



Frameworkx Specification

Change Management API REST Specification

TMF655
Release 16.5.1
April 2017

Latest Update: Frameworkx Release 16.5	TM Forum Approved
Version 1.2.1	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:

240 Headquarters Plaza,
East Tower – 10th Floor,
Morristown, NJ 07960 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 Figures	6
Introduction	7
SAMPLE USE CASES	8
CHANGE MANAGEMENT GOVERNANCE IN VIP ASSURANCE FOR PLANNED SERVICE OUTAGE	8
RESOURCE MODEL.....	10
Managed Entity and Task Resource Models	10
Change request Resource	10
Change Request lifecycle	17
<i>Notification Resource Models</i>	18
Change Request Creation Notification.....	19
Change Request Attribute Value Change Notification.....	20
Change Request Remove Notification	20
API OPERATIONS.....	21
Operations on Change Request	21
List Change request	21
Retrieve Change Request	22
Create Change Request.....	23
Patch Change Request.....	25
Delete Change Request	26
API NOTIFICATIONS.....	28
Register listener	28
Unregister listener	28
Publish Event to listener	29
Acknowledgements.....	31

Release History	31
Contributors to Document.....	31

LIST OF FIGURES

Fig.1. Example of planned service outage scenario	8
Fig.2. Example of planned service outage scenario	9
Fig.3. Change Request Resource	10

INTRODUCTION

The following document is the specification of the REST API for Change Management. It includes the model definition as well as all available operations. Possible actions are creating, updating and retrieving Change Request.

Change Management process is to respond to the customer's changing business requirements while maximizing value and reducing incidents, disruption and network. The Change Management API provides the standard integration capabilities between external applications and Change Management Application. The API consists of a simple set of operations that interact with Change Request in a consistent manner. A Change Request is an IT service management discipline. The objective of change management in this context is to ensure that standardized methods and procedures are used for efficient and prompt handling of all changes to control IT infrastructure and Network, in order to minimize the number and impact of any related incidents upon service.

Change Request API performs the following operations on Change Request:

- Retrieval of a change request or a collection of change request s depending on filter criteria
- Partial update of a change request (including approve/reject, complete, abort and worklog exchange etc)
- Creation of a change request (including default values and creation rules)
- Deletion of change request (for administration purposes e.g. backup & archive)
- Notification of events on change request.

SAMPLE USE CASES

CHANGE MANAGEMENT GOVERNANCE IN VIP ASSURANCE FOR PLANNED SERVICE OUTAGE

To decrease Customers impact at Construction work and Request for Changes (RFCs), NW Service Provider changes routes with planned service outage. The schedule is adjusted based on customer’s impact. Customer can query for checking the outage schedule and status.

