



## *Frameworkx Specification*

# Customer Management API REST Specification

**TMF629**  
**Release 14.5.0**  
**September 2014**

<b>Latest Update: Frameworkx Release 14.5</b>	<b>Member Evaluation</b>
<b>Version 1.2.0</b>	<b>IPR Mode: RAND</b>

## NOTICE

Copyright © TM Forum 2014. All Rights Reserved.

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

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

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

Direct inquiries to the TM Forum office:

240 Headquarters Plaza,  
East Tower – 10<sup>th</sup> 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)

## TABLE OF CONTENTS

NOTICE.....	2
Table of Contents.....	3
List of Tables.....	5
Introduction .....	6
SAMPLE USE CASES.....	8
RESOURCE MODEL.....	9
Managed Entity and Task Resource Models .....	9
Customer Resource.....	9
Customer Account Resource.....	14
Payment Mean Resource .....	21
Notification Resource Models .....	24
Customer Create Notification.....	24
Customer Delete Notification.....	25
Customer Update Notification.....	25
Customer Account Create Notification .....	26
Customer Account Delete Notification .....	27
Customer Account Update Notification.....	27
payment Mean Create Notification .....	29
payment Mean Delete Notification.....	29
payment Mean Update Notification .....	29
API OPERATION TEMPLATES .....	31
GET /customerManagement/customer/{ID}.....	31
PUT /customerManagement/customer/{ID}.....	37
PATCH API/customerManagement/customer/{ID} .....	41
POST /customerManagement/customer .....	51
DELETE /customerManagement/customer/{ID} .....	56
GET customerManagement/customerAccount/{ID}.....	56

PUT customerManagement/customerAccount/{ID} ..... 64

PATCH customerManagement/customerAccount/{ID} ..... 64

POST customerManagement/customerAccount/{ID} ..... 67

DELETE customerManagement/customerAccount/{ID} ..... 70

GET customerManagement/paymentMean/{ID} ..... 71

PUT customerManagement/paymentMean/{ID} ..... 72

PATCH customerManagement/paymentmean/{ID} ..... 72

POST customerManagement/paymentmean/{ID} ..... 74

DELETE customerManagement/paymentMean/{ID} ..... 76

API NOTIFICATION TEMPLATES ..... 78

    REGISTER LISTENER POST for customermanagement ..... 78

    UNREGISTER LISTENER DELETE customermanagment/customeraccount/hub/{id} ..... 79

Release History ..... 81

    Contributors to Document ..... 81

## LIST OF TABLES

N/A

## INTRODUCTION

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

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

Customer can be a person, an organization or another service provider who buys products from an enterprise. Customer management API allows management of identification and financial information about him.

Customer management API manages the following data resources:

- **Customer**
  - o Customer represents a person or organization that buys products and services from the enterprise or receives free offers or services. Customers can also be other service providers who resell the enterprises products, other service providers that lease the enterprise's resources for utilization by the other service provider's products and services, and so forth.
  - o Customer resource contains information about the customer. Main attributes are its identifier, name, status and validity, description, characteristics, contact medium, related customer account, related party, customer credit profile information
- **Customer Account**
  - o Customer account represents a financial entity. It records all customer accounting events (payment and invoices amounts)
  - o Customer account main attributes are its identifier, name, account type, status, description, credit limit, receivable balance, tax exemption, relationships, contacts, balance, payment plan and payment mean
- **Payment Mean**
  - o Payment mean represents how the customer will pay the invoice or payment

Customer management API performs the following operation on customer, customer account or payment mean

- Retrieval of a customer, customer account, payment mean or of a collection of them or depending on filter criteria
- Full update of a customer, customer account or payment mean
- Partial update of a customer, customer account or payment mean
- Creation of a customer, customer account or payment mean
- Deletion of a customer, customer account or payment mean
- Notification of events:
  - o Customer create
  - o Customer update
  - o Customer delete
  - o Customer Account create
  - o Customer Account update
  - o Customer Account delete
  - o Payment Mean create
  - o Payment Mean update
  - o Payment Mean delete

Notes:

- Business Interactions are put out of scope of this Customer API. They can be handled and retrieved through specific tasks (an example of task for order retrieval is detailed at the end of this document)
- For the moment, we assume that a customerAccount cannot be shared between multiple customers
- Payment plans may be created during ordering process: this case is not covered yet.

## SAMPLE USE CASES

Reader will find example of use cases using Customer management API in “Open Digital Business Scenarios and Use Cases” document.



## RESOURCE MODEL

### Managed Entity and Task Resource Models

#### CUSTOMER RESOURCE

The customer resource is the main entity to access to customer information.

This entity contains mainly financial information about the customer: touch point to customer account(s) and credit profiles.

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

```
{
  "id": "c1234",
  "href": "http://serverlocation:port/customerManagement/customer/c1234",
  "name": "DisplayName",
  "status": "Active",
  "description": "Description string",
  "validFor": {
    "startDateTime": "2013-06-19T04:00:00.0Z",
    "endDateTime": "2013-04-19T20:42:23.0Z"
  },
  "customerRank": "3",
  "relatedParty": {
    "id": "1",
    "href": "http://serverlocation:port/partyManagement/individual/1",
    "role": "customer",
    "name": "John Doe"
  },
  "characteristic": [
    {
      "name": "characteristicname1",
      "value": "characteristicvalue1"
    },
    {
      "name": "characteristicname2",
      "value": "characteristicvalue2"
    }
  ],
  "contactMedium": [
    {
      "type": "Email",
      "validFor": {
        "startDateTime": "2013-04-19T20:42:23.0Z"
      },
      "medium": {
        "emailAddress": "abc@tmforum.com"
      }
    },
    {
      "type": "PostalAddress",
```

```

    "validFor": {
      "startTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "city": "Wien",
      "country": "Austria",
      "postcode": "1020",
      "stateOrProvince": "Quebec",
      "street1": "Lassallestrasse7",
      "street2": ""
    }
  },
  {
    "type": "TelephoneNumber",
    "validFor": {
      "startTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "type": "mobile",
      "number": "+436641234567"
    }
  },
  {
    "preferred": true,
    "type": "TelephoneNumber",
    "validFor": {
      "startTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "type": "business",
      "number": "+436641234567"
    }
  }
],
"customerAccount": [
  {
    "id": "1",
    "href": "http://serverlocation:port/customerManagement/customerAccount/1",
    "name": "CustomerAccount1",
    "description": "CustomerAccountDesc1",
    "status": "Active"
  },
  {
    "id": "2",
    "href": "http://serverlocation:port/customerManagement/customerAccount/2",
    "name": "CustomerAccount2",
    "description": "CustomerAccountDesc2",
    "status": "Active"
  }
],
"customerCreditProfile": [
  {
    "creditProfileDate": "2013-04-19T20:42:23.0Z",
    "validFor": {
      "startTime": "2013-04-19T20:42:23.0Z",
      "endTime": "2013-06-19T04:00:00.0Z"
    }
  }
]

```

```

    },
    "creditRiskRating": 1,
    "creditScore": 1
  },
  {
    "creditProfileDate": "2013-04-19T20:42:23.0Z",
    "validFor": {
      "startDateTime": "2013-04-19T20:42:23.0Z",
      "endDateTime": "2013-06-19T04:00:00.0Z"
    },
    "creditRiskRating": 1,
    "creditScore": 1
  }
],
"paymentMean": [
  {
    "id": "45",
    "href": "http://serverlocation:port/customerManagement/paymentMean/45",
    "name": "my favourite payment mean"
  },
  {
    "id": "64",
    "href": "http://serverlocation:port/customerManagement/paymentMean/64",
    "name": "my credit card payment mean"
  }
]
}

```

### Field Descriptions :

**Characteristic:** describes the characteristics of the individual or the organization such as individual hobbies, center of interests.

Field	Description
<b>name</b>	Name of the characteristic
<b>value</b>	Value of the characteristic

**ContactMedium:** indicates the contact medium that could be used to contact the customer.

Field	Description
<b>preferred</b>	If true, indicates that is the preferred contact medium
<b>type</b>	Email address, telephone number, postal address
<b>validFor</b>	The time period that the contact medium is valid for

**Customer:** A person or organization that buys products and services from the enterprise or receives free offers or services. This is modeled as a Party playing the role of Customer. A Customer is a type of PartyRole.

Field	Description
-------	-------------

<b>customerRank</b>	Relative importance of this customer compared to other customers
<b>description</b>	Detailed description of the customer
<b>href</b>	Unique reference of the customer
<b>id</b>	Unique identifier of the customer
<b>name</b>	Displayable name
<b>status</b>	Used to track the lifecycle status, e.g. existing, prospective or former customers
<b>validFor</b>	To manage startDate and endDate of customer states

**CustomerAccount:** Is used to represent an account for the customer to manage the billing aspects.

Field	Description
<b>description</b>	Detailed description of the customer account
<b>href</b>	Unique reference of the customer account
<b>id</b>	Unique identifier of the customer account
<b>name</b>	Name of the customer account
<b>status</b>	The condition of the account, such as due, paid, in arrears, in collection

**CustomerCreditProfile:** Credit profile for the customer (containing credit scoring, ...). By default only the current credit profile is retrieved. It can be used as a list to give the customer credit profiles history, the first one in the list will be the current one.

Field	Description
<b>creditProfileDate</b>	The date the profile was established.
<b>creditRiskRating</b>	This is an integer whose value is used to rate the risk
<b>creditScore</b>	A measure of a person's or organization's creditworthiness calculated on the basis of a combination of factors such as their income and credit history.
<b>validFor</b>	The period for which the profile is valid.

