redemption	None	Redeem code or coupon

From the comparison, it is self-evident the promotion is another type of data entity which is not same as Product Offering.

✓ **Difference with Service Qualification**

This API is used to provide the validation on eligibility and compatibility business rules during the purchase phase of the offering.

On the other side, **Promotion API** is used after the product offering is validated and chosen for purchase. The confirmed and selected product offering is the prerequisite of filtering the promotion.

✓ **Difference with Product Ordering**

This API focuses on the order (purchase) action and the implementation after the chosen offering and promotion has been submitted.

As for Promotion API, it only acts before the product order is created.

✓ **Difference with Shopping Cart API**

The Shopping Cart API is a container to load the selected offerings for the customer to purchase. It does not replace the offering or promotion.

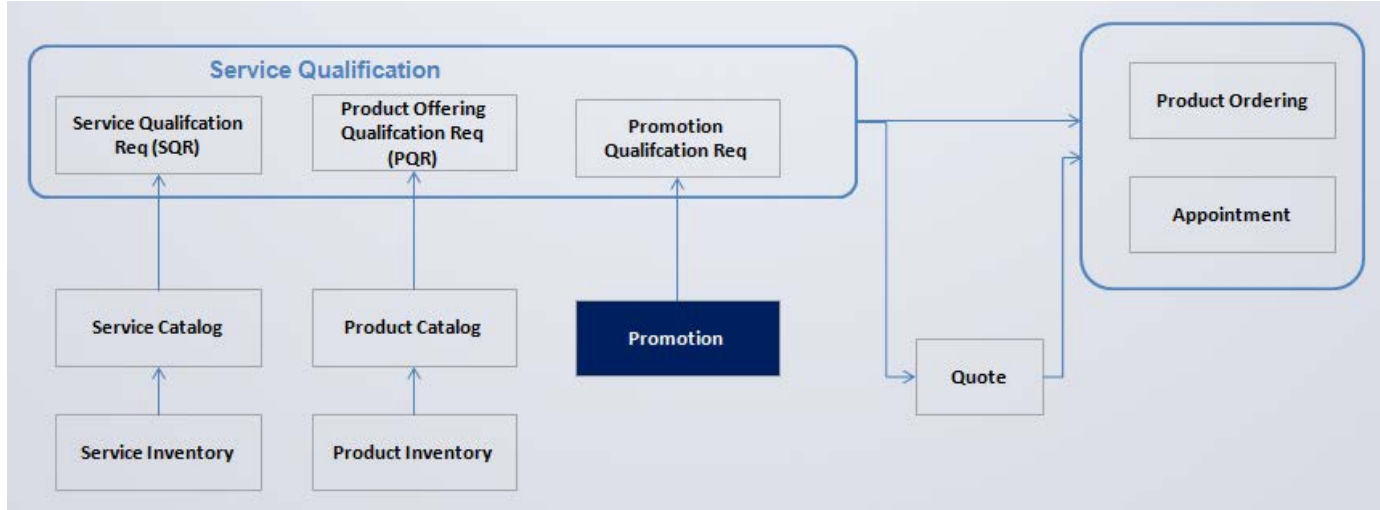
Promotion API only depends on the “selected offering”, not the shopping cart itself. There could be two scenarios:

- 1) The offering in the shopping cart will be used as the input condition of the Promotion API. Based on the selected offering in the cart, the relevant promotion can be queried and shown to the user (customer or agent) on e-commerce portal. If the user changes the offering in the shopping cart, promotion will also be impacted.
- 2) The user is also allowed to purchase the offering without adding it into the cart. The promotion will still be effective based on the chosen offering information.