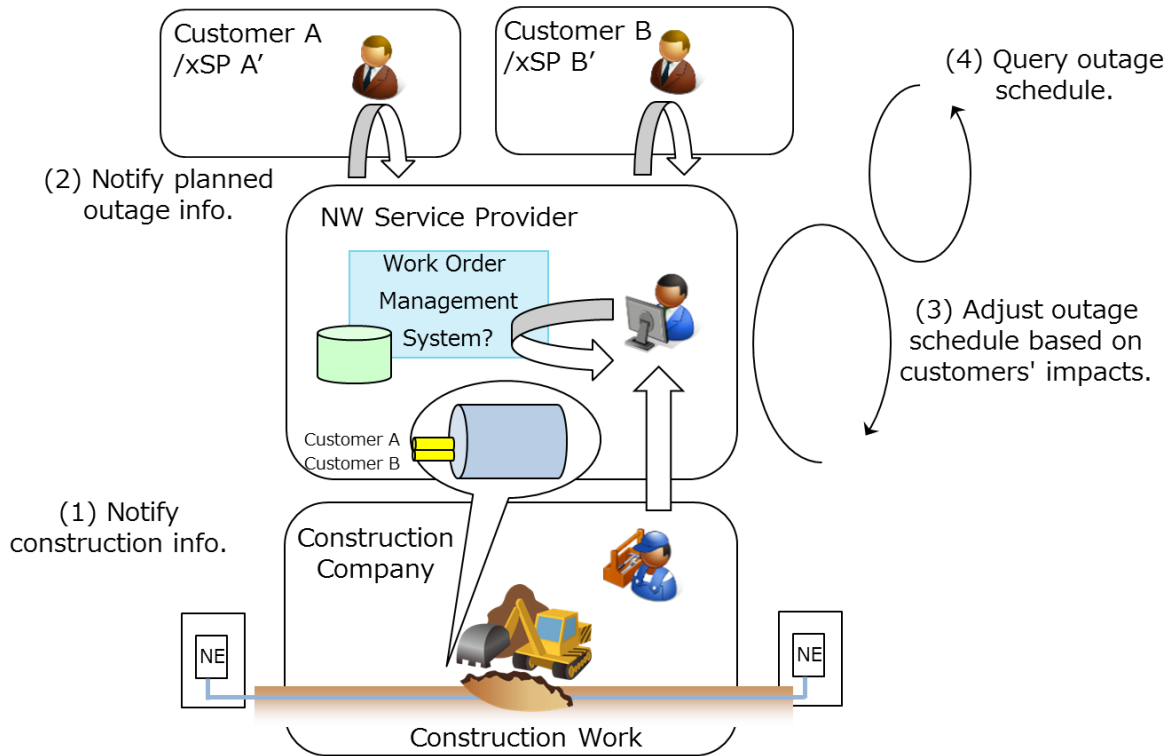


Fig.1. Example of planned service outage scenario

The Supplier has identified that the PR resolution will require disruption to common infrastructure /equipment serving other services owned by that Buyer. Other Buyers’ services are not affected. The Supplier requests permission to disrupt those services at a specified date/time, which the Buyer authorizes or rejects.

