CATALOG MANAGEMENT

REST SPECIFICATION TEMPLATE

Document Number: <###>

Document Version: : <V1.6>

Date: May, 2014

Document Status: Draft

# NOTICE

Copyright © TeleManagement Forum 2013. 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](http://www.tmforum.org/IPRPolicy/11525/home.html), 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:

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](http://www.tmforum.org/)

TM Forum Web Page: [www.tmforum.org](http://www.tmforum.org/)

# Table of Contents

NOTICE 2

Table of Contents 3

List of Tables 4

Introduction 5

RESOURCE MODEL 8

Managed Entity and Task Resource Models 8

Notification Resource Models 25

API OPERATION TEMPLATES 34

GET /api/<RESOURCE>/{ID} **Erreur ! Signet non défini.**

PUT API/{RESOURCE}/{ID} **Erreur ! Signet non défini.**

PATCH API/{RESOURCE}/{ID} **Erreur ! Signet non défini.**

POST API/{RESOURCE}/{ID} **Erreur ! Signet non défini.**

DELETE API/{RESOURCE}/{ID} 170

API NOTIFICATIOn TEMPLATES 172

REGISTER LISTENER POST /hub 172

UNREGISTER LISTENER DELETE hub/{id} 173

publish {EventTYPE} POST /listener 173

Release History 182

# List of Tables

**Aucune entrée de table d'illustration n'a été trouvée.**

# Introduction

The following document is the template for the REST API documentation.

The catalog management API allows the management of the entire lifecycle of the catalog elements, the consultation of catalog elements during several processes such as ordering process, campaign management, sales management…

# SAMPLE USE CASES

## Lifecycle management Use Case

The catalog management API allows the management of the entire lifecycle of the catalog elements.

UC1: A partner updates his catalog. He notifies his distributor the catalog change. The distributor requests a catalog export. Then, he retrieves the catalog at the provided URL.



UC2: A partner updates his catalog. He notifies all catalog changes in detail to his distributor. This one updates his catalog copy.



UC3: A catalog administrator wants to retrieve effective duration of a product offering based on product offering identifier or other search criteria (GET/productOffering).

UC4: A catalog administrator wants to retrieve all the bundled product offerings included in a product offering bundle (GET/productOffering).

UC5: A catalog administrator wants to update the lifecycle status (from Launched to retired for example) of a product offering (PUT or PATCH/productOffering).

## Order management Use Case

The catalog management API allows the lookup of the catalog elements during order capture.

UC1: A distributor wants to browse product offerings during order capture and to retrieve product offerings belonging to a given category, for a given channel and a given place. He gets all details on product offerings, product offering prices, product specifications and its characteristics (GET/depth).

UC2: A distributor wants to retrieve the characteristics including prices of a given product offering (GET/productOffering).

UC3: A distributor wants to know if a product offering is available at customer’s location (GET/productOffering).

UC4: A distributor wants to know which SLA applies on a given product offering (GET/productOffering).

UC5: A distributor wants to retrieve all product offerings belonging to a partner (GET/productOffering).

# RESOURCE MODEL

## Managed Entity and Task Resource Models

Product Lifecycle Management:

Product Lifecycle Management (PLM) is responsible for managing the entire lifecycle of the catalog element and its underlying components. This include all of the processes required to design, build, deploy, maintain and ultimately retire the catalog element.



When the macro conception of a catalog element is started the first status of the later is “In Study”.

When the conception of the catalog element is accepted its status is changed to “In Design”.

If the design is approved its status is changed to “In Test”.

Then either the test is OK and then its status is changed to “Active” or the test is KO and its status is changed to “Rejected”. The Rejected status is a final status.

When a catalog element is in a “Active” status it means it has been validated and tested, but it is still not available for customers.

When the beginning of marketing is reached, its status is changed to “Launched”. At this moment customers can buy it.

If the catalog element is not launched, its status is changed to “Retired”.

The same status is achieved when a catalog element reaches the end of marketing.

The “Retired” status means it can not be sold to any new customers, but previous customers can still have it.

When no more customer holds the catalog element, its status is changed to “Obsolete” meaning it can be removed from the catalog.

### Catalog

The root entity for catalog management.

A ServiceCatalog is a collection of ServiceCandidates representing ServiceSpecifications.

Below is a representation of the Catalog resource in JSON format.

|  |
| --- |
| {  "id": "10",  "href": "http://serverlocation:port/catalogManagement/catalog/10",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "type": "Service Catalog",  "category": [  {  "id": "12",  "href": "http://serverlocation:port/catalogManagement/category/12",  "version": "2.0",  "name": "Cloud offerings"  }  ],  "name": "Catalog Wholesale Business",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "relatedParty": [  {  "role": "Owner",  "id": "1234",  "href": "http ://serverLocation:port/partyManagement/partyRole/1234"  }, {  "role": "Reviser",  "name": "Roger Collins"  }  ] } |
|  |

Field Descriptions :

**Catalog**: is a collection of ServiceCandidates representing ServiceSpecifications.

|  |  |
| --- | --- |
| Field | Description |
| category | Reference to the category |
| href | Reference of the catalog |
| id | Unique identifier of the catalog |
| lastUpdate | Date and time of the last update |
| lifecycleStatus | Used to indicate the current lifecycle status |
| name | Name of the catalog |
| type | Indicates if the catalog is a product, service or resource catalog |
| validFor | The period for which the catalog is valid |
| version | Catalog version |

**Category**: is used to group product offerings, service and resource candidates in logical containers. Categories can contain other categories and/or product offerings, resource or service candidates.

|  |  |
| --- | --- |
| Field | Description |
| description | Description of the category |
| href | Reference of the catalog |
| id | Unique identifier of the category |
| isRoot | If true, this Boolean indicates that the category is a root of categories |
| lastUpdate | Date and time of the last update |
| lifecycleStatus | Used to indicate the current lifecycle status |
| name | Name of the category |
| parentId | Unique identifier of the parent category |
| validFor | The period for which the category is valid |
| version | Category version |

**RelatedParty**: Defines party or partyRole linked to a specific entity.

|  |  |
| --- | --- |
| Field | Description |
| id | Unique identifier of related party |
| href | Reference of the relatedParty, could be a party reference or a partyRole reference |
| name | Name of the related party |
| role | Role of the related party |
| validFor | The period for which the related party is linked to the entity |

UML model:



### Category

The category resource is used to group service candidates in logical containers.

Categories can contain other categories and/or service candidates.

Resource IDs for categories are strings and are defined by the catalog application.

Below is a representation of the Category resource in JSON format.

|  |
| --- |
| {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/category/42",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "version": "2.0",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": "41",  "isRoot": "false",  "name": "Cloud Services",  "description": "A category to hold all available cloud service offers" }  Or for a root category (e.g. no parent)  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/category/42",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "version": "2.0",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": "",  "isRoot": "true",  "name": "Cloud Services",  "description": "A category to hold all available cloud services" } |

Field Descriptions :

**Category**: is used to group product offerings, service and resource candidates in logical containers. Categories can contain other categories and/or product offerings, resource or service candidates.

|  |  |
| --- | --- |
| Field | Description |
| description | Description of the category |
| href | Reference of the catalog |
| id | Unique identifier of the category |
| isRoot | If true, this Boolean indicates that the category is a root of categories |
| lastUpdate | Date and time of the last update |
| lifecycleStatus | Used to indicate the current lifecycle status |
| name | Name of the category |
| parentId | Unique identifier of the parent category |
| validFor | The period for which the category is valid |
| version | Category version |

UML model:



### Service Candidate

A ServiceCandidate is an entity that makes a ServiceSpecification available to a catalog. A ServiceCandidate and its associated ServiceSpecification may be published - made visible - in any number of ServiceCatalogs, or in none. One ServiceSpecification can be composed of other ServiceSpecifications. The Service Candidate consists of a subset of ServiceSpecification characteristics to fulfill a product offering. **These ServiceSpecifications may also be published**.