**Medium:** describes the contact medium that could be used to contact the customer

Field	Description
<b>city</b>	City
<b>country</b>	Country
<b>emailAddress</b>	Full email address in standard format
<b>type</b>	Type of medium (fax, mobile phone...)
<b>number</b>	Phone number
<b>postcode</b>	Postcode
<b>stateOrProvince</b>	State or province
<b>street1</b>	Describes the street
<b>street2</b>	Complementary street description

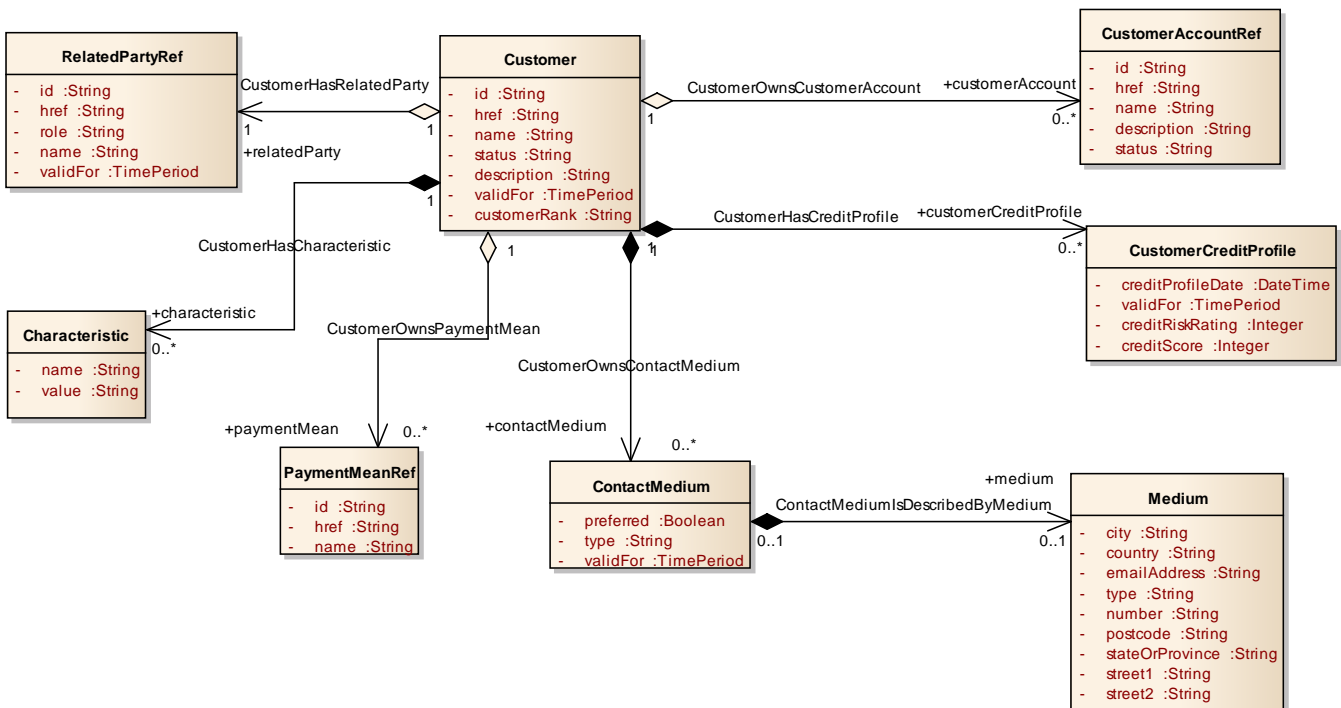
**PaymentMean:** defines a specific mean of payment (e.g. direct debit with all details associated).

Field	Description
<b>id</b>	Unique identifier of the payment mean
<b>href</b>	Reference of the payment mean
<b>name</b>	Name of the payment mean

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
role	Role of the related party
name	Name of the related party
validFor	The period for which the related party is valid

UML Model:



## CUSTOMER ACCOUNT RESOURCE

The customer account resource is used to represent an account for the customer to manage billing aspects.

Customer account can contain customer tax exemption, related accounts, contact information, customer relation, account balances and payment plans.

Below is a representation of the Customer Account resource in JSON format.

```
{
  "id": "ca1234",
  "href": "http://serverlocation:port/customerManagement/customerAccount/ca1234",
  "lastModified": "2014-09-24T18:27:37.059899Z",
  "name": "sampleaccount",
  "accountType": "Residential",
  "status": "in progress",
  "description": "Description string",
  "creditLimit": 1212121,
  "pin": "pin0",
  "receivableBalance": 52.3,
  "customerAccountTaxExemption": [
    {
      "issuingJurisdiction": "SampleJurisdiction",
      "certificateNumber": "CATaxExemption1",
      "reason": "Reason",
      "validFor": {
        "startDateTime": "2013-04-19T16:42:23.0Z",
        "endDateTime": "2013-06-19T00:00:00.0Z"
      }
    },
    {
      "issuingJurisdiction": "SampleJurisdiction",
      "certificateNumber": "CATaxExemption1",
      "reason": "Reason",
      "validFor": {
        "startDateTime": "2013-04-19T16:42:23.0Z",
        "endDateTime": "2013-06-19T00:00:00.0Z"
      }
    }
  ],
  "customerAccountRelationship": [
    {
      "relationshipType": "Type xx",
      "validFor": {
        "startDateTime": "2013-04-19T16:42:23.0Z",
        "endDateTime": "2013-06-19T00:00:00.0Z"
      },
      "customerAccount": [
        {
          "id": "1",
          "href":
"http://serverlocation:port/customerManagement/customerAccount/1",
          "name": "CustomerAccount1",

```

```
        "description": "CustomerAccountDesc1"
      }
    ]
  },
],
"contact": [
  {
    "contactType": "primary",
    "validFor": {
      "startDateTime": "2013-04-19T16:42:23.0Z",
      "endDateTime": "2013-06-19T00:00:00.0Z"
    },
    "contactName": "DisplayName",
    "partyRoleType": "CustomerAccountRepresentative",
    "contactMedium": [
      {
        "type": "Email",
        "medium": {
          "emailAddress": "abc@tmforum.com"
        }
      },
      {
        "type": "PostalAddress",
        "medium": {
          "city": "Wien",
          "country": "Austria",
          "postcode": "1020",
          "stateOrProvince": "Quebec",
          "street1": "Lassallestrasse7"
        }
      },
      {
        "type": "TelephoneNumber",
        "medium": {
          "type": "mobile",
          "number": "+436641234567"
        }
      },
      {
        "preferred": true,
        "type": "TelephoneNumber",
        "medium": {
          "type": "business",
          "number": "+436641234567"
        }
      }
    ],
    "relatedParty": {
      "id": "1",
      "href": "http://serverlocation:port/partyManagement/individual/1",
      "role": "customer"
    }
  }
],
"customer": {
  "id": "1",
```

```
    "href": "http://serverlocation:port/customerManagement/customer/1",
    "name": "Customer1",
    "description": "CustomerDesc1"
  },
  "customerAccountBalance": [
    {
      "type": "ReceivableBalance",
      "amount": 52.3,
      "validFor": {
        "startDateTime": "2013-04-19T16:42:23.0Z",
        "endDateTime": "2013-06-19T00:00:00.0Z"
      },
      "status": "Due"
    },
    {
      "type": "DepositBalance",
      "amount": 52.3,
      "validFor": {
        "startDateTime": "2013-04-19T16:42:23.0Z",
        "endDateTime": "2013-06-19T00:00:00.0Z"
      },
      "status": "Paid"
    }
  ],
  "paymentPlan": [
    {
      "status": "Effective",
      "type": "Type1",
      "priority": 1,
      "amount": 15.3,
      "paymentFrequency": "monthly",
      "numberOfPayments": 4,
      "validFor": {
        "startDateTime": "2013-04-19T16:42:23.0Z",
        "endDateTime": "2013-06-19T00:00:00.0Z"
      }
    },
    {
      "status": "Ineffective",
      "type": "Type2",
      "priority": 2,
      "amount": 20,
      "paymentFrequency": "monthly",
      "numberOfPayments": 2,
      "validFor": {
        "startDateTime": "2013-04-19T16:42:23.0Z",
        "endDateTime": "2013-06-19T00:00:00.0Z"
      },
      "paymentMean": {
        "id": "70",
        "href": "http://serverlocation:port/customerManagement/paymentMean/70"
      }
    }
  ]
}
```



### Field Descriptions :

Contact: An individual or an organization used as a contact point for a CustomerAccount and accessed via some contact medium.

Field	Description
<b>contactType</b>	Type of contact: Primary, secondary...
<b>validFor</b>	Validity period of that contact
<b>contactName</b>	A displayable name for the contact
<b>partyRoleType</b>	Identifies what kind of partyRole type is linked to the contact (for instance customer account manager)

ContactMedium: indicates the contact medium that could be used to contact the customer.

Field	Description
<b>preferred</b>	If true, indicates that is the preferred contact medium
<b>type</b>	Email address, telephone number, postal address
<b>validFor</b>	The time period that the contact medium is valid for

Customer: A person or organization that buys products and services from the enterprise or receives free offers or services. This is modeled as a Party playing the role of Customer. A Customer is a type of PartyRole.

Field	Description
<b>customerRank</b>	Relative importance of this customer compared to other customers
<b>description</b>	Detailed description of the customer
<b>href</b>	Unique reference of the customer
<b>id</b>	Unique identifier of the customer
<b>name</b>	Displayable name
<b>status</b>	Used to track the lifecycle status, e.g. existing, prospective or former customers
<b>validFor</b>	To manage startDate and endDate of customer states

CustomerAccount: Is used to represent an account for the customer to manage the billing aspects.