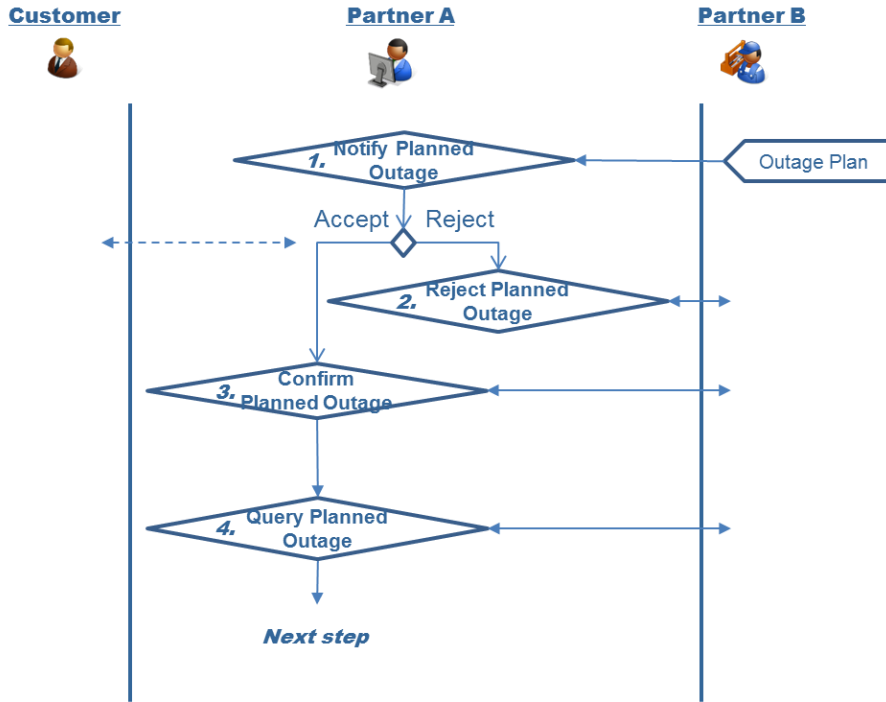


Fig.2. Example of planned service outage scenario

RESOURCE MODEL

MANAGED ENTITY AND TASK RESOURCE MODELS

CHANGE REQUEST RESOURCE

Resource Model

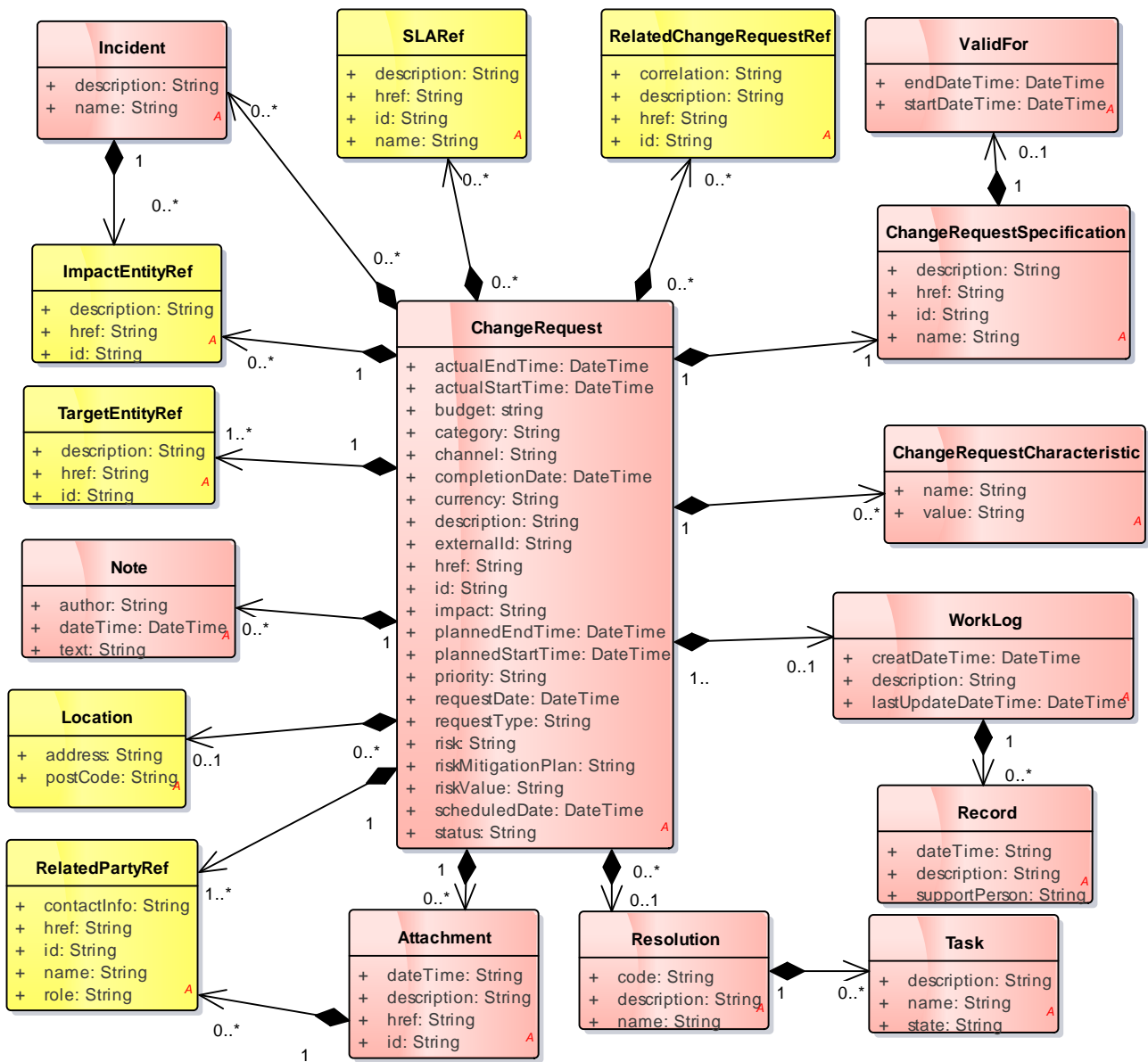


Fig.3. Change Request Resource

Field descriptions

Attachment fields

A document or other file attached to the information about a Change Request to help explain the change being implemented as part of this Change Request.

dateTime	A String. Date and time of the attachment generated.
description	A String. The description of the attachment.
href	A String. Reference of the attachment.
id	A String. Identifier of the attachment.

relatedPartyRef	Defines parties which are involved in the change request and the role they are playing. It may be the customer, operator or 3 rd party.
-----------------	--

ChangeRequest fields

Change Request is a type of request which can be used for the management and control of Change Management process between a customer and a service provider or between a service provider and a partner and vice versa.

actualEndTime	A DateTime. Date and time when the change implementation is finished actually.
actualStartTime	A DateTime. Date and time when the change implementation is started actually.
budget	A string. The budget reserved for the change.
category	A String. The Category of change request
channel	A String. A channel represents the way the Change Request was created
completionDate	A DateTime. Date and time when the change request is confirmed to be completed
currency	A String. The used currency related with cost
description	A String. Description of the change request.
externalId	A String. ID given by the requestor to facilitate the relationship set up and searches afterwards)
href	A String. Hyperlink to access a change request
id	A String. Identifier of a Change Request, it is created on repository side (Change Management system).
impact	A String. Indicates how about the impact by this change
plannedEndTime	A DateTime. Date and time when the change implementation is planned to be finished.
plannedStartTime	A DateTime. Date and time when the change implementation is planned to be started.
priority	A String. A way that can be used by consumers to prioritize change request in Change Management system
requestDate	A DateTime. Date and time when the change request is raised
requestType	A String. Indicates the type of the change request.
risk	A String. The risk to implement this change request.
riskMitigationPlan	A String. The risk mitigation plan.
riskValue	A String. The additional cost if the risk will happen.
scheduledDate	A DateTime. Date and time that the schedule is made.
status	A String. Status of the change request and it's a sub-state.
SLARef	A component that holds details of the SLA being applied to the change request.
changeRequestCharacteristic	Characteristics of the change request to instantiate or to modify
changeRequestSpecification	Defines a set of attributes related to Change Request.
workLog	A record of the work performed on the change request during the investigation and resolution process.
targetEntityRef	The entity which will be changed or configured while the change request is executed.
impactEntityRef	Defines the entities (product, service, resource, and other related objects) which will be impacted while the change request is executed.
relatedChangeRequestRef	An existing Change Request that has some form of correlation with the given Change Request.
location	The place at which the change request occurred.
note	Defines the additional information of the change request.
resolution	The way one or more change request has been implementation through a direct remedy or task.
relatedPartyRef	Defines parties which are involved in the change request and the role they are playing. It may be the customer, operator or 3 rd party.
attachment	A document or other file attached to the information about a Change Request to help explain the change being implemented as part of this Change Request.
incident	An Incident is a record of an event that has altered the operational state of a entity (Resource, Service or Customer's Product).