The Service Candidate resource (JSON format) represents a set of characteristics that define the values given by the ServiceSpecification.

Resource IDs are numeric and generated by the Catalog application.

Below is a representation of the ServiceCandidate resource in JSON format.

|  |
| --- |
| {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/serviceCandidate/42",  "name": "Virtual Storage Medium",  "version": "2.0",  "description": "Virtual Storage Medium",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "category": [  {  "id": "12",  "href": "http://serverlocation:port/catalogManagement/category/12",  "version": "2.0",  "name": "Cloud service"  }  ],  "serviceLevelAgreement": {  "id": "28",  "href": "http://serverlocation:port/slaManagement/serviceLevelAgreement/28",  "name": "Standard SLA"  },  "serviceSpecification": {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "version": "1.1"  } } |

Field Descriptions :

**Category**: is used to group product offerings, service and resource candidates in logical containers. Categories can contain other categories and/or product offerings, resource or service candidates.

|  |  |
| --- | --- |
| Field | Description |
| description | Description of the category |
| href | Reference of the catalog |
| id | Unique identifier of the category |
| isRoot | If true, this Boolean indicates that the category is a root of categories |
| lastUpdate | Date and time of the last update |
| lifecycleStatus | Used to indicate the current lifecycle status |
| name | Name of the category |
| parentId | Unique identifier of the parent category |
| validFor | The period for which the category is valid |
| version | Category version |

**ServiceCandidate**: Is an entity that makes a ServiceSpecification available to a catalog. A ServiceCandidate and its associated ServiceSpecification may be published - made visible - in any number of ServiceCatalogs, or in none. One ServiceSpecification can be composed of other ServiceSpecifications. The Service Candidate consists of a subset of ServiceSpecification characteristics to fulfill a product offering. These ServiceSpecifications may also be published.

|  |  |
| --- | --- |
| Field | Description |
| description | A narrative that explains in detail what the service candidate is |
| href | Reference of the service candidate |
| id | Unique identifier of the service candidate |
| lastUpdate | Date and time of the last update |
| lifecycleStatus | Used to indicate the current lifecycle status |
| name | Name of the service candidate |
| validFor | The period for which the service candidate is valid |
| version | Service candidate version |

**ServiceLevelAgreement**: A service level agreement (SLA) is a type of agreement that represents a formal negotiated agreement between two parties designed to create a common understanding about products, services, priorities, responsibilities, and so forth. The SLA is a set of appropriate procedures and targets formally or informally agreed between parties in order to achieve and maintain specified Quality of Service.

|  |  |
| --- | --- |
| Field | Description |
| href | Reference of the service level agreement |
| id | Unique identifier of the service level agreement |
| name | Name of the service level agreement |

**ServiceSpecification**: is an abstract base class for representing a generic means for implementing a particular type of Service. In essence, a ServiceSpecification defines the common portions of a set of Services, while Service defines a specific instance that is based on a particular ServiceSpecification.

|  |  |
| --- | --- |
| Field | Description |
| brand | The manufacturer or trademark of the specification |
| description | A narrative that explains in detail what the service specification is |
| href | Reference of the service specification |
| id | Unique identifier of the service specification |
| lastUpdate | Date and time of the last update |
| lifecycleStatus | Used to indicate the current lifecycle status |
| name | Name of the service specification |
| validFor | The period for which the service specification is valid |
| version | Service specification version |

UML model:



### Service Specification

ServiceSpecification is an abstract base class for representing a generic means for implementing a particular type of Service.

In essence, a ServiceSpecification defines the common portions of a set of Services, while Service defines a specific instance that is based on a particular ServiceSpecification.

Note: The configurable attribute on the serviceSpecCharacteristics determines if an instance of the serviceSpecification can override the value of the attribute. If set to true, the value can be overridden, if set to false, the value is set by the catalog and cannot be changed.

Below is a representation of the ServiceSpecification resource in JSON format.

|  |
| --- |
| {  "id": "22",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/22",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "lifecycleStatus": "Active",  "name": "iPhone 42",  "description": "Siri works on this iPhone",  "brand": "Apple",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "attachment": [  {  "id": "56",  "href": "http://serverlocation:port/documentManagment/attachment/56",  "type": "Picture",  "url": "http://xxxxx"  }  ],  "relatedParty": [  {  "role": "Owner",  "id": "1234",  "href": "http ://serverLocation:port/partyManagement/partyRole/1234"  }  ],  "serviceSpecRelationship": [  {  "type": "dependency",  "id": "23",  "href": " http://serverlocation:port/catalogManagement/serviceSpecification/23", "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredServiceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredResourceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/resourceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "serviceSpecCharacteristic": [  {  "name": "Screen Size",  "description": "Screen size",  "valueType": "number",  "configurable": "false",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "serviceSpecCharacteristicValue": [  {  "valueType": "number",  "default": "true",  "value": "4.2",  "unitOfMeasure": "inches",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }, {  "name": "Colour",  "description": "Product colour",  "valueType": "string",  "configurable": "true",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "ServiceSpecCharacteristicValue": [  {  "valueType": "string",  "default": "true",  "value": "Black",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }, {  "valueType": "string",  "default": "false",  "value": "White",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }  ] } |

Field Descriptions :

**Attachment**: describes a product through video, pictures...

|  |  |
| --- | --- |
| Field | Description |
| href | Reference of the attachment |
| id | Unique identifier of the attachment |
| type | Attachment type such as video, picture |
| url | Uniform Resource Locator, is a web page address (a subset of URI) |

**RelatedParty**: Defines party or partyRole linked to a specific entity.

|  |  |
| --- | --- |
| Field | Description |
| id | Unique identifier of related party |
| href | Reference of the relatedParty, could be a party reference or a partyRole reference |
| name | Name of the related party |
| role | Role of the related party |
| validFor | The period for which the related party is linked to the entity |

**RequiredResourceSpecification:** ResourceSpecification required to realize a ResourceSpecification**.**

|  |  |
| --- | --- |
| Field | Description |
| id | Unique identifier of the requiredResourceSpecification |
| href | Reference of the requiredResourceSpecification |
| name | Name of the requiredResourceSpecification |
| validFor | The period for which the RequiredResourceSpecification is valid |

**RequiredServiceSpecification:** ServiceSpecification required to realize a ServiceSpecification.

|  |  |
| --- | --- |
| Field | Description |
| id | Unique identifier of the requiredServiceSpecification |
| href | Reference of the requiredServiceSpecification |
| name | Name of the requiredServiceSpecification |
| validFor | The period for which the RequiredServiceSpecification is valid |

**ServiceSpecCharacteristic**: This is an abstract base class that represents the key features of this ServiceSpecification. For example, bandwidth is characteristic of many different types of services; if bandwidth is important (e.g., from the point-of-view of a Customer obtaining this Service via a Product) then bandwidth would be a ServiceCharacteristicValue for that particular Service.