Field	Description
<b>accountType</b>	A categorization of an account, such as individual, joint, and so forth, whose instances share some of the same characteristics.
<b>creditLimit</b>	The maximum amount of money that may be charged on an account.
<b>description</b>	Detailed description of the customer account
<b>href</b>	Unique reference of the customer account
<b>id</b>	Unique identifier of the customer account
<b>lastModified</b>	Date of last modification of customer account
<b>name</b>	Name of the customer account
<b>pin</b>	A multidigit personal identification number that is used
<b>receivableBalance</b>	Overall receivable balance for the customer account
<b>status</b>	The condition of the account, such as due, paid, in arrears, in collection

CustomerAccountRelationship: Significant connection between customer accounts

Field	Description
relationshipType	Type of relationship
validFor	Validity period of that relationship

CustomerAccountTaxExemption: Proof of freedom from taxes imposed by a taxing jurisdiction.

Field	Description
certificateNumber	Identifier of a document that shows proof of exemption from taxes for the taxing jurisdiction
issuingJurisdiction	Name of the taxing jurisdiction for which taxes are exempt.
reason	Reason of the tax exemption
validFor	Period for which the exemption is valid

Medium: describes the contact medium that could be used to contact the customer

Field	Description
city	City
country	Country
emailAddress	Full email address in standard format
type	Type of medium (fax, mobile phone...)
number	Phone number
postcode	Postcode
stateOrProvince	State or province
street1	Describes the street
street2	Complementary street description

PaymentMean: defines a specific mean of payment (e.g. direct debit with all details associated).

Field	Description
id	Unique identifier of the payment mean
href	Reference of the payment mean
name	Name of the payment mean

PaymentPlan: Defines a plan for payment (when a customer wants to spread his payments)

Field	Description
amount	Amount paid.
numberOfPayments	Number of payments used to spread the global payment.
paymentFrequency	Monthly, Bimonthly, ...
priority	Priority of the payment plan
status	Status of the payment plan (effective, ineffective).
type	Type of payment plan
validFor	Validity period of the payment plan.

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
role	Role of the related party
name	Name of the related party
validFor	The period for which the related party is valid

CustomerAccountBalance: Balances linked to the customer account.

Field	Description
amount	Balance amount
status	Due, Paid, ...
type	Deposit balance, Disputed balance, Loyalty balance, Receivable balance
validFor	Balance validity period

NOTE:

- Customer credit profile is not a resource anymore, embedded in customer.
- The customer tax exemption is not a resource anymore, embedded in customerAccount.
- Each Payment plan has to be linked to a payment method -> paymentMean

UML model:



## PAYMENT MEAN RESOURCE

The payment mean resource is used to define a specific mean of payment (e.g. direct debit with all associated details).

Below are representations of the PaymentMean resource in JSON format.

```
{
  "id": "45",
  "href": "http://serverlocation:port/customerManagement/paymentMean/45",
  "name": "My favourite payment mean",
  "validFor": {
    "startDateTime": "2013-04-19T16:42:23.0Z",
    "endDateTime": "2013-06-19T00:00:00.0Z"
  },
  "type": "BankAccountDebit",
  "bankAccount": {
    "BIC": "PSSTFRPPPAR",
    "domiciliation": "LaBanquePostale-75900ParixCedex15",
    "IBAN": "FR4620061009010835927F33098",
    "accountHolder": "Mr.GustaveFlaubert"
  },
  "relatedParty": {
    "id": "1",
    "href": "http://serverlocation:port/partyManagement/individual/1",
    "role": "customer",
    "name": "Gustave Flaubert"
  }
}
```

```
{
  "id": "70",
  "href": "http://serverlocation:port/customerManagement/paymentMean/70",
  "name": "My credit card payment mean",
  "validFor": {
    "startDateTime": "2013-04-19T16:42:23.0Z"
  },
  "type": "CreditCard",
  "creditCard": {
    "creditCardHolder": "Mr.GustaveFlaubert",
    "type": "Visa",
    "number": "xxxxx1234",
    "expirationDate": "2014-09-30T00:00:00.0Z"
  },
  "relatedParty": {
    "id": "1",
    "href": "http://serverlocation:port/partyManagement/individual/1",
    "role": "customer",
    "name": "Gustave Flaubert"
  }
}
```

### Field Descriptions :

BankAccount: describes the information of a bank account.

Field	Description
iban	Is an international format to describe a bank account
bic	Identifier of the bank according to its international code
domiciliation accountHolder	Address of the bank agency which manages the bank account

CreditCard: describes the credit card information.

Field	Description
creditCardHolder	Name of the credit card holder
type	Credit card type (Mastercard, Visa,...)
number	Card number
expirationDate	Expiration date of the card

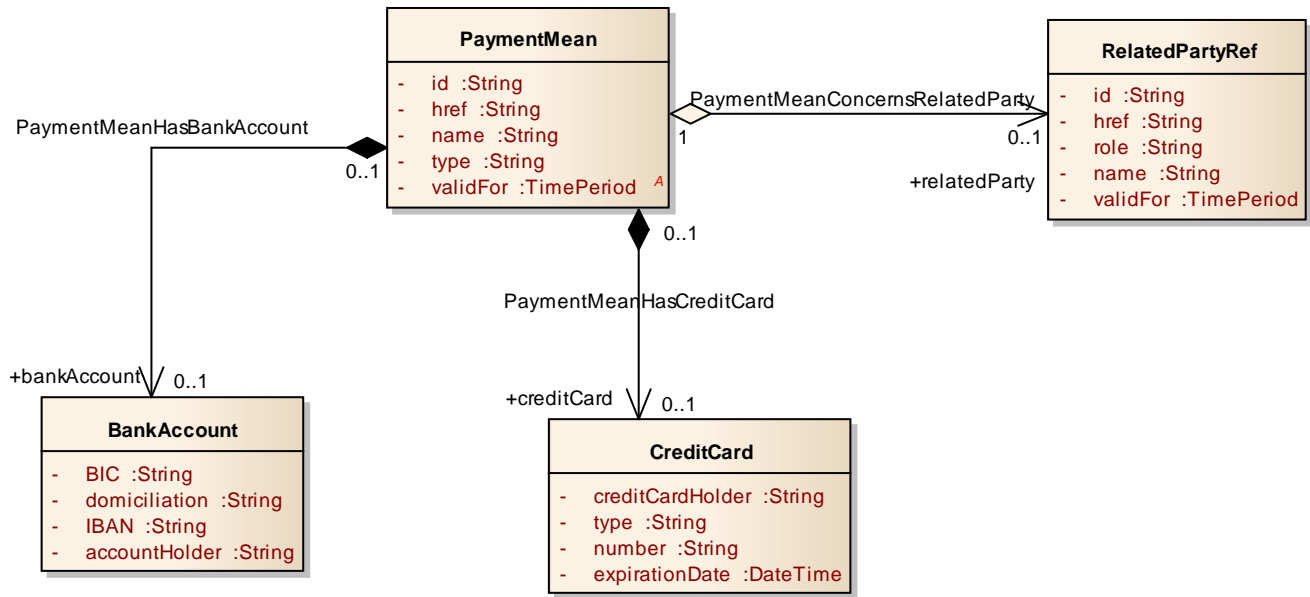
PaymentMean: defines a specific mean of payment (e.g. direct debit with all details associated).

Field	Description
id	Unique identifier for the payment mean
href	Reference of the payment mean
name	Name of the payment mean
validFor	Payment mean validity period
type	Type of the payment mean

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
role	Role of the related party
name	Name of the related party
validFor	The period for which the related party is valid

UML model:

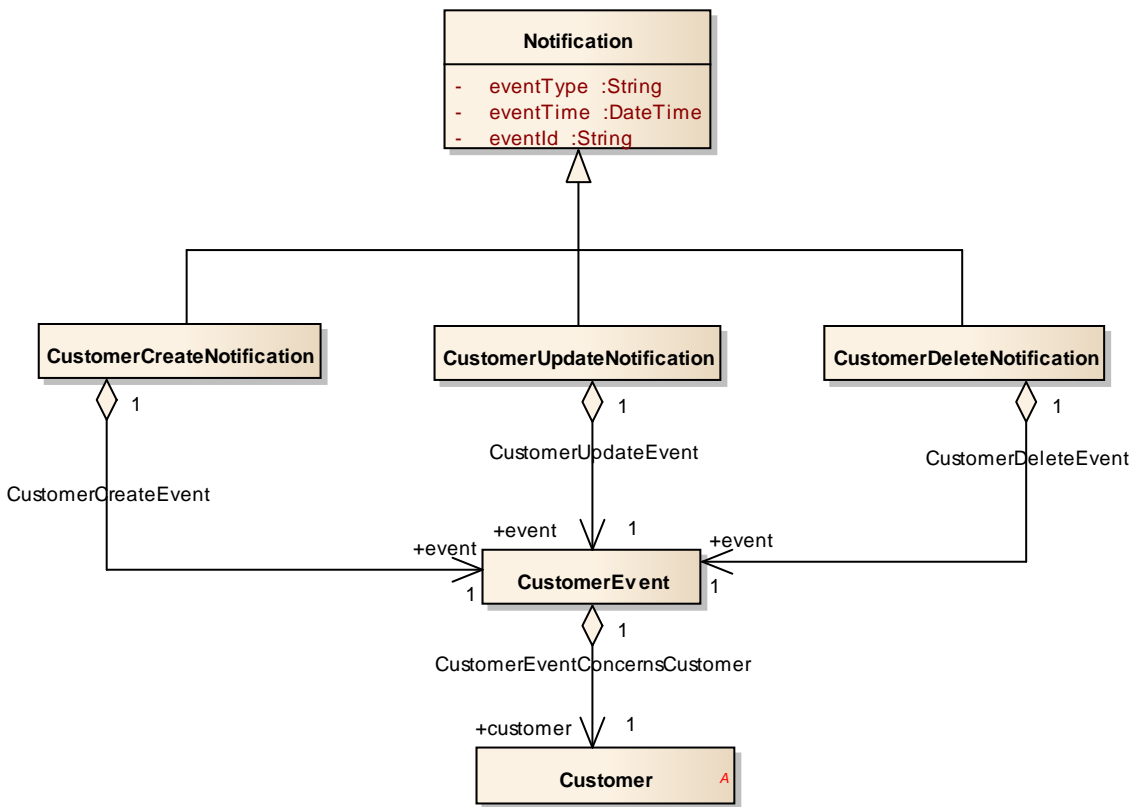


Notification Resource Models

Nine notification event types are defined:

1. CustomerCreateNotification
2. CustomerDeleteNotification
3. CustomerUpdateNotification
4. CustomerAccountCreateNotification
5. CustomerAccountDeleteNotification
6. CustomerAccountUpdateNotification
7. PaymentMeanCreateNotification
8. PaymentMeanDeleteNotification
9. PaymentMeanUpdateNotification

Customer Notification UML:



CUSTOMER CREATE NOTIFICATION

```

{
    "eventType": "CustomerCreateNotification",
}
    
```



```
"eventTime": "2014-09-27T05:46:25.0Z",
"eventId": "1562231",
"event":
{
  "customer":
  {
    "id": "c1234",
    Following a whole representation of the Customer with all its attributes
    See Customer Resource.
  }
}
```

---

## CUSTOMER DELETE NOTIFICATION

```
{
  "eventType": "CustomerDeleteNotification",
  "eventTime": "2014-09-27T05:46:25.0Z",
  "eventId": "114445231",
  "event":
  {
    "customer":
    {
      "id": "c1234",
      Following a whole representation of the Customer with all its attributes
      See Customer Resource.
    }
  }
}
```

---

## CUSTOMER UPDATE NOTIFICATION

```
{
  "eventType": "CustomerUpdateNotification",
  "eventTime": "2014-09-27T05:46:25.0Z",
  "eventId": "1144231",
```

```
"event":
```

```
{
```

```
  "customer":
```

```
  {
```

```
    "id": "c1234",
```

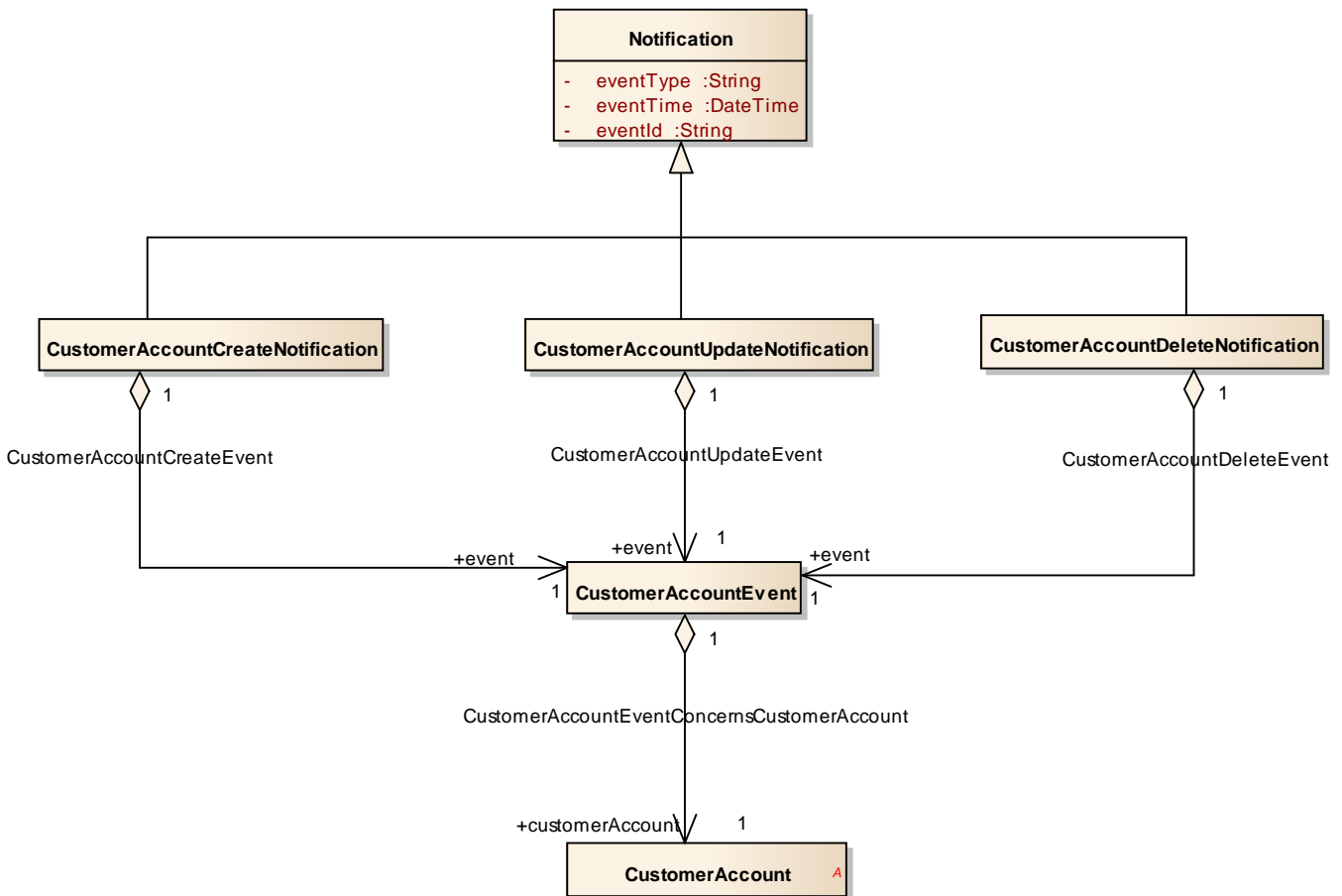
Following a whole representation of the Customer with all its attributes  
See Customer Resource.

```
  }
```

```
}
```

```
}
```

Customer Account Notification UML:



CUSTOMER ACCOUNT CREATE NOTIFICATION

```
{
  "eventType": "CustomerAccountCreateNotification",
  "eventTime": "2014-09-27T05:46:25.0Z",
  "eventId": "1134231",
  "event":
  {
    "customerAccount":
    {
      "id": "ca1234",
      Following a whole representation of the Customer Account with all its attributes
      See Customer Account Resource.
    }
  }
}
```

---

## CUSTOMER ACCOUNT DELETE NOTIFICATION

```
{
  "eventType": "CustomerAccountDeleteNotification",
  "eventTime": "2014-09-27T05:46:25.0Z",
  "eventId": "1134231",
  "event":
  {
    "customerAccount":
    {
      "id": "ca1234",
      Following a whole representation of the Customer Account with all its attributes
      See Customer Account Resource.
    }
  }
}
```

---

## CUSTOMER ACCOUNT UPDATE NOTIFICATION

```
{
  "eventType": "CustomerAccountUpdateNotification",
```

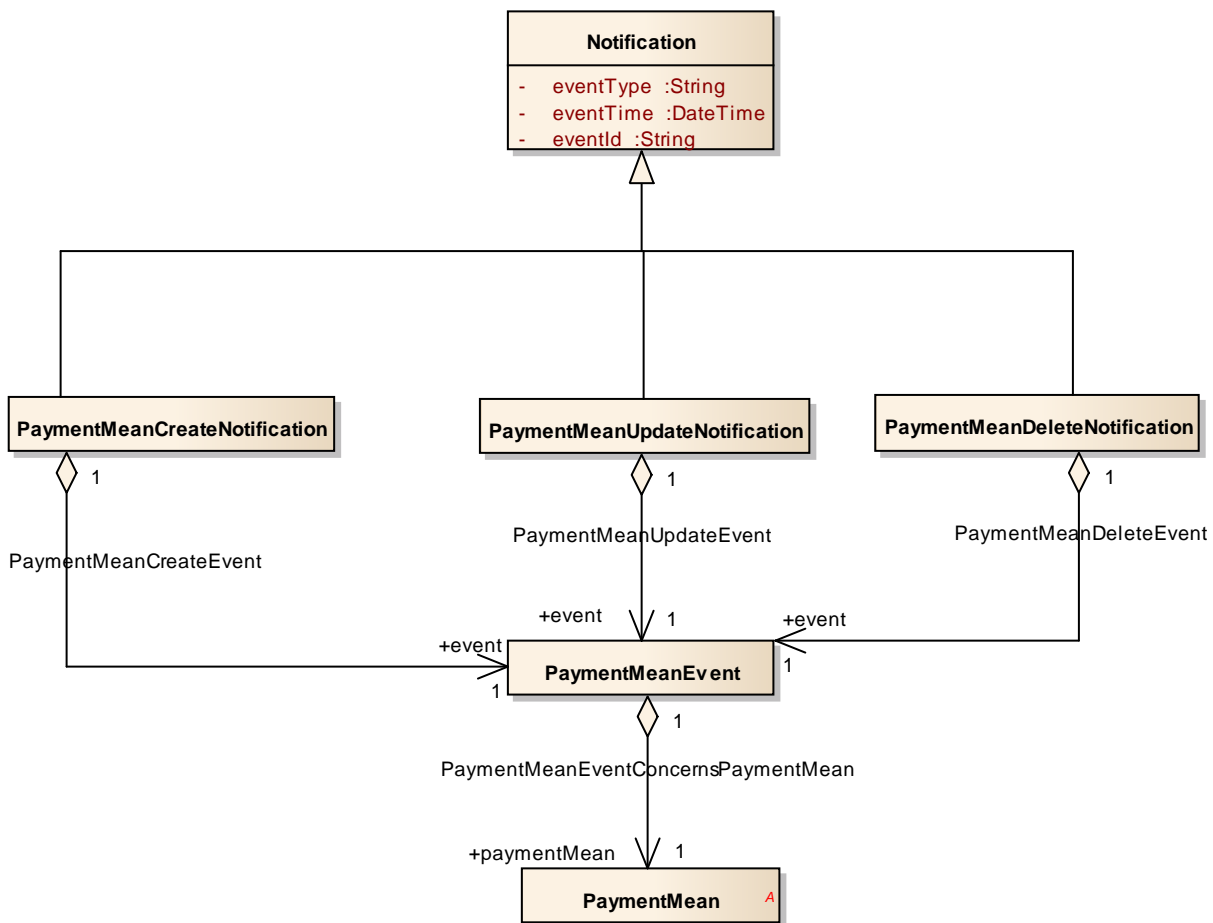
```

"eventTime": "2014-09-27T05:46:25.0Z",
"eventId": "33211",
"event":
{
  "customerAccount":
  {
    "id": "ca1234",

    Following a whole representation of the Customer Account with all its attributes
    See Customer Account Resource.

  }
}
}
    
```