ChangeRequestCharacteristic fields

Characteristics of the change request to instantiate or to modify

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

ChangeRequestSpecification fields

Defines a set of attributes related to Change Request.

description	A String. Description of the specification
href	A String. Hyperlink to access the specification.
id	A String. Identifier of the specification.
name	A String. Name of the specification.
validFor	A valid duration of a thing.

ImpactEntityRef fields

Defines the entities (product, service, resource, and other related objects) which will be impacted while the change request is executed.

description	A String. Description and analysis of the impact of the change request on the Product, Service or Resource.
href	A String. Hyperlink to access the impacted party, it could be a party reference or a party role reference.
id	A String. Identifier of impacted entity.

Incident fields

An Incident is a record of an event that has altered the operational state of a entity (Resource, Service or Customer's Product).

description	A String. The description of the incident.
name	A String. The name of the incident.

Location fields

The place at which the change request occurred.

address	A String. The detail address of the location.
postCode	A String. The post code of an address.

Note fields

Defines the additional information of the change request.

author	A String. Indicates who provides the description.
dateTime	A DateTime. Date and time when the description was inputted.
text	A String. The additional description.

Record fields

A record in a worklog.

dateTime	A String. The date time that a record is generated.
description	A String. The detail description in a record.
supportPerson	A String. The person who logged that record.

RelatedChangeRequestRef fields

An existing Change Request that has some form of correlation with the given Change Request.

correlation	A String. The correlation between two change requests
description	A String. Description of a change request.
href	A String. Hyper link to access a change request.
id	A String. Identifier of an Change Request

RelatedPartyRef fields

Defines parties which are involved in the change request and the role they are playing. It may be the customer, operator or 3rd party.

contactInfo	A String. Contact information of the party.
href	A String. Hyperlink to access a related party.

id	A String. Identifier of a related party.
name	A String. Name of related party.
role	A String. Role of related Party.

Resolution fields

The way one or more change request has been implementation through a direct remedy or task.

code	A String. The resolution's code, it can be configured as a multiple level hierarchy.
description	A String. The description of the resolution.
name	A String. The name of the resolution
task	A step or task along in the process of implementation a Change Request.

SLARef fields

A component that holds details of the SLA being applied to the change request.

description	A String. Description of a SLA
href	A String. Hyper link to a SLA
id	A String. Identifier of a SLA
name	A String. The name of a SLA

TargetEntityRef fields

The entity which will be changed or configured while the change request is executed.

description	A String. The description to the related party.
href	A String. Hyperlink to access the target entity.
id	A String. Identifier of target entity

Task fields

A step or task along in the process of implementation a Change Request.

description	A String. The description of the task.
name	A String. The name of the task.
state	A String. The state of the task.

ValidFor fields

A valid duration of a thing.

endTime	A DateTime. The end date and time.
startTime	A DateTime. The start date and time.

WorkLog fields

A record of the work performed on the change request during the investigation and resolution process.

createDateTime	A DateTime. Date and time of worklog generated.
description	A String. The description of the worklog.
lastUpdateDateTime	A DateTime. Date and time when the worklog updated.
Record	A record in a worklog.