|  |  |
| --- | --- |
| Field | Description |
| configurable | If true, the Boolean indicates that the serviceSpecCharacteristic is configurable |
| description | A narrative that explains in detail what the serviceSpecCharacteristic is |
| name | Name of the serviceSpecCharacteristic |
| valueType | A kind of value that the characteristic can take on, such as numeric, text and so forth |
| validFor | The period for which the serviceSpecCharacteristic is valid |

**ServiceSpecCharacteristicValue**: Is used to define a set of attributes, each of which can be assigned to a corresponding set of attributes in a ServiceSpecCharacteristic object. The values of the attributes in the ServiceSpecCharacteristicValue object describe the values of the attributes that a corresponding ServiceSpecCharacteristic object can take on.

|  |  |
| --- | --- |
| Field | Description |
| default | Indicates if the value is the default value for a characteristic |
| unitOfMeasure | Could be minutes, GB… |
| validFor | The period of time for which a value is applicable |
| value | A discrete value that the characteristic can take on |
| valueFrom | The low range value that a characteristic can take on |
| valueTo | The upper range value that a characteristic can take on |
| valueType | A kind of value that the characteristic can take on, such as numeric, text, and so forth |

**ServiceSpecification**: is an abstract base class for representing a generic means for implementing a particular type of Service. In essence, a ServiceSpecification defines the common portions of a set of Services, while Service defines a specific instance that is based on a particular ServiceSpecification.

|  |  |
| --- | --- |
| Field | Description |
| brand | The manufacturer or trademark of the specification |
| description | A narrative that explains in detail what the service specification is |
| href | Reference of the service specification |
| id | Unique identifier of the service specification |
| lastUpdate | Date and time of the last update |
| lifecycleStatus | Used to indicate the current lifecycle status |
| name | Name of the service specification |
| validFor | The period for which the service specification is valid |
| version | Service specification version |

**ServiceSpecRelationship**: allows for Service Specifications to contain other Service Specifications.

|  |  |
| --- | --- |
| Field | Description |
| id | Unique identifier of the serviceSpecification |
| href | Reference of the serviceSpecification |
| validFor | The period for which the serviceSpecRelationship is valid |
| Type | Type of relationship such as migration, substitution, dependency, exclusivity |

UML model:



### Export Job Resource

An ExportJob resource represents a TASK used to export resources to a File

The ExportJob resource supports the following properties:

|  |  |
| --- | --- |
| Attribute name | Description |
| query | Used to scope the exported data (identical to GET filter construct using target ID as base)  “query”: ”type=serviceSpecification&version=2.0” |
| path | URL of the root resource acting as the source for streaming content to the file specified by the ExportJob  ../serviceManagement/serviceSpecification |
| content-type | The format of the exported data .By default “application/json” |
| status | notstarted, running, succeeded, failed |
| url | URL of the File containing the data to be exported  a file URL, which is of the form  file://host/path  where host is the fully qualified domain name of the system on which the path is accessible, and path is a hierarchical directory path of the form directory/directory/.../name |
| completionDate | Date at which the Job was completed. |
| creationDate | Date at which the Job was created. |
| errorLog | Reason for Failure |

JSON representation of an ExportJob

|  |
| --- |
| {  "id": "54",  "href": "http:/api/catalogManagement/exportJob/54",  "status": "running",  "path": "catalogManagement/",  "content-type": "application/json",  "errorLog": "",  "creationDate": "2013-04-19T16:42:23-04:00",  "completionDate": "2013-04-21T16:42:23-04:00",  "url": "ftp://ftp.myCatalog.com/serviceSpecification/54"  } |

UML Model:



### Import Job Resource

An ImportJob resource represent a TASK used to import resources from a File

The ImportJob resource supports the following properties:

|  |  |
| --- | --- |
| Attribute name | Description |
| content-type | The format of the imported data. By default “application/json” |
| path | URL of the root resource where the content of the file specified by the ImportJob must be applied  ../serviceManagement/serviceSpecification |
| status | notstarted, running, succeeded, failed |
| url | URL of the File containing the data to be imported  a file URL, which is of the form  file://host/path  where host is the fully qualified domain name of the system on which the path is accessible, and path is a hierarchical directory path of the form directory/directory/.../name |
| completionDate | Date at which the Job was completed. |
| creationDate | Date at which the Job was created. |
| errorLog | Reason for Failure if status is failed |

JSON representation of an ImportJob

|  |
| --- |
| {  "id": "54",  "href": "http:/api/catalogManagement/importJob/54",  "status": "completed",  "path": "catalogManagement/",  "content-type": "application/json",  "errorLog": "",  "creationDate": "2013-04-19T16:42:23-04:00",  "completionDate": "2013-04-21T16:42:23-04:00",  "url": "ftp://ftp.myCatalog.com/serviceSpecification/54"  } |

UML Model:



## Notification Resource Models

Two notification event types are defined:

1. CatalogChangeNotification
2. CatalogChangeBatchNotification

Catalog Notification UML Diagram:



Catalog change notification: notifies that a catalog has been updated.

|  |
| --- |
| {  "event": {  "id": "01",  "dateTime": "2013-04-19T16:42:25-30:00",  "eventType": "Catalog change notification",  "catalog": [  {  "id": "10",  "href": "http://serverlocation:port/catalogManagement/catalog/10",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "type": "Service Catalog",  "name": "Catalog Wholesale Business"  }  ]  } } |

Catalog change batch notification: notifies in detail all catalog updates.

|  |
| --- |
| {  "event": {  "id": "01",  "dateTime": "2013-04-19T16:42:25-30:00",  "eventType": "Catalog change batch notification",  "catalog": [  {  "id": "10",  "href": "http://serverlocation:port/catalogManagement/catalog/10",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "type": "Service Catalog",  "name": "Catalog Wholesale Business",  "update": [  {  "op": "add",  "path": "/category",  "value": {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/category/42",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "version": "2.0",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": "",  "isRoot": "true",  "name": "Home, Audio/Entertainement",  "description": "A category to hold all inside offers"  }  }, {  "op": "add",  "path": "/category",  "value": {  "id": "421",  "href": "http://serverlocation:port/catalogManagement/catalog/421",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "version": "2.0",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": "42",  "isRoot": "false",  "name": "iPhone",  "description": "A category to hold all iPhone"  }  }, {  "op": "add",  "path": "/serviceSpecification",  "value": {  "id": "14",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/14",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "name": "iPhone 42 ",  "description": "Siri works on this iPhone",  "isBundle": "false",  "brand": "Apple",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "relatedParty": [  {  "role": "Owner",  "id": "1234",  "href": "http ://serverLocation:port/partyManagement/partyRole/1234"  }  ],  "serviceSpecCharacteristic": [  {  "id": "34",  "name": "Colour",  "description": "Colour",  "valueType": "string",  "configurable": "true",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "ProductSpecCharacteristicValue": [  {  "valueType": "string",  "default": "true",  "value": "Black",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }, {  "valueType": "string",  "default": "false",  "value": "White",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }  ]  }  }  ]  }  ]  } } |

Export and Import Job Notifications

1. Export Job Completion Notification
2. Import Job Completion Notification

UML Model:



Event: ExportJobCompletionNotification

|  |
| --- |
| {  "eventType": "ExportJobCompletionNotification",  "eventTime": "2014-09-27T05:46:25.0Z",  "eventId": "1154278",  "event": {"exportJob": {  "id": "54",  "href": "http:/api/catalogManagement/exportJob/54",  "status": "succeeded",  "path": "catalogManagement/",  "content-type": "application/json",  "errorLog": "",  "creationDate": "2013-04-19T16:42:23-04:00",  "completionDate": "2013-04-21T16:42:23-04:00",  "url": "ftp://ftp.mycatalog.com/catalogManagement/54.json"  }} } |

Event: ImportJobCompletionNotification

|  |
| --- |
| {  "eventType": "ImportJobCompletionNotification",  "eventTime": "2014-09-27T05:46:25.0Z",  "eventId": "1154278",  "event": {"importJob": {  "id": "54",  "href": "http:/api/catalogManagement/importJob/54",  "status": "succeeded",  "path": "catalogManagement/",  "content-type": "application/json",  "errorLog": "",  "creationDate": "2013-04-19T16:42:23-04:00",  "completionDate": "2013-04-21T16:42:23-04:00",  "url": "ftp://ftp.mycatalog.com/catalogManagement/54.json"  }} } |

# API OPERATION TEMPLATES

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

Remember that the following Uniform Contract rules must be used :

|  |  |  |
| --- | --- | --- |
| 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.

## GET /catalogManagement/category

This Uniform Contract operation is used to retrieve the representation of a managed entity or a task.

Note that collections can be retrieved via GET /api/<RESOURCE> with no {ID}

Description :

* This operation retrieves productCategory from a catalog. The category resource is used to group product offerings, service and resource candidates in logical containers. Categories can contain other categories and/or product offerings, resource or service candidates.
* The resource represents a managed entity or a collection depending on the query pattern.
* The identifier is a string that can consist of numbers, not necessarily alphanumeric.

Behavior :

* Standard behavior and response codes for GET operations
* Filtering is enabled on all attributes, particularly due to the hierarchical nature of productCategory, children of a productCategory are obtained by filtering on the parentId attribute of the resource.
* Attribute selection is not enabled for this simple resource

|  |
| --- |
| **REQUEST** |
| GET /catalogManagement/category/{42}  Accept: application/json |
| **RESPONSE** |
| 200  Content-Type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/category/42",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "version": "2.0",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": " 41",  "isRoot": "false",  "name": "Cloud Services",  "description": "A category to hold all available cloud service offers" } |

## GET /catalogManagement/serviceCandidate/{ID}

This Uniform Contract operation is used to retrieve the representation of a managed entity or collection.

Note that collections can be retrieved via GET /api/<RESOURCE> with no {ID}

Description :

* This operation returns ServiceCandidates from a catalog. A ServiceCandidate is an entity that makes a ServiceSpecification available to a catalog.
* The resource instance being returned is a ServiceCandidate or an array of ServiceCandidates if the query returns multiple specifications.
* The resource represents a managed entity or a collection.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
  + 200 if no serviceCandidate found
  + 404 Not found when the supplied ID doesn’t match a known ServiceCandidate.
* Filtering is enabled on all ServiceCandidate attributes.
* Attribute selection is enabled.

|  |
| --- |
| **REQUEST** |
| GET /catalogManagement/serviceCandidate/{42}  Accept: application/json |
| **RESPONSE** |
| {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/serviceCandidate/42",  "name": "Virtual Storage Medium",  "version": "2.0",  "description": "Virtual Storage Medium",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "category": [  {  "id": " 12",  "href": "http://serverlocation:port/catalogManagement/category/12",  "version": "2.0",  "name": "Cloud service"  }  ],  "serviceLevelAgreement": {  "id": " 28",  "href": "http://serverlocation:port/slaManagement/serviceLevelAgreement/28",  "name": "Standard SLA"  },  "serviceSpecification": {  "id": " 13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "version": "1.1"  } } |

## GET /catalogManagement/serviceSpecification/{ID}

This Uniform Contract operation is used to retrieve the representation of a managed entity or collection.

Note that collections can be retrieved via GET /api/<RESOURCE> with no {ID}

Description :

* This operation returns ServiceSpecifications from a catalog. This is an abstract base class for defining the ServiceSpecification hierarchy. All Services are characterized as either being possibly visible and usable by a Customer or not.
* The resource instance being returned is a ServiceSpecification or an array of ServiceSpecifications if the query returns multiple specifications.
* The resource represents a managed entity or a collection.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
  + 200 if no ServiceSpecification found
  + 404 Not found when the supplied ID doesn’t match a known ServiceSpecification.
* Filtering is enabled on all ServiceSpecification attributes.
* Attribute selection is enabled.

|  |
| --- |
| **REQUEST** |
| GET /catalogManagement/serviceSpecification/{22}  Accept: application/json |
| **RESPONSE** |
| {  "id": "22",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/22",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "lifecycleStatus": "Active",  "name": "iPhone 42",  "description": "Siri works on this iPhone",  "brand": "Apple",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "attachment": [  {  "id": " 56",  "href": "http://serverlocation:port/documentManagment/attachment/56",  "type": "Picture",  "url": "http://xxxxx"  }  ],  "relatedParty": [  {  "role": "Owner",  "id": " 1234",  "href": "http ://serverLocation:port/partyManagement/partyRole/1234"  }  ],  "serviceSpecRelationship": [  {  "type": "dependency",  "id": " 23",  "href": " http://serverlocation:port/catalogManagement/serviceSpecification/23", "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredServiceSpecification": [  {  "id": " 13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredresourceSpecification": [  {  "id": " 13",  "href": "http://serverlocation:port/catalogManagement/resourceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "serviceSpecCharacteristic": [  {  "name": "Screen Size",  "description": "Screen size",  "valueType": "number",  "configurable": "false",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "serviceSpecCharacteristicValue": [  {  "valueType": "number",  "default": "true",  "value": "4.2",  "unitOfMeasure": "inches",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }, {  "name": "Colour",  "description": "Product colour",  "valueType": "string",  "configurable": "true",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "ServiceSpecCharacteristicValue": [  {  "valueType": "string",  "default": "true",  "value": "Black",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }, {  "valueType": "string",  "default": "false",  "value": "White",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }  ] } |

## GET /catalogManagement/exportJob/{ID}

This Uniform Contract operation is used to retrieve the representation of a managed entity or collection.

Note that collections can be retrieved via GET /api/<RESOURCE> with no {ID}

Description :

* This operation returns the status of an exportJob.
* The resource represents a managed entity or a collection.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
  + 200 if no ExportJob found
  + 404 Not found when the supplied ID doesn’t match a known ExportJob.
* Filtering is enabled on all ExportJob attributes.
* Attribute selection is enabled.

|  |
| --- |
| GET /catalogManagement/exportJob/{10}  Content-type: application/json |
| **RESPONSE** |
| 200  Content-Type: application/json  {  "id": "10",  "href": "http:/api/catalogManagement/exportJob/54",  "status": "running",  "path": "catalogManagement/",  "content-type": "application/json",  "errorLog": "",  "creationDate": "2013-04-19T16:42:23-04:00",  "completionDate": "2013-04-21T16:42:23-04:00",  "url": "ftp://ftp.catalogManagement.com/catalog/54"  } |

## GET /catalogManagement/importJob/{ID}

This Uniform Contract operation is used to retrieve the representation of a managed entity or collection.