Promotion API in the Transaction Flow

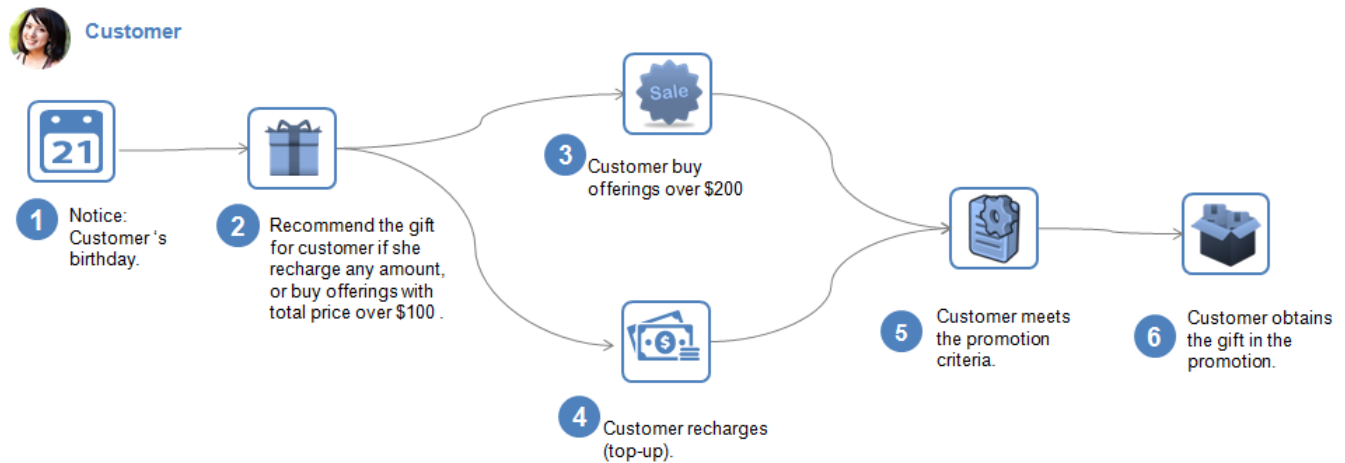
Based on the flow in the *TMF645_Service_Qualification_API_REST_Specification_R16.0.1*, the various APIs are organized in the sequence to show their relationship with the customer journey.

The position of Promotion API can be found in this diagram. It is invoked after the product offering has been selected and before the Product Ordering is submitted (created).



SAMPLE USE CASES

Examples of use cases using Promotion API is as follows.



Use Case Id	UC_TMF_Promotion_0001
Use Case Name	Customer consumes and acquires promotion.
Summary	This case describes the customer purchases the offering, meets the criteria of the promotion, and obtains the promotion.
Actor(s)	Customer (person)
Pre-Conditions	Promotion API has been used to configure the promotion.
Begins When	When the marketing team or marketing system starts disseminating the information (advertisement) for the promotion.
Description	<ol style="list-style-type: none"> 1) The customer gets the notice. It is the customer's birthday so there is promotion. 2) The system or agent recommends the gift for customer if she recharge any amount, or buy offerings with total price over \$100. <i>Note: Promotion API is used in this step.</i> 3) Customer buy offerings over \$200 4) (Or) Customer recharges (top-up). 5) Customer meets the promotion criteria. 6) Customer obtains the gift which is defined in the promotion.

<p>Ends When</p>	<p><i>In case of submitting successfully:</i></p> <p>The promotion is handled by the e-commerce platform and the customer gains the promotion.</p> <p><i>In case of failure:</i></p> <p>The promotion is not given to the customer even if he/she should have one. Such exception needs manual interference to redeem (save) the situation.</p>
<p>Post-Conditions</p>	
<p>Exceptions</p>	
<p>Traceability</p>	