JSON representation sample

We provide below the JSON representation of an example of Change Request Resource object:

```
{
  "id": "20",
  "href": "http://serverlocation:port/changeManagement/changeRequest/20",
  "actualEndTime": "2013-04-19T16:42:23+0800",
  "actualStartTime": "2013-04-19T16:42:23+0800",
  "budget": "1000",
  "category": "The Name",
  "channel": "web",
  "completionDateTime": "2013-04-19T16:42:23+0800",
  "currency": "$",
  "description": "the description of change request",
```

```

"externalId": "ChangeRequest_22",
"impact": "Non-service affecting",
"plannedEndTime": "2013-04-19T16:42:23+0800",
"plannedStartTime": "2013-04-12T16:42:23+0800",
"priority": "Critical ",
"requestDate": "2013-04-12T16:42:23+0800",
"requestType": "Normal",
"risk": "only 2 engineers have the knowledge to execute the task",
"riskMitigationPlan": "document encoding with base64",
"riskValue": "500",
"scheduledDateTime": "2013-04-12T16:42:23+0800",
"status": "Scheduled for Approval",
"SLARef":[
  {
    "id": "2",
    "href": "http://serverlocation:port/SLAManagement/SLA/2",
    "description": "SLA description",
    "name": "name2"
  },
  {
    "id": "3",
    "href": "http://serverlocation:port/SLAManagement/SLA/3",
    "description": "SLA description",
    "name": "name3"
  }
],
"changeRequestCharacteristic": [
  {
    "name": "additionalRequest",
    "value": "text"
  },
  {
    "name": "test",
    "value": "text"
  }
],
"changeRequestSpecification": {
  "id": "20",
  "href": "http://serverlocation:port/changeManagement/changerequestspecification/20",
  "name": "ChangeRequestSpec_SiteMaintenance",
  "description": "Change Request Specification for Site Maintenance",
  "validFor":
  {
    "startDateTime": "2013-04-12T16:42:23+0800",
    "endDateTime": "2018-04-12T16:42:23+0800"
  }
},
"workLog": {
  "description": "approved by email from Jack",
  "creatDateTime": "2015-10-23T08:00:06+0800",
  "creatDateTime": "2015-10-24T08:00:06+0800",
  "record": [
    {
      "dateTime": "2015-10-23T08:00:06+0800",
      "description": "receive the change request",
      "supportPerson": "Bob"
    },
    {
      "dateTime": "2015-10-24T08:00:06+0800",

```

```
    "description": "Scheduled for Approval",
    "supportPerson": "Bob"
  },
]
},
"targetEntity": [
  {
    "id": "11",
    "href": "http://serverlocation:port/serviceInventory/11",
    "description": "This is the target"
  }
],
"impactedEntity": [
  {
    "id": "12",
    "href": "http://serverlocation:port/serviceInventory/12",
    "description": "This is a related entity"
  },
  {
    "id": "13",
    "href": "http://serverlocation:port/serviceInventory/13",
    "description": "This is a related entity"
  }
],
"relatedChangeRequestRef": [
  {
    "id": "19",
    "href": "http://serverlocation:port/changeManagement/changeRequest/19",
    "description": "the former change request",
    "correlation": "former",
  }
],
"location": {
  "address": "The address of the location",
  "postCode": "the post code of the location"
},
"note": [
  {
    "text": "CreatedBy John at 2015-10-23T08: 00: 05+0800",
    "dateTime": "2015-10-23T08: 00: 05+0800",
    "author": "John"
  },
  {
    "text": "Approved Smith at 2015-10-23T08: 00: 06+0800",
    "dateTime": "2015-10-23T08: 00: 06+0800",
    "author": "Smith"
  }
],
"relatedParty": [
  {
    "role": "requestor",
    "id": "100",
    "href": "http://serverlocation:port/partyManagement/customer/100",
    "name": "John Doe",
    "contactInfo": {
      "email": "myAddress@hotmail.com",
      "phone": "+861088888889"
    }
  }
]
```

```

    }
  },
  {
    "role": "currentHandler",
    "id": "101",
    "href": "http://serverlocation:port/partyManagement/customer/101",
    "name": "Pierre",
    "contactInfo": {
      "email": "myAddress@hotmail.com",
      "phone": "+861088888888"
    }
  }
],
"resolution": {
  "code": "223-112-21-23",
  "description": "need to pause the network",
  "name": "pause network",
  "task": [
    {
      "description": "pause the network",
      "name": "pause",
      "state": "plan to do",
    }
  ]
},
"attachment": [
  {
    "id": "121",
    "href": "http://serverlocation:port/contenetManagement/file/121",
    "dateTime": " 2015-10-23T08: 00: 06+0800",
    "type": "file",
    "description": "Approval mail from Jack",
    "relatedParty": {
      "role": "requestor",
      "id": "100",
      "href": "http://serverlocation:port/partyManagement/operator/100",
      "name": "John Doe"
    }
  },
  {
    "id": "122",
    "href": "http://serverlocation:port/contenetManagement/file/122",
    "dateTime": " 2015-10-23T08: 00: 06+0800",
    "type": "file",
    "description": "Impact estimation document",
    "relatedParty": {
      "role": "requestor",
      "id": "100",
      "href": "http://serverlocation:port/partyManagement/operator/100",
      "name": "John Doe"
    }
  }
],
"incident": {
  "description": "prepare to pause a network",
  "name": "preparation",
  "impactEntity": [
    {
      "id": "11",