Note that collections can be retrieved via GET /api/<RESOURCE> with no {ID}

Description :

* This operation returns the status of an importJob.
* The resource represents a managed entity or a collection.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
  + 200 if no ImportJob found
  + 404 Not found when the supplied ID doesn’t match a known ImportJob.
* Filtering is enabled on all ImportJob attributes.
* Attribute selection is enabled.

|  |
| --- |
| GET /catalogManagement/importJob/{54}  Content-type: application/json |
| **RESPONSE** |
| 200  Content-Type: application/json  {  "id": "54",  "href": "http:/api/catalogManagement/importJob/54",  "status": "running",  "path": "catalogManagement",  "content-type": "application/json",  "errorLog": "",  "creationDate": "2013-04-19T16:42:23-04:00",  "completionDate": "2013-04-21T16:42:23-04:00",  "url": "ftp://ftp.myCatalog.com/catalog.json"  } |

## PUT /catalogManagement/category/{ID}

This Uniform Contract operation is used to completely update the representation of a managed entity or a task.

Description :

* This operation completely updates a category.
* The resource represents a managed entity.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
* Returns HTTP/1.1 status code 201 if the request was successful.
* Any other special return and/or exception codes.

|  |
| --- |
| **REQUEST** |
| PUT /catalogManagement/category{42}  Content-type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/category/42",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "version": "2.0",  "lifecycleStatus": "Launched",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": " 41",  "isRoot": "false",  "name": "Cloud Services",  "description": "A category to hold all available cloud service offers" } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/category/42",  "lastUpdate": "2013-04-25T09:28:10-04:00",  "version": "2.0",  "lifecycleStatus": "Launched",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": "41",  "isRoot": "false",  "name": "Cloud Services",  "description": "A category to hold all available cloud service offers" } |

## PUT /catalogManagement/serviceCandidate/{ID}

This Uniform Contract operation is used to completely update the representation of a managed entity or a task.

Description :

* This operation completely updates a service candidate.
* The resource represents a managed entity.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
* Returns HTTP/1.1 status code 201 if the request was successful.
* Any other special return and/or exception codes.

|  |
| --- |
| **REQUEST** |
| PUT /catalogManagement/serviceCandidate{42}  Content-type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/serviceCandidate/42",  "name": "Virtual Storage Medium",  "version": "2.0",  "description": "Virtual Storage Medium",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "lifecycleStatus": "Launched",  "validFor": {  "startDateTime": "2013-06-19T00:00:00-04:00",  "endDateTime": ""  },  "category": [  {  "id": "12",  "href": "http://serverlocation:port/catalogManagement/category/12",  "version": "2.0",  "name": "Cloud service"  }  ],  "serviceLevelAgreement": {  "id": "28",  "href": "http://serverlocation:port/slaManagement/serviceLevelAgreement/28",  "name": "Standard SLA"  },  "serviceSpecification": {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "version": "1.1"  } } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/serviceCandidate/42",  "name": "Virtual Storage Medium",  "version": "2.0",  "description": "Virtual Storage Medium",  "lastUpdate": "2013-06-19T00:00:00-04:00",  "lifecycleStatus": "Launched",  "validFor": {  "startDateTime": "2013-06-19T00:00:00-04:00",  "endDateTime": ""  },  "category": [  {  "id": "12",  "href": "http://serverlocation:port/catalogManagement/category/12",  "version": "2.0",  "name": "Cloud service"  }  ],  "serviceLevelAgreement": {  "id": "28",  "href": "http://serverlocation:port/slaManagement/serviceLevelAgreement/28",  "name": "Standard SLA"  },  "serviceSpecification": {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "version": "1.1"  } } |

## PUT /catalogManagement/serviceSpecification/{ID}

This Uniform Contract operation is used to completely update the representation of a managed entity or a task.

Description :

* This operation completely updates a service specification.
* The resource represents a managed entity.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
* Returns HTTP/1.1 status code 201 if the request was successful.
* Any other special return and/or exception codes.

|  |
| --- |
| **REQUEST** |
| PUT /catalogManagement/serviceSpecification{22}  Content-type: application/json  {  "id": "22",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/22",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "lifecycleStatus": "Launched",  "name": "iPhone 42",  "description": "Siri works on this iPhone",  "brand": "Apple",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "attachment": [  {  "id": "56",  "href": "http://serverlocation:port/documentManagment/attachment/56",  "type": "Picture",  "url": "http://xxxxx"  }  ],  "relatedParty": [  {  "role": "Owner",  "id": "1234",  "href": "http ://serverLocation:port/partyManagement/partyRole/1234"  }  ],  "serviceSpecRelationship": [  {  "type": "dependency",  "id": "23",  "href": " http://serverlocation:port/catalogManagement/serviceSpecification/23", "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredServiceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredResourceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/resourceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "serviceSpecCharacteristic": [  {  "name": "Screen Size",  "description": "Screen size",  "valueType": "number",  "configurable": "false",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "serviceSpecCharacteristicValue": [  {  "valueType": "number",  "default": "true",  "value": "4.2",  "unitOfMeasure": "inches",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }, {  "name": "Colour",  "description": "Product colour",  "valueType": "string",  "configurable": "true",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "ServiceSpecCharacteristicValue": [  {  "valueType": "string",  "default": "true",  "value": "Black",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }, {  "valueType": "string",  "default": "false",  "value": "White",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }  ] } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "22",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/22",  "version": "2.0",  "lastUpdate": "2013-06-19T16:42:23-04:00",  "lifecycleStatus": "Launched",  "name": "iPhone 42",  "description": "Siri works on this iPhone",  "brand": "Apple",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "attachment": [  {  "id": "56",  "href": "http://serverlocation:port/documentManagment/attachment/56",  "type": "Picture",  "url": "http://xxxxx"  }  ],  "relatedParty": [  {  "role": "Owner",  "id": "1234",  "href": "http ://serverLocation:port/partyManagement/partyRole/1234"  }  ],  "serviceSpecRelationship": [  {  "type": "dependency",  "id": "23",  "href": " http://serverlocation:port/catalogManagement/serviceSpecification/23", "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredServiceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredResourceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/resourceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "serviceSpecCharacteristic": [  {  "name": "Screen Size",  "description": "Screen size",  "valueType": "number",  "configurable": "false",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "serviceSpecCharacteristicValue": [  {  "valueType": "number",  "default": "true",  "value": "4.2",  "unitOfMeasure": "inches",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }, {  "name": "Colour",  "description": "Product colour",  "valueType": "string",  "configurable": "true",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "ServiceSpecCharacteristicValue": [  {  "valueType": "string",  "default": "true",  "value": "Black",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }, {  "valueType": "string",  "default": "false",  "value": "White",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }  ] } |

## PATCH /catalogManagement/category/{ID}

This Uniform Contract operation is used to partially update the representation of a managed entity or a task.

Description :

* This operation partially updates a category.
* The resource represents a managed entity.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
* Returns HTTP/1.1 status code 201 if the request was successful.
* Any other special return and/or exception codes.

Specify which attributes are patchable using the following table (to capture RO attributes)

|  |  |  |
| --- | --- | --- |
| Attribute name | Patchable | Rule |
| Id | No |  |
| lastUpdate | No |  |
| version | Y | Should be greater than the current one |
| lifecycleStatus | Y | Status should be valid |
| startDateTime | Y |  |
| endDateTime | Y | Should be greater than startDateTime |
| parentId | Y | The parent category should exist |
| isRoot | Y | To be updated in compliance with the parentId |
| name | Y |  |
| description | Y |  |

Further document any rules that must be implemented when patching attributes.

|  |  |
| --- | --- |
| Rule name | Rule/Pre Condition/Side Effects/Post Conditons |
| Root | When isRoot is patched to false, parentId should indicate the parent category.  When isRoot is patched to true, parentId should be empty. |

|  |
| --- |
| **REQUEST** |
| PATCH /catalogManagement/category/{42}  Content-type: application/json  {  "parentId": "",  "isRoot": "true" } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/category/42",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "version": "2.0",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": "",  "isRoot": "true",  "name": "Cloud Services",  "description": "A category to hold all available cloud service offers" } |

## PATCH /catalogManagement/serviceCandidate/{ID}

This Uniform Contract operation is used to partially update the representation of a managed entity or a task.