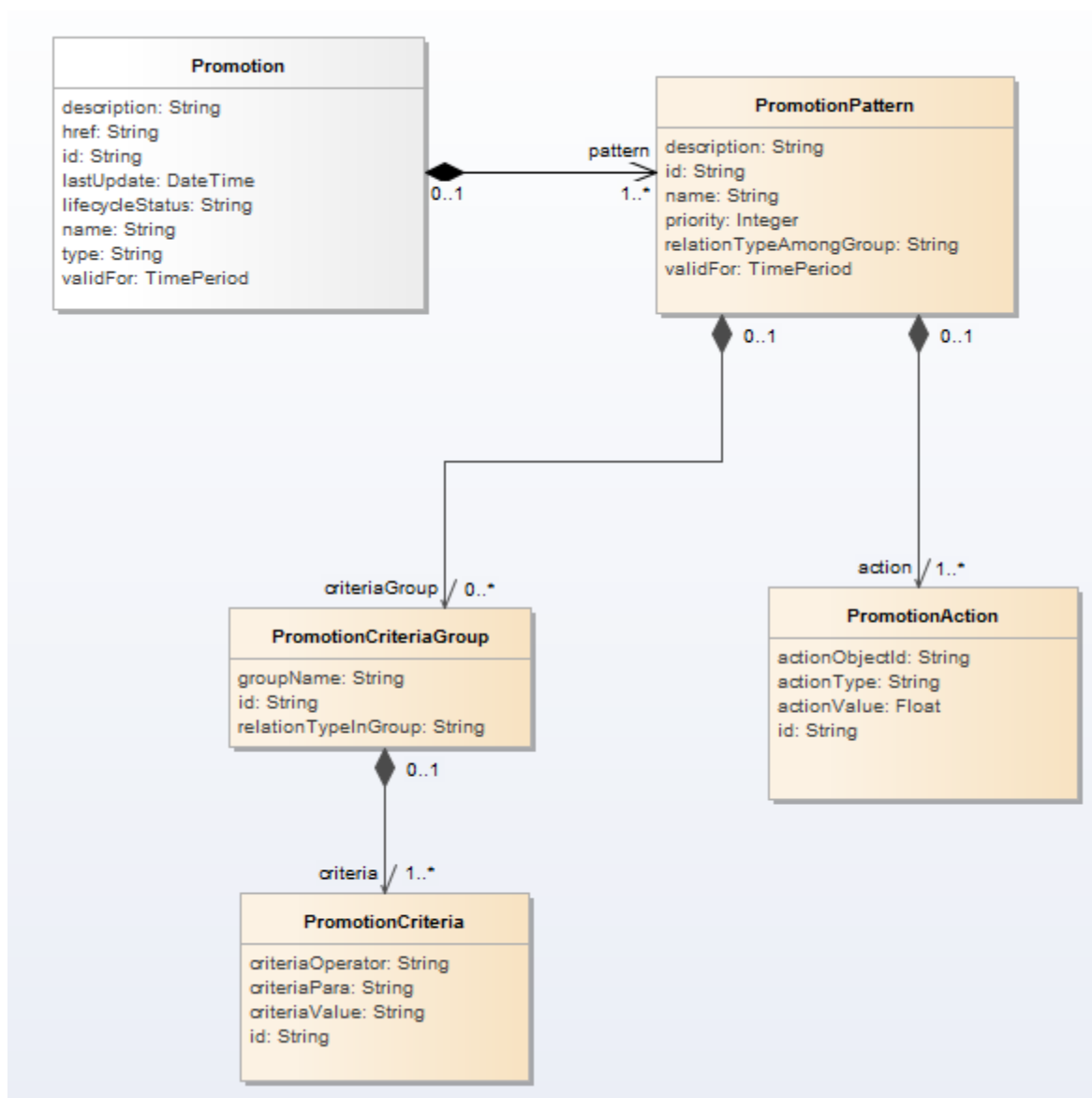
RESOURCE MODEL

Managed Entity and Task Resource Models

PROMOTION RESOURCE

Promotion Resource is used to provide the additional discount, voucher, bonus or gift to the customer who meets the pre-defined criteria. Using promotion, the enterprise is able to attract the users and encourage more consumption, especially continuous purchases. Normally Promotion is not regarded as one type of product or product offering. It is often applied when the customer buys the product offerings with the price or amount surpassing the certain limit.

PROMOTION RESOURCE MODEL



FIELD DESCRIPTIONS

Promotion Resource

Field	Type	Description
description	String	Description of Promotion
lastUpdate	DateTime	Latest update date of Promotion
lifecycleStatus	String	Status of Promotion, incl. <ul style="list-style-type: none"> ✓ Draft ✓ Test ✓ WaitForApproval ✓ Release ✓ Suspend ✓ Retirement
name	String	Name of Promotion
type	String	This field is optional. Type of promotion. The basic type is : 1: Award. It means the gift or telecom free unit is given to the customer for free. 2: Discount. It means the promotion will cause the discount on the price of the offering. 3: Reduction. It means the promotion will cause the reduction on the price of the offering. More types can be extended in future.
validFor	TimePeriod	The period in which the Promotion is valid.
id	String	Unique identifier of Promotion
href	String	Hypertext Reference of the Promotion.
promotionPattern	Array	A list of promotion patterns.

PromotionPattern sub-resource

Detailed pattern of the promotion.

The pattern decides the conditions of promotion and the benefit of the promotion to be given to the eligible customer.

Field	Type	Description
id	String	Unique identifier.
name	String	Name
priority	integer	This field is optional. Priority. Smaller number means "high".
description	String	This field is optional. Description
relationTypeAmongGroup	String	This field is optional. The logical relation type amongst the "promotion criteria group". "AND": AND logical relation "OR": OR logical relation
promotionAction	Array	A list of promotion actions
promotionCriteriaGroup	Array	A list of promotion criteria groups

PromotionCriteriaGroup sub-resource

The group of the criteria of the promotion.