```



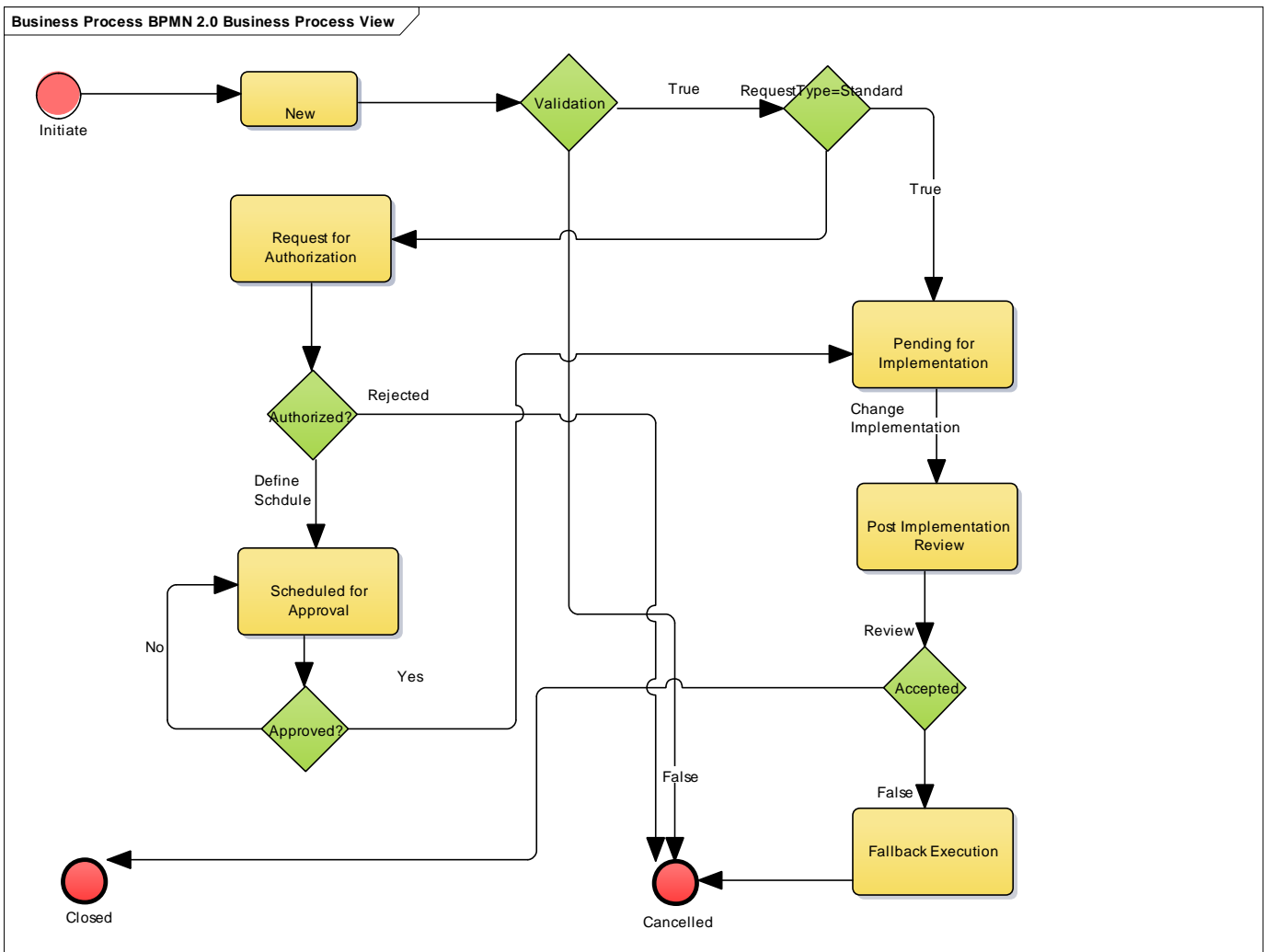
```

    "href": "http://serverlocation:port/serviceInventory/11",
    "description": "This is the target that need to be pause"
  }
]
}
}

```

CHANGE REQUEST LIFECYCLE

Here is the state machine diagram for a change request. Each change request states are described in the tab below.



New	The Change Request is submitted.
Request for Authorization	The Change Request is waiting authorization
Scheduled for Approval	The Change Request is scheduled and waiting the approval for execution
Pending for Implementation	The Change Request is approved and waiting for execution according pre-defined schedule.
Post Implementation Review	The Change Request is executed and waiting the review for result

Fallback Execution	The result is not successful and needs to fallback
Cancelled	The Change Request is cancelled
Closed	The Change Request is closed after review.