Description :

* This operation partially updates a serviceCandidate.
* The resource represents a managed entity.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
* Returns HTTP/1.1 status code 201 if the request was successful.
* Any other special return and/or exception codes.

Specify which attributes are patchable using the following table (to capture RO attributes)

|  |  |  |
| --- | --- | --- |
| Attribute name | Patchable | Rule |
| Id | No |  |
| startDateTime | Y |  |
| endDateTime | Y | Should be greater than startDateTime |
| name | Y |  |
| version | Y |  |
| description | Y |  |
| lastUpdate | N |  |
| lifecycleStatus | Y | Status should be valid |
| category | Y | It should exist |
| serviceSpecification | Y | It should exist |

Further document any rules that must be implemented when patching attributes.

|  |  |
| --- | --- |
| Rule name | Rule/Pre Condition/Side Effects/Post Conditons |
|  |  |

|  |
| --- |
| **REQUEST** |
| PATCH /catalogManagement/serviceCandidate/{42}  Content-type: application/json  {  "lifecycleStatus": "Launched",  "validFor": {  "startDateTime": "2013-06-20T00:00:01-04:00",  "endDateTime": ""  }  } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/serviceCandidate/42",  "name": "Virtual Storage Medium",  "version": "2.0",  "description": "Virtual Storage Medium",  "lastUpdate": "2013-06-20T00:00:01-04:00",  "lifecycleStatus": "Launched",  "validFor": {  "startDateTime": "2013-06-20T00:00:01-04:00",  "endDateTime": ""  },  "category": [  {  "id": "12",  "href": "http://serverlocation:port/catalogManagement/category/12",  "version": "2.0",  "name": "Cloud service"  }  ],  "serviceLevelAgreement": {  "id": "28",  "href": "http://serverlocation:port/slaManagement/serviceLevelAgreement/28",  "name": "Standard SLA"  },  "serviceSpecification": {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "version": "1.1"  } } |

## PATCH /catalogManagement/serviceSpecification/{ID}

This Uniform Contract operation is used to partially update the representation of a managed entity or a task.

Description :

* This operation partially updates a serviceSpecification.
* The resource represents a managed entity.
* The ID may be a string (or a string containing numbers).

Behavior :

* What status and exception codes are returned.
* Returns HTTP/1.1 status code 201 if the request was successful.
* Any other special return and/or exception codes.

Specify which attributes are patchable using the following table (to capture RO attributes)

|  |  |  |
| --- | --- | --- |
| Attribute name | Patchable | Rule |
| Id | No |  |
| lastUpdate | No |  |
| version | Y | Should be greater than the current one |
| lifecycleStatus | Y | Status should be valid |
| startDateTime | Y |  |
| endDateTime | Y | Should be greater than startDateTime |
| name | Y |  |
| description | Y |  |
| brand | Y |  |
| relatedParty | Y | It should exist |
| attachment | Y | It should exist |
| serviceSpecRelationship | Y | It should exist |
| requiredServiceSpecification | Y | It should exist |
| requiredResourceSpecification | Y | It should exist |
| serviceSpecCharacteristic | Y | It should exist |
| serviceSpecCharacteristicValue | Y |  |

Further document any rules that must be implemented when patching attributes.

|  |  |
| --- | --- |
| Rule name | Rule/Pre Condition/Side Effects/Post Conditons |
|  |  |

|  |
| --- |
| **REQUEST** |
| PATCH /catalogManagement/serviceSpecification/{22}  Content-type: application/json-patch+json  {  "op": "add",  "path": "/attachment",  "value": {  "id": "44",  "href": "http://serverlocation:port/documentManagement/attachment/44",  "type": "Video",  "url": "http://yyyyy"  } } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "22",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/22",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "lifecycleStatus": "Active",  "name": "iPhone 42",  "description": "Siri works on this iPhone",  "brand": "Apple",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "attachment": [  {  "id": "56",  "href": "http://serverlocation:port/documentManagment/attachment/56",  "type": "Picture",  "url": "http://xxxxx"  },  {  "id": "44",  "href": "http://serverlocation:port/documentManagment/attachment/44",  "type": "Video",  "url": "http://yyyyy"  }  ],  "relatedParty": [  {  "role": "Owner",  "id": "1234",  "href": "http ://serverLocation:port/partyManagement/partyRole/1234"  }  ],  "serviceSpecRelationship": [  {  "type": "dependency",  "id": "23",  "href": " http://serverlocation:port/catalogManagement/serviceSpecification/23", "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredServiceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredResourceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/resourceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "serviceSpecCharacteristic": [  {  "name": "Screen Size",  "description": "Screen size",  "valueType": "number",  "configurable": "false",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "serviceSpecCharacteristicValue": [  {  "valueType": "number",  "default": "true",  "value": "4.2",  "unitOfMeasure": "inches",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }, {  "name": "Colour",  "description": "Product colour",  "valueType": "string",  "configurable": "true",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "ServiceSpecCharacteristicValue": [  {  "valueType": "string",  "default": "true",  "value": "Black",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }, {  "valueType": "string",  "default": "false",  "value": "White",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }  ] } |

## POST /catalogManagement/category/{ID}

This Uniform Contract operation is used to create a managed entity or a task.

Description :

* This operation creates a category
* The resource represents a managed entity.
* The ID may be a string (or a string containing numbers).

Behavior :

* Standard POST behavior.
* Returns HTTP/1.1 status code 201 if the request was successful.
* Any other special return and/or exception codes.

ID Management :

The response provide the category id.

Specify the attributes required when an entity is created (and their default values if not):

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute name | Mandatory | Default | Rule |
| isRoot | N | true |  |
| name | Y |  |  |
| parentId | Y |  | If isRoot is set to “false”, parentId is mandatory |

* Further specify any rules on the creation of the entity

|  |  |
| --- | --- |
| Rule name | Rule |
|  |  |

|  |
| --- |
| **REQUEST** |
| POST catalogManagement/category  Content-type: application/json  {  "version": "1.0",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": " 41",  "isRoot": "false",  "name": "Cloud Services",  "description": "A category to hold all available cloud service offers" } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/category/42",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "version": "1.0",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "parentId": " 41",  "isRoot": "false",  "name": "Cloud Services",  "description": "A category to hold all available cloud service offers" } |

## POST /catalogManagement/serviceCandidate/{ID}

This Uniform Contract operation is used to create a managed entity or a task.

Description :

* This operation creates a serviceCandidate.
* The resource represents a managed entity.
* The ID may be a string (or a string containing numbers).

Behavior :

* Standard POST behavior.
* Returns HTTP/1.1 status code 201 if the request was successful.
* Any other special return and/or exception codes.

ID Management :

The response provides the serviceCandidate id.