Payment Mean Notification UML:



---

## PAYMENT MEAN CREATE NOTIFICATION

```
{
  "eventType": "PaymentMeanCreateNotification",
  "eventTime": "2014-09-27T05:46:25.0Z",
  "eventId": "1134231",
  "event":
  {
    "paymentMean":
    {
      "id": "11341",
      Following a whole representation of the Payment Mean with all its attributes
      See Payment Mean Resource.
    }
  }
}
```

---

## PAYMENT MEAN DELETE NOTIFICATION

```
{
  "eventType": "PaymentMeanDeleteNotification",
  "eventTime": "2014-09-27T05:46:25.0Z",
  "eventId": "11231",
  "event":
  {
    "paymentMean":
    {
      "id": "11341",
      Following a whole representation of the Payment Mean with all its attributes
      See Payment Mean Resource.
    }
  }
}
```

---

## PAYMENT MEAN UPDATE NOTIFICATION

```
{
```

```
"eventType": "PaymentMeanUpdateNotification",
```

```
"eventTime": "2014-09-27T05:46:25.0Z",
```

```
"eventId": "33211",
```

```
"event":
```

```
{
```

```
  "paymentMean":
```

```
  {
```

```
    "id": "11341",
```

Following a whole representation of the Payment Mean with all its attributes  
See Payment Mean Resource.

```
  }
```

```
}
```

```
}
```

## 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 /customerManagement/customer/{ID}

Note that collections can be retrieved via GET /customerManagement/customer without ID. In that case an array of customers is returned.

#### Description:

- This operation retrieves collection of customers (array) or only one customer when specifying the ID in the URI.
- Filtering is enabled on all attributes
- Attribute selection is enabled
- The resource represents one managed entity or a collection depending on the query pattern
- The identifier is a string that can consist of numbers, not necessarily alphanumeric

#### Behavior:

- Return Status Codes
  - 200 OK – the request was successful (includes criteria where the filtering condition didn't match to any records)
  - 400 Bad Request – Error
  - 404 If no record was found for the supplied criteria
  - 500 The server encountered an unexpected condition which prevented it from fulfilling the request

#### REQUEST

GET /customerManagement/customer/{ID}/?{fields=attributes}&{filtering expression}  
Accept: application/json

#### RESPONSE

Status:200  
Content-Type: application/json

```
{
  "id": "c1234",
  "href": "http://serverlocation:port/customerManagement/customer/c1234",
  "name": "DisplayName",
  "status": "Active",
  "description": "Description string",
  "validFor": {
    "startDateTime": "2013-06-19T04:00:00.0Z",
    "endDateTime": "2013-04-19T20:42:23.0Z"
  },
  "customerRank": "3",
```



```
"relatedParty": {
  "id": "1",
  "href": "http://serverlocation:port/partyManagement/individual/1",
  "role": "customer",
  "name": "John Doe"
},
"characteristic": [
  {
    "name": "characteristicname1",
    "value": "characteristicvalue1"
  },
  {
    "name": "characteristicname2",
    "value": "characteristicvalue2"
  }
],
"contactMedium": [
  {
    "type": "Email",
    "validFor": {
      "startTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "emailAddress": "abc@tmforum.com"
    }
  },
  {
    "type": "PostalAddress",
    "validFor": {
      "startTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "city": "Wien",
      "country": "Austria",
      "postcode": "1020",
      "stateOrProvince": "Quebec",
      "street1": "Lassallestrasse7",
```

```
        "street2": ""
    }
},
{
    "type": "TelephoneNumber",
    "validFor": {
        "startDateTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
        "type": "mobile",
        "number": "+436641234567"
    }
},
{
    "preferred": true,
    "type": "TelephoneNumber",
    "validFor": {
        "startDateTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
        "type": "business",
        "number": "+436641234567"
    }
}
],
"customerAccount": [
    {
        "id": "1",
        "href": "http://serverlocation:port/customerManagement/customerAccount/1",
        "name": "CustomerAccount1",
        "description": "CustomerAccountDesc1",
        "accountStatus": "Active"
    },
    {
        "id": "2",
        "href": "http://serverlocation:port/customerManagement/customerAccount/2",
        "name": "CustomerAccount2",
```

```
        "description": "CustomerAccountDesc2",
        "accountStatus": "Active"
    }
],
"customerCreditProfile": [
    {
        "creditProfileDate": "2013-04-19T20:42:23.0Z",
        "validFor": {
            "startDateTime": "2013-04-19T20:42:23.0Z",
            "endDateTime": "2013-06-19T04:00:00.0Z"
        },
        "creditRiskRating": 1,
        "creditScore": 1
    },
    {
        "creditProfileDate": "2013-04-19T20:42:23.0Z",
        "validFor": {
            "startDateTime": "2013-04-19T20:42:23.0Z",
            "endDateTime": "2013-06-19T04:00:00.0Z"
        },
        "creditRiskRating": 1,
        "creditScore": 1
    }
],
"paymentMean": [
    {
        "id": "45",
        "href": "http://serverlocation:port/customerManagement/paymentMean/45",
        "name": "my favourite payment mean"
    },
    {
        "id": "64",
        "href": "http://serverlocation:port/customerManagement/paymentMean/64",
        "name": "my credit card payment mean"
    }
]
}
```

If we need to retrieve only limited set of attributes and also filter the records based on certain filters then the below query pattern:

REQUEST
GET /customerManagement/customer/c1234?fields=id,name,status,customerAccount Accept: application/json
RESPONSE
Status: 200 Content-Type: application/json <pre>{   "id": "c1234",   "href": "http://serverlocation:port/customerManagement/customer/c1234",   "name": "DisplayName",   "status": "Active",   "customerAccount": [     {       "id": "1",       "href": "http://serverlocation:port/customerManagement/customerAccount/1",       "name": "CustomerAccount1",       "description": "CustomerAccountDesc1",       "accountStatus": "Active"     },     {       "id": "2",       "href": "http://serverlocation:port/customerManagement/customerAccount/2",       "name": "CustomerAccount2",       "description": "CustomerAccountDesc2",       "accountStatus": "Active"     }   ] }</pre>

**REQUEST**

GET /customerManagement/customer?fields=id,name ,validFor&status=Active&validFor.startDateTime.gt=2013-05-

05

Accept: application/json

**RESPONSE**

Status: 200

Content-Type: application/json

```
[
  {
    "id": "c1234",
    "name": "DisplayName",
    "validFor": {
      "startDateTime": "2013-06-06T16:42:23.0Z",
      "endDateTime": ""
    }
  },
  {
    "id": "c1234",
    "name": "DisplayName",
    "validFor": {
      "startDateTime": "2013-08-09T16:42:23.0Z",
      "endDateTime": ""
    }
  }
]
```

**PUT /customerManagement/customer/{ID}**

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

**Description:**

- This operation is used to completely update a customer
- The resource represents a managed entity.
- The identifier is a string that can consist of numbers, not necessarily alphanumeric

**Behavior:**

- Return Status Codes:
  - 200 Customer data modified successfully
  - 400 Bad Request Error
  - 404 If no record was found for the supplied criteria

- 500 The server encountered an unexpected condition which prevented it from fulfilling the request

The below query pattern can be used to modify all the data for a single customer, if an attribute is not provided, the value of the attribute is deleted:

REQUEST
<pre>PUT /customerManagement/customer/{ID} Content-type: application/json {   "id": "c1234",   "href": "http://serverlocation:port/customerManagement/customer/c1234",   "name": "DisplayName",   "status": "Active",   "description": "Description string",   "validFor": {     "startDateTime": "2013-06-19T04:00:00.0Z",     "endDateTime": "2013-04-19T20:42:23.0Z"   }, }, "customerRank": "3", "relatedParty": {   "id": "1",   "href": "http://serverlocation:port/partyManagement/individual/1",   "role": "customer",   "name": "John Doe" }, "characteristic": [   {     "name": "characteristicname1",     "value": "characteristicvalue1"   },   {     "name": "characteristicname2",     "value": "characteristicvalue2"   } ], "contactMedium": [   {</pre>

```
    "type": "Email",
    "validFor": {
      "startDateTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "emailAddress": "abc@tmforum.com"
    }
  },
  {
    "type": "PostalAddress",
    "validFor": {
      "startDateTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "city": "Wien",
      "country": "Austria",
      "postcode": "1020",
      "stateOrProvince": "Quebec",
      "street1": "Lassallestrasse7",
      "street2": ""
    }
  },
  {
    "type": "TelephoneNumber",
    "validFor": {
      "startDateTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "type": "mobile",
      "number": "+436641234567"
    }
  },
  {
    "preferred": true,
    "type": "TelephoneNumber",
    "validFor": {
      "startDateTime": "2013-04-19T20:42:23.0Z"
    }
  }
}
```

```
    },
    "medium": {
      "type": "business",
      "number": "+436641234567"
    }
  }
],
"customerAccount": [
  {
    "id": "1",
    "href": "http://serverlocation:port/customerManagement/customerAccount/1",
    "name": "CustomerAccount1",
    "description": "CustomerAccountDesc1",
    "accountStatus": "Active"
  },
  {
    "id": "2",
    "href": "http://serverlocation:port/customerManagement/customerAccount/2",
    "name": "CustomerAccount2",
    "description": "CustomerAccountDesc2",
    "accountStatus": "Active"
  }
],
"customerCreditProfile": [
  {
    "creditProfileDate": "2013-04-19T20:42:23.0Z",
    "validFor": {
      "startDateTime": "2013-04-19T20:42:23.0Z",
      "endDateTime": "2013-06-19T04:00:00.0Z"
    },
    "creditRiskRating": 1,
    "creditScore": 1
  },
  {
    "creditProfileDate": "2013-04-19T20:42:23.0Z",
    "validFor": {
      "startDateTime": "2013-04-19T20:42:23.0Z",
```



```
        "endTime": "2013-06-19T04:00:00.0Z"
      },
      "creditRiskRating": 1,
      "creditScore": 1
    }
  ],
  "paymentMean": [
    {
      "id": "45",
      "href": "http://serverlocation:port/customerManagement/paymentMean/45",
      "name": "my favourite payment mean"
    },
    {
      "id": "64",
      "href": "http://serverlocation:port/customerManagement/paymentMean/64",
      "name": "my credit card payment mean"
    }
  ]
}
```

**RESPONSE**

Status: 200  
Content-Type: application/json

**PATCH API/customerManagement/customer/{ID}**

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

**Description:**

- Patch operation can be used partially update one or more customers
- The resource represents a managed entity or a collection
- The identifier is a string that can consist of numbers, not necessarily alphanumeric

**Behavior:**

- Return Status Codes:
  - 200 – Customer(s) data modified successfully
  - 400 – Bad Request Error

- 404 - If no record was found for the supplied criteria
- 500 - The server encountered an unexpected condition which prevented it from fulfilling the request

## Patchable attributes:

Attribute name	Patchable	Rule
id	N	
name	Y	
status	Y	
description	Y	
validFor	Y	
customerRank	Y	
relatedParty	Y	
characteristic	Y	
contactMedium	Y	
customerAccount	Y	
customerCreditProfile	Y	
paymentMean	Y	

## Rules:

Rule name	Rule/Pre Condition/Side Effects/Post Conditions
party	id is mandatory
characteristic	name and value are mandatory

contactMedium	type and medium are mandatory
customerAccount	id, name and accountStatus are mandatory
customerCreditProfile	creditProfileDate and validFor are mandatory
paymentMean	Id, href are mandatory

Update of single attributes. In the following the name and the customerRank are updated. The request just consists of the two attributes with its new values. The response includes all attributes.

<b>REQUEST</b>
<pre> PATCH /customerManagement/customer/c1234 Content-type: application/json {   "name": "New Name to display",   "customerRank": 780 } </pre>
<b>RESPONSE</b>
<pre> Status: 200 Content-Type: application/json {   "id": "c1234",   "href": "http://serverlocation:port/customerManagement/customer/c1234",   "name": "New Name to display",   "status": "Active",   "description": "Description string",   "validFor": {     "startDateTime": "2013-06-19T04:00:00.0Z",     "endDateTime": "2013-04-19T20:42:23.0Z"   },   "customerRank": "780",   "relatedParty": {     "id": "1",     "href": "http://serverlocation:port/partyManagement/individual/1", </pre>

```
    "role": "customer",
    "name": "John Doe"
  },
  "characteristic": [
    {
      "name": "characteristicname1",
      "value": "characteristicvalue1"
    },
    {
      "name": "characteristicname2",
      "value": "characteristicvalue2"
    }
  ],
  "contactMedium": [
    {
      "type": "Email",
      "validFor": {
        "startDateTime": "2013-04-19T20:42:23.0Z"
      },
      "medium": {
        "emailAddress": "abc@tmforum.com"
      }
    },
    {
      "type": "PostalAddress",
      "validFor": {
        "startDateTime": "2013-04-19T20:42:23.0Z"
      },
      "medium": {
        "city": "Wien",
        "country": "Austria",
        "postcode": "1020",
        "stateOrProvince": "Quebec",
        "street1": "Lassallestrasse7",
        "street2": ""
      }
    }
  ],
}
```

```
{
  "type": "TelephoneNumber",
  "validFor": {
    "startDateTime": "2013-04-19T20:42:23.0Z"
  },
  "medium": {
    "type": "mobile",
    "number": "+436641234567"
  }
},
{
  "preferred": true,
  "type": "TelephoneNumber",
  "validFor": {
    "startDateTime": "2013-04-19T20:42:23.0Z"
  },
  "medium": {
    "type": "business",
    "number": "+436641234567"
  }
}
],
"customerAccount": [
  {
    "id": "1",
    "href": "http://serverlocation:port/customerManagement/customerAccount/1",
    "name": "CustomerAccount1",
    "description": "CustomerAccountDesc1",
    "accountStatus": "Active"
  },
  {
    "id": "2",
    "href": "http://serverlocation:port/customerManagement/customerAccount/2",
    "name": "CustomerAccount2",
    "description": "CustomerAccountDesc2",
    "accountStatus": "Active"
  }
]
```

```
],
  "customerCreditProfile": [
    {
      "creditProfileDate": "2013-04-19T20:42:23.0Z",
      "validFor": {
        "startDateTime": "2013-04-19T20:42:23.0Z",
        "endDateTime": "2013-06-19T04:00:00.0Z"
      },
      "creditRiskRating": 1,
      "creditScore": 1
    },
    {
      "creditProfileDate": "2013-04-19T20:42:23.0Z",
      "validFor": {
        "startDateTime": "2013-04-19T20:42:23.0Z",
        "endDateTime": "2013-06-19T04:00:00.0Z"
      },
      "creditRiskRating": 1,
      "creditScore": 1
    }
  ],
  "paymentMean": [
    {
      "id": "45",
      "href": "http://serverlocation:port/customerManagement/paymentMean/45",
      "name": "my favourite payment mean"
    },
    {
      "id": "64",
      "href": "http://serverlocation:port/customerManagement/paymentMean/64",
      "name": "my credit card payment mean"
    }
  ]
}
```

PATCH update for an array

**REQUEST**

PATCH /customerManagement/customer/c1234

Content-type: application/json-patch+json

```
{
  "op": "add",
  "path": "/customer/contactMedium",
  "value": {
    "contactMedium": [
      {
        "type": "PostalAddress",
        "medium": {
          "street1": "Lassallestrasse7",
          "street2": "",
          "postcode": "1020",
          "stateOrProvince": "Quebec",
          "city": "Wien",
          "country": "Austria"
        }
      }
    ]
  }
}
```

**RESPONSE**

Status: 200

Content-Type: application/json

```
{
  "id": "c1234",
  "href": "http://serverlocation:port/customerManagement/customer/c1234",
  "name": "DisplayName",
  "status": "Active",
  "description": "Description string",
  "validFor": {
    "startDateTime": "2013-06-19T04:00:00.0Z",
    "endDateTime": "2013-04-19T20:42:23.0Z"
  },
}
```

```
"customerRank": "3",
"relatedParty": {
  "id": "1",
  "href": "http://serverlocation:port/partyManagement/individual/1",
  "role": "customer",
  "name": "John Doe"
},
"characteristic": [
  {
    "name": "characteristicname1",
    "value": "characteristicvalue1"
  },
  {
    "name": "characteristicname2",
    "value": "characteristicvalue2"
  }
],
"contactMedium": [
  {
    "type": "Email",
    "validFor": {
      "startDateTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "emailAddress": "abc@tmforum.com"
    }
  },
  {
    "type": "PostalAddress",
    "validFor": {
      "startDateTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
      "city": "Wien",
      "country": "Austria",
      "postcode": "1020",
      "stateOrProvince": "Quebec",
```



```
        "street1": "Lassallestrasse7",
        "street2": ""
    }
},
{
    "type": "TelephoneNumber",
    "validFor": {
        "startTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
        "type": "mobile",
        "number": "+436641234567"
    }
},
{
    "preferred": true,
    "type": "TelephoneNumber",
    "validFor": {
        "startTime": "2013-04-19T20:42:23.0Z"
    },
    "medium": {
        "type": "business",
        "number": "+436641234567"
    }
}
],
"customerAccount": [
    {
        "id": "1",
        "href": "http://serverlocation:port/customerManagement/customerAccount/1",
        "name": "CustomerAccount1",
        "description": "CustomerAccountDesc1",
        "accountStatus": "Active"
    },
    {
        "id": "2",
        "href": "http://serverlocation:port/customerManagement/customerAccount/2",
```

```
        "name": "CustomerAccount2",
        "description": "CustomerAccountDesc2",
        "accountStatus": "Active"
    }
],
"customerCreditProfile": [
    {
        "creditProfileDate": "2013-04-19T20:42:23.0Z",
        "validFor": {
            "startDateTime": "2013-04-19T20:42:23.0Z",
            "endDateTime": "2013-06-19T04:00:00.0Z"
        },
        "creditRiskRating": 1,
        "creditScore": 1
    },
    {
        "creditProfileDate": "2013-04-19T20:42:23.0Z",
        "validFor": {
            "startDateTime": "2013-04-19T20:42:23.0Z",
            "endDateTime": "2013-06-19T04:00:00.0Z"
        },
        "creditRiskRating": 1,
        "creditScore": 1
    }
],
"paymentMean": [
    {
        "id": "45",
        "href": "http://serverlocation:port/customerManagement/paymentMean/45",
        "name": "my favourite payment mean"
    },
    {
        "id": "64",
        "href": "http://serverlocation:port/customerManagement/paymentMean/64",
        "name": "my credit card payment mean"
    }
]
```

```
}

```

## POST /customerManagement/customer

This Uniform Contract operation is used to create a customer.