The logical relationship between different groups is decided by the "relationTypeInGroup".

Field	Type	Description
id	String	Unique identifier.

Field	Type	Description
groupName	String	Name of the group
relationTypeInGroup	String	The logical relation type amongst the various criteria inside a "promotion criteria group". "AND": AND logical relation "OR": OR logical relation
promotionCriteria	Array	A list of promotion criteria.

PromotionCriteria sub-resource

Criteria of the promotion.

Field	Type	Description
id	String	Unique identifier.
criteriaPara	String	<p>The parameter (factor) of the criteria.</p> <p>The basic factors are abstracted from these data sources:</p> <ul style="list-style-type: none"> ✓ customer information ✓ subscribed offering ✓ subscribed product ✓ category of the subscriber offering <p>The detail parameters are:</p> <ul style="list-style-type: none"> - 1.1: Customer id - 1.2: Customer name - 1.3 Customer birthday - 1.4 customer level - 2.1 subscribed offering ID - 2.2 monthly fee amount of subscriber offering - 2.3 business fee amount of subscriber offering - 2.4 total monthly fee amount of all the subscriber offering in the single order - 2.5 total business fee amount of all the subscriber offering in the single order - 3.1 subscribed product ID - 4.1 category ID of the subscribed offering - 5.1 recharge amount per day - 5.2 recharge amount per week - 5.3 recharge amount per month

Field	Type	Description
		<ul style="list-style-type: none"> - 6.1 bill amount - 7.1 channel type (e.g. retail shop, mobile app) <p>More factors can be extended in future.</p>
criteriaOperator	String	<p>The logic expression including parameter and operator.</p> <p>“=”</p> <p>“>”</p> <p>“<”</p> <p>“>=”</p> <p>“<=”</p> <p>“<>”</p>
criteriaValue	String	<p>The value is filled for the comparison of the criteria.</p> <p>For example, when the criteriaPara is “1.2 customer birthday”, the value can be “1997/12/30”.</p>

PromotionAction sub-resource

Action of the promotion.

When the customer meets the conditions in the “promotion pattern”, the customer can be given the benefits in the “action”.

Field	Type	Description
id	String	Unique identifier.
actionType	Integer	<p>When the “Promotion -> type” is “1: Award”, the action type can be one of the following:</p> <ul style="list-style-type: none"> 1.1. voice (minute) 1.2 data (MB) 1.3 data (GB) 1.4 SMS 2.1 bonus point 3.1 physical gift (e.g. a cup, a USB disk) 3.2 voucher

Field	Type	Description
		<p>4.1 promotion code (it could be used by the customer to redeem other gift)</p> <p>5.1 an existing offering</p> <p>6.1 currency (i.e. money, balance on the account)</p> <p>The customer who meets the criteria can acquire the awards.</p> <p>Nota bene: More types of action can be extended in future.</p>
actionValue	Float	<p>When the "Promotion -> type" is "1: Award", it means the amount or value of the awards decided by "actionType", such as:</p> <ul style="list-style-type: none"> ✓ Amount of gift ✓ Amount of bonus (money or point) ✓ Value of discount ▪ When the "Promotion -> type" is "2: Discount", it means the value of the discount ▪ When the "Promotion -> type" is "3: reduction", it means the value of the reduced money (e.g. USD50, 20 Euro).
actionObjectId	Integer	<p>It refers to the object (entity) ID which is impacted by the promotion action.</p> <ul style="list-style-type: none"> ▪ When the "Promotion -> type" is "1: Award", and the actionType is "1.1, 1.2, 1.3, 1.4", the "actionObjectId" is the sub-account ID. ▪ When the "Promotion -> type" is "2: Discount", it is the offering ID <p>When the "Promotion -> type" is "3: reduction", it is the offering ID.</p>