Specify the attributes required when an entity is created (and their default values if not):

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute name | Mandatory | Default | Rule |
| name | Y |  |  |
| serviceSpecification | Y |  |  |

* Further specify any rules on the creation of the entity

|  |  |
| --- | --- |
| Rule name | Rule |
|  |  |

|  |
| --- |
| **REQUEST** |
| POST catalogManagement/serviceCandidate  Content-type: application/json  {  "name": "Virtual Storage Medium",  "version": "1.0",  "description": "Virtual Storage Medium",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "category": [  {  "id": "12",  "href": "http://serverlocation:port/catalogManagement/category/12",  "version": "2.0",  "name": "Cloud service"  }  ],  "serviceLevelAgreement": {  "id": "28",  "href": "http://serverlocation:port/slaManagement/serviceLevelAgreement/28",  "name": "Standard SLA"  },  "serviceSpecification": {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "version": "1.1"  } } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/serviceCandidate/42",  "name": "Virtual Storage Medium",  "version": "1.0",  "description": "Virtual Storage Medium",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "category": [  {  "id": "12",  "href": "http://serverlocation:port/catalogManagement/category/12",  "version": "2.0",  "name": "Cloud service"  }  ],  "serviceLevelAgreement": {  "id": "28",  "href": "http://serverlocation:port/slaManagement/serviceLevelAgreement/28",  "name": "Standard SLA"  },  "serviceSpecification": {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "version": "1.1"  } } |

## POST /catalogManagement/serviceSpecification/{ID}

This Uniform Contract operation is used to create a managed entity or a task.

Description :

* This operation creates a serviceSpecification.
* The resource represents a managed entity.
* The ID may be a string (or a string containing numbers).

Behavior :

* Standard POST behavior.
* Returns HTTP/1.1 status code 201 if the request was successful.
* Any other special return and/or exception codes.

ID Management :

The response provides the serviceSpecification id.

Specify the attributes required when an entity is created (and their default values if not):

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute name | Mandatory | Default | Rule |
| name | Y |  |  |
| serviceSpecCharacteristic | Y |  |  |
| serviceSpecCharacteristicValue | Y |  |  |

* Further specify any rules on the creation of the entity

|  |  |
| --- | --- |
| Rule name | Rule |
|  |  |

|  |
| --- |
| **REQUEST** |
| POST catalogManagement/serviceSpecification  Content-type: application/json  {  "version": "1.0",  "lifecycleStatus": "Active",  "name": "iPhone 42",  "description": "Siri works on this iPhone",  "brand": "Apple",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "attachment": [  {  "id": "56",  "href": "http://serverlocation:port/documentManagment/attachment/56",  "type": "Picture",  "url": "http://xxxxx"  }  ],  "relatedParty": [  {  "role": "Owner",  "id": "1234",  "href": "http ://serverLocation:port/partyManagement/partyRole/1234"  }  ],  "serviceSpecRelationship": [  {  "type": "dependency",  "id": "23",  "href": " http://serverlocation:port/catalogManagement/serviceSpecification/23", "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredServiceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredResourceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/resourceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "serviceSpecCharacteristic": [  {  "name": "Screen Size",  "description": "Screen size",  "valueType": "number",  "configurable": "false",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "serviceSpecCharacteristicValue": [  {  "valueType": "number",  "default": "true",  "value": "4.2",  "unitOfMeasure": "inches",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }, {  "name": "Colour",  "description": "Product colour",  "valueType": "string",  "configurable": "true",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "ServiceSpecCharacteristicValue": [  {  "valueType": "string",  "default": "true",  "value": "Black",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }, {  "valueType": "string",  "default": "false",  "value": "White",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }  ] } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "22",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/22",  "version": "1.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "lifecycleStatus": "Active",  "name": "iPhone 42",  "description": "Siri works on this iPhone",  "brand": "Apple",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  "attachment": [  {  "id": "56",  "href": "http://serverlocation:port/documentManagment/attachment/56",  "type": "Picture",  "url": "http://xxxxx"  }  ],  "relatedParty": [  {  "role": "Owner",  "id": "1234",  "href": "http ://serverLocation:port/partyManagement/partyRole/1234"  }  ],  "serviceSpecRelationship": [  {  "type": "dependency",  "id": "23",  "href": " http://serverlocation:port/catalogManagement/serviceSpecification/23", "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredServiceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/serviceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "requiredResourceSpecification": [  {  "id": "13",  "href": "http://serverlocation:port/catalogManagement/resourceSpecification/13",  "name": "specification 1",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ],  "serviceSpecCharacteristic": [  {  "name": "Screen Size",  "description": "Screen size",  "valueType": "number",  "configurable": "false",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "serviceSpecCharacteristicValue": [  {  "valueType": "number",  "default": "true",  "value": "4.2",  "unitOfMeasure": "inches",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }, {  "name": "Colour",  "description": "Product colour",  "valueType": "string",  "configurable": "true",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  },  "ServiceSpecCharacteristicValue": [  {  "valueType": "string",  "default": "true",  "value": "Black",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }, {  "valueType": "string",  "default": "false",  "value": "White",  "unitOfMeasure": "",  "valueFrom": "",  "valueTo": "",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": ""  }  }  ]  }  ] } |

## POST catalogManagement/importJob

ImportJob Tasks are created as resources.

The ImportJob is attached to the URL of the root resource where the content of the file specified by the ImportJob will be applied.

For example to apply the content of the import file located at ftp://ftp.myCatalog.com/catalog.json to the catalogManagement root.

|  |
| --- |
| **REQUEST** |
| POST ../partyManagement/importJob  Content-type: application/json  {  "url": "ftp://ftp.myCatalog.com/catalog.json"  } |
| **RESPONSE** |
| 201  Content-Type: application/json  Location: ../partyManagement/importJob/554  {  "id": "54",  "href": "http:/api/catalogManagement/importJob/54",  "status": "running",  "path": "catalogManagement/",  "content-type": "application/json",  "errorLog": "",  "creationDate": "2013-04-19T16:42:23-04:00",  "url": "ftp://ftp.myCatalog.com/catalog.json"   } |

## POST catalogManagement/exportJob

ExportJob Tasks are created as resources.

The ExportJob is attached to a specific resource acting as the root for the collection of resources to be streamed to a File.

An ExportJob can be attached to a specific Resource in the Catalog Management application for example:

* ../catalogManagement/exportJob

Export all the resources within catalogManagement subject to query and path assignments.

* ../catalogManagement/catalog/2/exportJob

Export all the resources within the catalog with “id” = 2 subject to query and path assignments.

For example:

|  |
| --- |
| **REQUEST** |
| POST ../catalogManagement/exportJob  Content-type: application/json  {  } |
| **RESPONSE** |
| 201  Content-Type: application/json  Location: ../catalogManagement/exportJob/54  {  "id": "54",  "href": "http:/api/catalogManagement/exportJob/54",  "status": "running",  "path": "catalogManagement",  "content-type": "application/json",  "errorLog": "",  "creationDate": "2013-04-19T16:42:23-04:00",  "completionDate": "",  "url": "ftp://ftp.catalogManagement.com/catalog/54"  } |

## DELETE API/{RESOURCE}/{ID}

No delete operations

This Uniform Contract operation is used to delete a managed entity or a task.

Description :

* Provide an overall description of the Operation
* Describe if the resource represents a managed entity or a task.
* Describe the structure of the identifier.