### Description:

- This operation is used to create one customer
- The identifier is a string that can consist of numbers, not necessarily alphanumeric
- The mandatory element(s) is/are
  - Customer Name
- The id is generated automatically

### Behavior:

- Return Status Codes:
  - 201 Customer create successfully. The Content-Location Header must be set and can be used to retrieve the customer
  - 400 Bad Request Error
  - 500 The server encountered an unexpected condition which prevented it from fulfilling the request

Attribute name	Mandatory	Default	Rule
id	N	Automatically generated	If not given, the id is generated by the system. It is also possible to add an ID in the POST request, which is then used
name	Y		
status	N		
description	N		
validFor	N		

customerRank	N		
party	N		
characteristic	N		
contactMedium	N		
customerAccount	N		
customerCreditProfile	N		
paymentMean	N		

## Rules:

Rule name	Rule/Pre Condition/Side Effects/Post Conditions
relatedParty	ref is mandatory
characteristic	name and value are mandatory
contactMedium	type and medium are mandatory
customerAccount	id, name and accountStatus are mandatory
customerCreditProfile	creditProfileDate and validFor are mandatory
paymentMean	Id, href are mandatory

The below query pattern can be used to create a single customer only the mandatory attributes:

REQUEST
POST /customerManagement/customer/ Content-type: application/json { "name": "DisplayName" }

```
}
```

**RESPONSE**

Status:201

Content-Type: application/json

Content-Location: <http://serverlocation:port/customerManagement/customer/c123423243>

```
{
  "id": "c123423243",
  "name": "DisplayName",
  "status": "New",
  "validFor": {
    "startDateTime": "2013-04-19T16:42:23-04:00",
    "endDateTime": ""
  }
}
```

The below query pattern can be used to create a single customer with more attributes:

**REQUEST**

POST /customerManagement/customer/

Content-type: application/json

```
{
  "name": "DisplayName",
  "customerRank": 3,
  "relatedParty": {
    "id": "1",
    "href": "http://serverlocation:port/partyManagement/individual/1"
  },
  "contactMedium": [
    {
      "type": "Email",
      "medium": {
        "emailAddress": "abc@tmforum.com"
      }
    }, {
      "preferred": true,
      "type": "PhoneNumber",

```

```

        "medium": {
            "type": "business",
            "number": "+436641234567"
        }
    },
    ],
    "customerCreditProfile": [
        {
            "creditProfileDate": "2013-04-19T16:42:23-04:00",
            "validFor": {
                "startDateTime": "2013-04-19T16:42:23-04:00",
                "endDateTime": "2013-06-19T00:00:00-04:00"
            },
            "creditRiskRating": "1",
            "creditScore": "1"
        }, {
            "creditProfileDate": "2013-04-19T16:42:23-04:00",
            "validFor": {
                "startDateTime": "2013-04-19T16:42:23-04:00",
                "endDateTime": "2013-06-19T00:00:00-04:00"
            },
            "creditRiskRating": "1",
            "creditScore": "1"
        }
    ]
}

```

## RESPONSE

Status:201

Content-Type: application/json

Content-Location: <http://serverlocation:port/customerManagement/customer/c1234>

```

{
    "id": "c1234",
    "name": "DisplayName",
    "status": "New",
    "validFor": {
        "startDateTime": "2013-04-19T16:42:23-04:00",
        "endDateTime": ""
    }
}

```

```
    },
    "customerRank": 3,
    "relatedParty": {
      "id": "1",
      "href": "http://serverlocation:port/partyManagement/individual/1"
    },
    "contactMedium": [
      {
        "type": "Email",
        "medium": {
          "emailAddress": "abc@tmforum.com"
        }
      }, {
        "preferred": true,
        "type": "TelephoneNumber",
        "medium": {
          "type": "business",
          "number": "+436641234567"
        }
      }
    ],
    "customerCreditProfile": [
      {
        "creditProfileDate": "2013-04-19T16:42:23-04:00",
        "validFor": {
          "startDateTime": "2013-04-19T16:42:23-04:00",
          "endDateTime": "2013-06-19T00:00:00-04:00"
        },
        "creditRiskRating": "1",
        "creditScore": "1"
      }, {
        "creditProfileDate": "2013-04-19T16:42:23-04:00",
        "validFor": {
          "startDateTime": "2013-04-19T16:42:23-04:00",
          "endDateTime": "2013-06-19T00:00:00-04:00"
        },
        "creditRiskRating": "1",
```

```

        "creditScore": "1"
    }
]
}

```

## DELETE /customerManagement/customer/{ID}

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

### Description:

- This operation is used to delete customer(s) using customer id(s)
- The resource represents a managed entity
- The identifier is a string that can consist of numbers, not necessarily alphanumeric

### Behavior:

- Return Status Codes:
  - 204 – NO Content. Customer Deleted successfully
  - 400 – Bad Request Error
  - 404 – Customer Data Not Found
  - 500 - The server encountered an unexpected condition which prevented it from fulfilling the request

### REQUEST

```

DELETE /customerManagement/customer/c1234
Content-type: application/json

```

### RESPONSE

```
Status:204
```

## GET customerManagement/customerAccount/{ID}

Note that collections can be retrieved via GET /customerManagement /customerAccount with no {ID}

### Description :

- This operation returns all customerAccounts, unless an ID is specified in which case a specific customerAccount resource would be returned.



- The resource instance being returned is a customerAccount or an array of customerAccounts if the query returns multiple resources
- Filtering is enabled on all customerAccount attributes.
- Attribute selection is enabled.
- The ID may be a string (or a string containing numbers).

#### Behavior :

- What status and exception codes are returned.
  - 200 OK – customer account was retrieved
  - 404 Not found when the supplied ID doesn't match a known customerAccount.
  - 500 Internal server Error
- Returns HTTP/1.1 status code 200 if the request was successful.
- Any other special return and/or exception codes.

Examples: Get request with one customer including all attributes in the response. Filtering and attribute selection is described in the examples following.

#### REQUEST

```
GET /customerManagement/customerAccount/{ID}/?{fields=attributes}&{filtering expression}
Accept: application/json
```

#### RESPONSE

```
{
  "id": "ca1234",
  "href": "http://serverlocation:port/customerManagement/customerAccount/ca1234",
  "lastModified": "2014-09-24T18:27:37.059899Z",
  "name": "sampleaccount",
  "accountType": "Residential",
  "status": "Active",
  "description": "Description string",
  "creditLimit": 1212121,
  "pin": "pin0",
  "receivableBalance": 52.3,
  "customerAccountTaxExemption": [
    {
      "issuingJurisdiction": "SampleJurisdiction",
      "certificateNumber": "CATaxExemption1",
      "reason": "Reason",
    }
  ]
}
```

```
    "validFor": {
      "startDateTime": "2013-04-19T16:42:23.0Z",
      "endDateTime": "2013-06-19T00:00:00.0Z"
    }
  },
  {
    "issuingJurisdiction": "SampleJurisdiction",
    "certificateNumber": "CATaxExemption1",
    "reason": "Reason",
    "validFor": {
      "startDateTime": "2013-04-19T16:42:23.0Z",
      "endDateTime": "2013-06-19T00:00:00.0Z"
    }
  }
],
"customerAccountRelationship": [
  {
    "validFor": {
      "startDateTime": "2013-04-19T16:42:23.0Z",
      "endDateTime": "2013-06-19T00:00:00.0Z"
    },
    "customerAccount": [
      {
        "id": "1",
        "href":
"http://serverlocation:port/customerManagement/customerAccount/1",
        "name": "CustomerAccount1",
        "description": "CustomerAccountDesc1"
      }
    ]
  }
],
"contact": [
  {
    "contactType": "primary",
    "validFor": {
      "startDateTime": "2013-04-19T16:42:23.0Z",
```

```
        "endDateTime": "2013-06-19T00:00:00.0Z"
    },
    "contactName": "DisplayName",
    "partyRoleType": "CustomerAccountRepresentative",
    "contactMedium": [
        {
            "type": "Email",
            "medium": {
                "emailAddress": "abc@tmforum.com"
            }
        },
        {
            "type": "PostalAddress",
            "medium": {
                "city": "Wien",
                "country": "Austria",
                "postcode": "1020",
                "stateOrProvince": "Quebec",
                "street1": "Lassallestrasse7"
            }
        },
        {
            "type": "TelephoneNumber",
            "medium": {
                "type": "mobile",
                "number": "+436641234567"
            }
        },
        {
            "preferred": true,
            "type": "TelephoneNumber",
            "medium": {
                "type": "business",
                "number": "+436641234567"
            }
        }
    ],
},
```

```
    "relatedParty": {
      "id": "1",
      "href": "http://serverlocation:port/partyManagement/individual/1",
      "role": "customer"
    }
  },
],
"customer": {
  "id": "1",
  "href": "http://serverlocation:port/customerManagement/customer/1",
  "name": "Customer1",
  "description": "CustomerDesc1"
},
"customerAccountBalance": [
  {
    "type": "ReceivableBalance",
    "amount": 52.3,
    "validFor": {
      "startDateTime": "2013-04-19T16:42:23.0Z",
      "endDateTime": "2013-06-19T00:00:00.0Z"
    },
    "status": "Due"
  },
  {
    "type": "DepositBalance",
    "amount": 52.3,
    "validFor": {
      "startDateTime": "2013-04-19T16:42:23.0Z",
      "endDateTime": "2013-06-19T00:00:00.0Z"
    },
    "status": "Paid"
  }
],
"paymentPlan": [
  {
    "status": "Effective",
    "type": "Type1",
```

```
    "priority": 1,
    "amount": 15.3,
    "paymentFrequency": "monthly",
    "numberOfPayments": 4,
    "validFor": {
      "startDateTime": "2013-04-19T16:42:23.0Z",
      "endDateTime": "2013-06-19T00:00:00.0Z"
    },
    "paymentMean": {
      "id": "45",
      "href": "http://serverlocation:port/customerManagement/paymentMean/45",
      "description": "My favourite payment mean"
    }
  },
  {
    "status": "Ineffective",
    "type": "Type2",
    "priority": 2,
    "amount": 20,
    "paymentFrequency": "monthly",
    "numberOfPayments": 2,
    "validFor": {
      "startDateTime": "2013-04-19T16:42:23.0Z",
      "endDateTime": "2013-06-19T00:00:00.0Z"
    },
    "paymentMean": {
      "id": "70",
      "href": "http://serverlocation:port/customerManagement/paymentMean/70",
      "name": "my credit card payment mean"
    }
  }
]
}
```