PROMOTION SAMPLE

```
{
  "id": "1001",
  "href": "http://serverlocation:port/promotion/v1/promotion/1001",
  "name": "Promotion of birthday",
  "description": "VIP Customer can get gift on the birthday if the customer
```

```

recharges over $50 or the same customer purchase $100 monthly fee in one order",
  "lastUpdate": "2016-05-21 T04:00:00.0Z",
  "validFor":
  {
    "startDateTime": "2017-12-19 T04:00:00.0Z",
    "endDateTime": "2017-12-31 T20:42:23.0Z"
  },
  "lifecycleStatus": "Test",

  "promotionPattern ":
  {
    "id": "1109",
    "name": "Gift for recharge or purchase on birthday",
    "description": "Gift is effective when the customer is on birthday,
and buys offerings with total price over $100 or recharge over $50",
    "priority": "0",
    "relationTypeAmongGroup": "AND",
    "validFor":
    {
      "startDateTime": "2016-06-19 T04:00:00.0Z",
      "endDateTime": "2016-09-19 T20:42:23.0Z"
    },
    "promotionCriteriaGroup":
    [
      {
        "id ": "Group.A ",
        "groupName ": "birthday as condition ",
        "relationTypeInGroup ": "AND"
        "promotionCriteria":
        [{
          "id": "Group.A.1",
          "criteriaPara": "1.3",
          "criteriaOperator": "=",
          "criteriaValue": "#currentDate"
        }
        ]
      },
      {
        "id": "Group.B",
        "groupName": "recharge amount or payment amount as condition",
        "relationTypeInGroup": "OR"
        "promotionCriteria": [
          {
            "id": "Group.B.1",
            "criteriaPara": "5.1",
            "criteriaOperator": ">=",
            "criteriaValue": "50"
          },
          {
            "id": "Group.B.2",
            "criteriaPara": "2.4",
            "criteriaOperator": ">=",
            "criteriaValue": "100"
          }
        ]
      }
    ]
  },
  "promotionAction":

```

```
[{
  "actionValue": "1"
  "actionObjectId": "2001",
  "actionType": "3.1",
  "id": "2209",
}]
}
```

Notification Resource Models

PROMOTION CREATION NOTIFICATION

Notification sent when a new Promotion resource is created.

Json representation sample

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

```
{
  "eventId": "8976",
  "eventTime": "2014-09-27T05:46:25.0Z",
  "eventType": "PromotionCreationNotification",
  "event":
    {
      "promotion": {-- SEE Promotion RESOURCE SAMPLE --}
    }
}
```

PROMOTION CHANGE NOTIFICATION

Notification sent when changing a Promotion resource.

Json representation sample

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

```
{  
  "eventId": "8976",  
  "eventTime": "2014-09-27T05:46:25.0Z",  
  "eventType": "PromotionChangeNotification",  
  "event":  
    {  
      "promotion": {"-- SEE Promotion RESOURCE SAMPLE --"}  
    }  
}
```

API OPERATION

For every single of operation on the entities use the following templates and provide sample REST requests and responses.

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.

LIST PROMOTIONS

HTTP Verb + URI:

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

Description

This operation is used to list promotion 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 Promotion resources.

REQUEST
GET /promotion/v1/promotion Accept: application/json
RESPONSE
200 Content-Type: application/json <pre>[{ <i>a whole representation of the promotion resource with all its attributes.</i> <i>Refer to promotion Resource.</i> }]</pre>

RETRIEVE PROMOTION

HTTP Verb + URI:

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

Description

This operation retrieves a promotion 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 promotion resource.

Request
GET /promotion/v1/promotion/11006 Accept: application/json
Response
200 Content-Type: application/json <pre>{ <i>a whole representation of the promotion resource with all its attributes.</i> <i>Refer to promotion Resource.</i> }</pre>

CREATE PROMOTION

HTTP Verb + URI:

POST /**promotion**

Description

This operation creates a promotion entity.

Mandatory and Non Mandatory Attributes

The following tables provide the list of mandatory and non mandatory attributes when creating a Promotion, 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
name	
type	

Non Mandatory Attributes	Default Value	Rule
validFor		
description		
lastUpdate		
promotionPattern		