Behavior :

* What status and exception codes are returned.
* Returns HTTP/1.1 status code 200 if the request was successful.
* Any other special return and/or exception codes.

|  |
| --- |
| **REQUEST** |
| DELETE API/{RESOURCE}/{ID} |
| **RESPONSE** |
| 200 |

Example see TMF REST Design Guidelines.

# API NOTIFICATIOn TEMPLATES

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.

|  |
| --- |
| **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.

|  |
| --- |
| **REQUEST** |
| DELETE /api/hub/{id}  Accept: application/json |
| **RESPONSE** |
| 204 |

## publish {EventTYPE} POST /listener

Description :

Provide the Event description

Behavior :

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

|  |
| --- |
| **REQUEST** |
| POST /client/listener  Accept: application/json  {    "event": {  EVENT BODY   },  "eventType": "eventType" } |
| **RESPONSE** |
| 201  Content-Type: application/json |

Example see TMF REST Design Guidelines.

# lifecycle management Extensions to Catalog

In Lifecycle Management there is a requirement to distinguish between entities existing with different life cycle version numbers and accessible via different ACL mechanisms.

For example the same Product Offerings may exist in a Catalog but with different version numbers.

It may be possible for an administrator to see all the existing versions or for a partner to see only a subset of all the existing versions.

The entity version number is not dependent on the version number of the API. For example in PLM the same API (running at a specific version number) may be used to retrieve entities with different PLM version numbers.

In order to distinguish resources representing entities running with different version numbers and accessible though the same API version the following directive can be used /id:(version=x) and the version attribute is added to each entity.

|  |
| --- |
|  |
| {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/productOffering/42",  "version": "1.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "name": "Virtual Storage Medium",  "description": "Virtual Storage Medium",  "isBundle": "true",  "lifecycleStatus": "Active",  …..  } |

Note that the catalog resources in this case may have the same ID but may be distinguished by the inclusion of the version number in their ID i.e /42:(version=1.0), /42:(version=2.0).

In the following examples we will assume that two versions of the VirtualStorage Product Offer exist in the Product Catalog. The Inactive and Active versions respectively version 1.0 and version 2.0.

## Query all versioned catalog resources

Users with different roles may have access to different versions of the entities in the catalog. For example user A may have access to only the version 1.0 of the entities while user B may have access to version 1.0 and version 2.0.

Admin user of Catalog have access to all the versions of the resources while non admin users have by default access to only the latest version of the entities in the Catalog.

For example the following request on the admin endpoint will return all the versioned resources matching a specific ID.

|  |
| --- |
| **REQUEST** |
| GET api/admin/catalogManagement/productOffering/?id=VirtualStorage  Accept: application/json |
| **RESPONSE** |
| 200  Content-Type: application/json  [{  "id": " VirtualStorage ",  "href": "http://serverlocation:port/catalogManagement/productOffering/VirtualStorage",  "version": "1.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "name": "Virtual Storage Medium",  "description": "Virtual Storage Medium",  "isBundle": "true",  "lifecycleStatus": "InActive",  …..  },  {  "id": " VirtualStorage ",  "href": "http://serverlocation:port/catalogManagement/productOffering/ VirtualStorage ",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "name": "Virtual Storage Medium",  "description": "Virtual Storage Medium",  "isBundle": "true",  "lifecycleStatus": "Active",  …..  }  ] |

## Query a specific versioned CATALOG resource

In general a non admin API user only has visibility to the latest version number or visibility to a subset of versioned catalog resources.

It may be possible for an admin API user to retrieve a resource with a specific version number by using an ID and versioning filtering criteria.

|  |
| --- |
| **REQUEST** |
| GET api/admin/catalogManagement/productOffering/?id=VirtualStorage&version=1.0  Accept: application/json |
| **RESPONSE** |
| 200  Content-Type: application/json  [{  "id": "42",  "href": "http://serverlocation:port/catalogManagement/productOffering/42",  "version": "1.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "name": "Virtual Storage Medium",  "description": "Virtual Storage Medium",  "isBundle": "true",  "lifecycleStatus": "Active",  …..  ] |

## Query current version of a catalog resource

By default only the most current version is returned (for admin and non admin).

|  |
| --- |
| **REQUEST** |
| GET api/admin/catalogManagement/productOffering/VirtualStorage  Accept: application/json |
| **RESPONSE** |
| 200  Content-Type: application/json  {  "id": "42",  "href": "http://serverlocation:port/catalogManagement/productOffering/42",  "version": "2.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "name": "Virtual Storage Medium",  "description": "Virtual Storage Medium",  "isBundle": "true",  "lifecycleStatus": "Active",  …..  } |

## Create new version of a CATALOG resource

POST is used to create a new version of a catalog resource.

The constraint is that the version numbers for the resource having the same ID must differ.

|  |
| --- |
| **REQUEST** |
| POST catalogManagement/productOffering  Content-type: application/json  {  “id”: “VirtualStorage”,  "version": "3.0",  "name": "Virtual Storage Medium",  "description": "Virtual Storage Medium",  "isBundle": "true",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  …  } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "VirtualStorage",  "href": "http://serverlocation:port/catalogManagement/productOffering/42",  "version": "3.0",  "lastUpdate": "2013-04-19T16:42:23-04:00",  "name": "Virtual Storage Medium",  "description": "Virtual Storage Medium",  "isBundle": "true",  "lifecycleStatus": "Active",  "validFor": {  "startDateTime": "2013-04-19T16:42:23-04:00",  "endDateTime": "2013-06-19T00:00:00-04:00"  },  …  } |

## Modify an existing version of a CATalog resource

By default PATCH or PUT will be acting only on the latest version of a Catalog Resource. For example PATCH /…/productOffering/VirtualStorage will only update the VirtualStoage ProductOffering at Version 2.0 (which is the most current).

To update a specific version of an entity the (Version=X) directive is added to the ID (i.e /id:(version=x) .

Note that this capability is only available to API users having the proper authorizations to change the catalog entities with specific version numbers.

For example to change the VirtualStorage versioned at 1.0 we could use /productOffering/VirtualStorage(Version=1.0)

|  |
| --- |
| **REQUEST** |
| PATCH /catalogManagement/productOffering/VirtualStorage(Version=1.0)  Content-type: application/json-patch+json  {  "lifecycleStatus": "Active"   } |
| **RESPONSE** |
| 201  Content-Type: application/json  {  "id": "VirtualStorage",  "href": "http://serverlocation:port/catalogManagement/productOffering/VirtualStorage",  "version": "1.0",  "lifecycleStatus": "Active",  …..  } |

## Role based Access Control

The user presents their credentials for authentication

If the credentials are valid

1. The user is given access to the catalog

2. As defined by their role(s)

3. As defined by their access rights

4. As defined by the access type: CRUD, discover

5. As defined by the pre-defined filter

For example if they issue a get on a catalog that a party has no access they get an error response

Or if they try to modify an area of the catalog but do not have Write Access they get an error response

Normally we anticipate that the OAUTH2 or Open ID Connect are used as the authorization APIs and that ACL are establisher between authorized parties with regards to the content of the Catalog (i.e GET but also enable of update operations on specific entities).

## Release History

|  |  |  |  |
| --- | --- | --- | --- |
| **Release Number** | **Date** | **Release led by:** | **Description** |
| Release 1.0 | 04/15/2013 | Pierre Gauthier  TM Forum  [pgauthier@tmforum.org](mailto:pgauthier@tmforum.org) | First Release of Draft Version of the Document. |
| Release 1.1 |  |  | Updated for use in the Paris Spec Jam – and rebranded,. |