NOTIFICATION RESOURCE MODELS

3 notifications are defined for this API

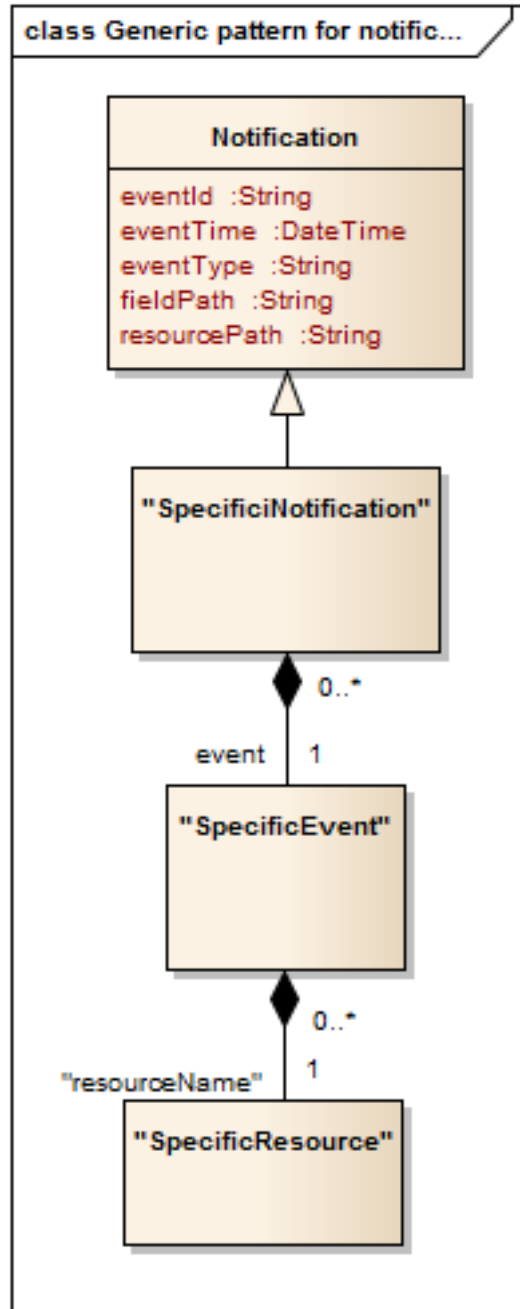
Notifications related to ChangeRequest:

- ChangeRequestCreationNotification
- ChangeRequestAttributeValueChangeNotification
- ChangeRequestRemoveNotification

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).



CHANGE REQUEST CREATION NOTIFICATION

Notification sent when a new ChangeRequest resource is created.

Json representation sample

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

```

{
  "eventId":"00004",
  "eventTime":"2016-03-01T16:42:25",
  "eventType":"ChangeRequestCreationNotification",
  "event": {
    "ChangeRequest" :
  }
}
    
```

```
    {-- SEE ChangeRequest RESOURCE SAMPLE --}  
  }  
}
```

CHANGE REQUEST ATTRIBUTE VALUE CHANGE NOTIFICATION

Notification sent when changing an attribute of a ChangeRequest resource.

Json representation sample

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

```
{  
  "eventId":"00005",  
  "eventTime":"2016-03-02T16:42:25",  
  "eventType":" ChangeRequest AttributeValueChangeNotification",  
  "event": {  
    "ChangeRequest" :  
      {-- SEE ChangeRequest RESOURCE SAMPLE --}  
  }  
}
```

CHANGE REQUEST REMOVE NOTIFICATION

Notification sent when changing an attribute of a ChangeRequest resource.