Additional Rules

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

Context	Mandatory Sub-Attributes

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
lifecycleStatus	Draft

Usage Samples

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

REQUEST
POST /promotion/v1/promotion Content-type: application/json <pre> { "name": "Promotion of birthday", "description": "VIP Customer can get gift on the birthday if the customer recharges over \$50 or the same customer purchase \$100 monthly fee in one order", "lastUpdate": "2016-05-21 T04:00:00.0Z", "validFor": { "startDateTime": "2017-12-19 T04:00:00.0Z", "endDateTime": "2017-12-31 T20:42:23.0Z" }, "lifecycleStatus": "Test", "promotionPattern ": { "id": "1109", "name": "Gift for <u>recharge</u> or purchase on birthday", </pre>

```
    "priority": "0",
    "relationTypeAmongGroup": "AND",
    "promotionCriteriaGroup":
    [
      {
        "id": "Group.B",
        "groupName": "recharge amount or payment amount as condition",
        "relationTypeInGroup": "OR"
        "promotionCriteria": [
          {
            "id": "Group.B.1",
            "criteriaPara": "5.1",
            "criteriaOperator": ">=",
            "criteriaValue": "50"
          },
          {
            "id": "Group.B.2",
            "criteriaPara": "2.4",
            "criteriaOperator": ">=",
            "criteriaValue": "100"
          }
        ]
      }
    ],
    "promotionAction":
    [{
      "actionValue": "1"
      "actionObjectId": "2001",
      "actionType": "3.1",
      "id": "2209"
    }]
  }
}
```

RESPONSE

```
201
Content-Type: application/json

{
  "id": "1001",
  "href": "http://serverlocation:port/promotion/v1/promotion/1001",
  "name": "Promotion of birthday",
  "description": "VIP Customer can get gift on the birthday if the customer
recharges over $50 or the same customer purchase $100 monthly fee in one
order",
  "lastUpdate": "2016-05-21 T04:00:00.0Z",
  "validFor":
  {
    "startDateTime": "2017-12-19 T04:00:00.0Z",
    "endDateTime": "2017-12-31 T20:42:23.0Z"
  },
  "lifecycleStatus": "Test",

  "promotionPattern ":
  {
    "id": "1109",
    "name": "Gift for recharge or purchase on birthday",
    "priority": "0",
    "relationTypeAmongGroup": "AND",
    "promotionCriteriaGroup":
    [
      {
        "id": "Group.B",
        "groupName": "recharge amount or payment amount as condition",
        "relationTypeInGroup": "OR"
        "promotionCriteria": [
          {
            "id": "Group.B.1",
            "criteriaPara": "5.1",
            "criteriaOperator": ">=",
            "criteriaValue": "50"
          },
          {
            "id": "Group.B.2",
```

```

        "criteriaPara": "2.4",
        "criteriaOperator": ">=",
        "criteriaValue": "100"
      }
    ]
  },
  "promotionAction":
  [
    {
      "actionValue": "1"
      "actionObjectId": "2001",
      "actionType": "3.1",
      "id": "2209"
    }
  ]
}
}

```

PATCH PROMOTION

HTTP Verb + URI:

PATCH /promotion/{id}

Description

This operation allows partial updates of a promotion 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
lifecycleStatus	
name	
validFor	
type	
description	
lastUpdate	
promotionPattern	

Non Patchable Attributes	Rule
id	
href	

Usage Samples

Here's an example of a request for patching a Promotion resource. It is a request for changing the contact medium.

Request
PATCH /promotion/v1/promotion/5074 Content-Type: application/merge-patch+json <pre>{ "name": "Promotion of new year", "validFor": { "startDateTime": "2018-1-1 T04:00:00.0Z", "endDateTime": "2018-2-15 T20:42:23.0Z" }, "lifecycleStatus": "Release", }</pre>
Response
201 Content-Type: application/json <i>Following a whole representation of the promotion resource with all its attributes. Refer to promotion Resource.</i>

DELETE PROMOTION

HTTP Verb + URI:

DELETE /promotion/{id}

Note: this operation is available only to ADMIN API users

Description

This operation deletes a Promotion entity.

Usage Samples

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

Request
DELETE /promotion/v1/promotion/42
Response
204

API NOTIFICATION

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

It is assumed that the Publish/Subscribe 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.

REQUEST
DELETE /api/hub/{id} 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 <pre>{ "event": { EVENT BODY }, "eventType": "EVENT_TYPE" }</pre>
Response
201

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

ACKNOWLEDGMENTS

RELEASE HISTORY

Release Number	Date	Release led by:	Description
Release 1.0	5/9/2017		First Release of Draft Version of the Document.
Release 2.0	6/14/2017		Updated version to address comments.
Release 17.0.1 Version 2.0.1	11/26/2017	Adrienne Walcott	Updated to reflect TM Forum Approved Status

CONTRIBUTORS TO DOCUMENT

<ul style="list-style-type: none"> MaXu, Huawei maxu@huawei.com Hongxia Hao, Huawei haohongxia@huawei.com 	Initial version
---	-----------------