Json representation sample

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

```
{  
  "eventId":"00005",  
  "eventTime":"2016-03-02T16:42:25",  
  "eventType":" ChangeRequestRemoveNotification",  
  "event": {  
    "ChangeRequest" :  
      {-- SEE ChangeRequest 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 CHANGE REQUEST

LIST CHANGE REQUEST

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

Description

This operation list Change Request 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 ChangeRequest resources.

Request
GET /ChangeManagement/ChangeRequest/fields=id,href,targetEntity,impactedEntity&id=20 Accept: application/json
Response
<pre> 200 [{ "id": "20", "href": "http://serverlocation:port/changeManagement/changeRequest/20", "targetEntity": [{ "id": "11", "href": "http://serverlocation:port/serviceInventory/11", "description": "This is the target" }], "impactedEntity": [{ "id": "12", "href": "http://serverlocation:port/serviceInventory/12", "description": "This is a related entity" }, { "id": "13", "href": "http://serverlocation:port/serviceInventory/13", "description": "This is a related entity" }] }] </pre>

RETRIEVE CHANGE REQUEST

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

Description

This operation retrieves a Change Request 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 ChangeRequest resource.

Request
GET /ChangeManagement/ChangeRequest/19?fields=href,id,description Accept: application/json
Response
200 <pre>{ "href": "https://host:port/ChangeManagement/ChangeRequest/19", "id": "19", "description": " the description of change request" }</pre>

CREATE CHANGE REQUEST

POST /ChangeRequest

Description

This operation creates a Change Request entity.

Mandatory and Non Mandatory Attributes

The following tables provides the list of mandatory and non mandatory attributes when creating a ChangeRequest, including any possible rule conditions and applicable default values.

Mandatory Attributes	Rule
status	
id	Automatically generated
href	Automatically generated
priority	
targetEntityRef	
changeRequestSpecification	

Additional Rules

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

Object	Mandatory Attributes
Attachment	"href", "id"
ChangeRequestCharacteristic	"name", "value"
ChangeRequestSpecification	"href", "id"
ImpactEntityRef	"href", "id"
Record	"dateTime"
RelatedChangeRequestRef	"id", "href"
RelatedPartyRef	"href", "id"
Resolution	"name", "description"
SLARef	"id", "href"
TargetEntityRef	"href", "id"
ValidFor	"startDateTime", "endDateTime"
Worklog	"creatDateTime"

Usage Samples

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

Request
POST /ChangeManagement/ChangeRequest Content-Type: application/json <pre>{ "status": "New", "priority": "Critical ", "changeRequestSpecification": { "id": "21", "href": "http://serverlocation:port/changeManagement/changerequestspecification/21", "name": "ChangeRequestSpec_SiteMaintenance", "description": "Change Request Specification for Site Maintenance", "validFor": { "startDateTime": "2013-04-12T16:42:23+0800", "endDateTime": "2018-04-12T16:42:23+0800" } }, "targetEntity": [{ "id": "12", "href": "http://serverlocation:port/serviceInventory/12", "description": "This is the target" }], }</pre>
Response
201 <pre>{ "id": "21", "href": "http://serverlocation:port/changeManagement/changeRequest/21", "status": "New",</pre>


```

"priority": "Critical ",
"changeRequestSpecification": {
  "id": "21",
  "href": "http://serverlocation:port/changeManagement/changerequestspecification/21",
  "name": "ChangeRequestSpec_SiteMaintenance",
  "description": "Change Request Specification for Site Maintenance",
  "validFor":
  {
    "startDateTime": "2013-04-12T16:42:23+0800",
    "endDateTime": "2018-04-12T16:42:23+0800"
  }
},
"targetEntity": [
  {
    "id": "12",
    "href": "http://serverlocation:port/serviceInventory/12",
    "description": "This is the target"
  }
],
}
    
```

PATCH CHANGE REQUEST

PATCH /ChangeRequest/{id}

Description

This operation allows partial updates of a Change Request 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.

Non Patchable Attributes	Rule
id	
href	

Usage Samples

Here's an example of requests for patching a ChangeRequest resource.

Changing the status to 'prospective' (using json-merge)

Request
PATCH /ChangeManagement/ChangeRequest/21 Content-Type: application/merge-patch+json

```
{
  "status": "Request for Authorization"
}
```

Response

```
201
{
  "id": "21",
  "href": "http://serverlocation:port/changeManagement/changeRequest/21",
  "status": "Request for Authorization",
  "priority": "Critical ",
  "changeRequestSpecification": {
    "id": "21",
    "href": "http://serverlocation:port/changeManagement/changerequestspecification/21",
    "name": "ChangeRequestSpec_SiteMaintenance",
    "description": "Change Request Specification for Site Maintenance",
    "validFor":
    {
      "startDateTime": "2013-04-12T16:42:23+0800",
      "endDateTime": "2018-04-12T16:42:23+0800"
    }
  },
  "targetEntity": [
    {
      "id": "12",
      "href": "http://serverlocation:port/serviceInventory/12",
      "description": "This is the target"
    }
  ],
}
```

DELETE CHANGE REQUEST

DELETE /ChangeRequest/{id}

Note: this operation is available only to ADMIN API users

Description

This operation deletes a Change Request entity.

Usage Samples

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

Request

```
DELETE /ChangeManagement/ChangeRequest/21
```

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
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 }}

<pre> }, "eventType": "EVENT_TYPE" }</pre>
Response
201

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

ACKNOWLEDGEMENTS

RELEASE HISTORY

Release Number	Date	Release led by:	Description
Release 1.2	11/28/2016	Pierre Gauthier TM Forum pgauthier@tmforum.org	Version issued for Fx16.5

CONTRIBUTORS TO DOCUMENT

Nicoleta Stoica	Vodafone
Kuang Chunguang	Huawei
Jiang Yisong	Huawei