In order to address a specific query a list of customer, with only a subset of the attributes in the response, the query pattern would look like the following:

REQUEST
GET /customerManagement/customerAccount/?fields=name,id,accountType,status,creditLimit,pin&accountType="residential" Accept: application/json
RESPONSE for Customer Account
Status:200 Content-Type: application/json <pre>[   {     "id": "ca1234",     "name": "sampleaccount",     "accountType": "Residential",     "status": "Active",     "creditLimit": 1212121,     "pin": "pin0"   }, {     "id": "badasf1234",     "name": "sampleaccount 2",     "accountType": "Residential",     "status": "Active",     "creditLimit": 1212121,     "pin": "pin0"   } ]</pre>

In order to address a specific query against the Primary customer account (having an id) of an hierarchical account structure and also to reflect the hierarchy, the query pattern in this case would be as follows:

REQUEST
GET /customerManagement/customerAccount/ca1234?fields=name,id,accountType,accountStatus,creditLimit,pin,customerAccountRelationship

**RESPONSE for Customer Account**

```
{
  "id": "ca1234",
  "name": "sampleaccount",
  "accountType": "Residential",
  "status": "Active",
  "creditLimit": 1212121,
  "pin": "pin0",
  "customerAccountRelationship": [
    {
      "relationshipType": "linked",
      "validFor": {
        "startDateTime": "2013-04-19T16:42:23-04:00",
        "endDateTime": ""
      },
      "customerAccount": {
        "id": "ca1235",
        "href":
"http://serverlocation:port/customerManagement/customerAccount/ca1235",
        "name": "sampleaccount2",
        "description": "Description sampleaccount2"
      }
    }
  ]
}
```

In order to address a specific query against those customer accounts which have been modified after a certain date time, the query pattern in this case would be as follows:

**REQUEST**

```
GET /customerManagement/customerAccount/?fields=name,id,accountType,
accountStatus,creditLimit,pin,lastModified&lastModified.gt="2013-08-08"
```

```
Accept: application/json
```

**RESPONSE for Customer Account**

```
Status:200
Content-Type: application/json
[
  {
    "id": "ca1234",
    "name": "sampleaccount",
    "accountType": "Residential",
    "status": "Active",
    "creditLimit": 1212121,
    "pin": "pin0",
    "lastModified": "2014-04-19T16:42:23-04:00"
  }, {
    "id": "ca123345",
    "name": "sampleaccount3",
    "accountType": "Residential",
    "status": "Active",
    "creditLimit": 757,
    "pin": "pin0",
    "lastModified": "2013-04-19T16:42:23-04:00"
  }
]
```

#### PUT customerManagement/customerAccount/{ID}

PUT is not supported for customerAccount as any modification can be handled through PATCH API.

#### PATCH CUSTOMERMANAGEMENT/CUSTOMERACCOUNT/{ID}

Description :

- This Uniform Contract operation is used to partially update the representation of a customer account.
- Resource represents the customer account.
- Behavior :
  - Update of customer account will be based on identifier( id)
  - Update of customer account is allowed to all attributes except attributes which are set by backend and are read-only (id, lastModified, etc.)



- Update of name, accountType, status, creditLimit is allowed through PATCH API.
- After creation of customer account using mandatory parameters(mentioned in POST API),Update / linking of customer[], financialcharge[], relatedParty[], paymentPlan[] should be possible through PATCH.
- Child customer account can also be linked/updated to the parent customer by updating customerAccountRelationship[] in customerAccount resource.
- The resource instance being returned is a customerAccount

#### Behavior :

- Returns HTTP/1.1 status code 200 if the request was successful.
- Returns HTTP/1.1 status code 400 (Bad request) if content is invalid (missing required attributes, ...).
- 500 – Internal server Error

Note: -The requester cannot update the id. The lastModified attributes is updated automatically in the back-end.

Attribute name	Patchable	Rule
id	N	Cannot be updated from outside
lastModified	N	Is updated automatically
name	Y	
accountType	Y	
status	Y	
description	Y	
creditLimit	Y	
pin	Y	
receivableBalance	Y	
customerAccountTaxExemption	Y	

customerAccountRelationship	Y	
contact	Y	
customer	Y	
customerAccountBalance	Y	
paymentPlan	Y	
paymentMean	Y	

Rule name	Rule/Pre Condition/Side Effects/Post Conditions
customerAccountTaxExemption	issuingJurisdiction and validFor are mandatory
customerAccountRelationship	relationshipType and validFor are mandatory
contact	contactType and validFor are mandatory
customer	id and name are mandatory
customerAccountBalance	id, type, amount, validFor and status are mandatory
paymentPlan	id, status, amount, paymentFrequency and validFor as well as paymentMean are mandatory
paymentMean	Id, href are mandatory

Eg: PATCH for adding a new customerAccountTaxExemption to customer account.

<b>REQUEST</b>
PATCH /customerManagement/customerAccount/{ID} Content-type: application/json-patch+json {

<pre> "op": "add", "path": "/customerAccount/customerAccountTaxExemption", "value": {   "issuingJurisdiction": "Sample Jurisdiction",   "certificateNumber": "CA Tax Exemption 3",   "reason": "Reason",   "validFor": {     "startDateTime": "2013-04-19T16:42:23-04:00",     "endDateTime": ""   } } </pre>
<b>RESPONSE</b>
Status:200 Content-Type: application/json { JSON Resource Representation with every attributes including the added tax exemption }

### PATCH of accountType in customerAccount resource

<b>REQUEST</b>
PATCH/customerManagement/customerAccount/ca1234 Content-type: application/json <pre> {   "accountType": "Business" } </pre>
<b>RESPONSE</b>
Status:200 Content-Type: application/json { JSON Resource Representation with every attributes including the changed account type }

## POST CUSTOMERMANAGEMENT/CUSTOMERACCOUNT/{ID}

Description :

- This Uniform Contract operation is used to create a customer account.

- Resource represents a managed entity.
- The resource instance being returned is a customerAccount
- Mandatory attributes that must be provided when you create the customer account :

#### Behavior :

- Returns HTTP/1.1 status code 201 if the request was successful.
- Returns HTTP/1.1 status code 400 (Bad request) if content is invalid (missing required attributes, ...).
- 500 – Internal server Error

The requester cannot generate the id. The id and lastModified attributes are generated automatically in the back-end.

Required Attributes: Name, AccountType

Read-Only Attributes: id, lastModified, receivableBalance, customerAccountBalance

Attribute name	Mandatory	Default	Rule
id	N	Automatically generated	If not given, the id is generated by the system. It is also possible to add an ID in the POST request, which is then used
lastModified	N	Automatically generated	Cannot be set from outside
name	Y		
accountType	Y		
status	N		
description	N		

creditLimit	N		
pin	N		
receivableBalance	N		
customerAccountTaxExemption	N		
customerAccountRelationship	N		
contact	N		
customer	N		
customerAccountBalance	N		
paymentPlan	N		
paymentMean	N		

## Rules:

Rule name	Rule/Pre Condition/Side Effects/Post Conditions
customerAccountTaxExemption	issuingJurisdiction and validFor are mandatory
customerAccountRelationship	relationshipType and validFor are mandatory
contact	contactType and validFor are mandatory
customer	id and name are mandatory
customerAccountBalance	id, type, amount, validFor and status are mandatory
paymentPlan	id, status, amount, paymentFrequency and validFor are mandatory

paymentMean	Id, href are mandatory
-------------	------------------------

Create a customer account only with mandatory attributes:

REQUEST
POST customerManagement/customerAccount/{ID} Content-type: application/json <pre>{   "name": "sample account number 1",   "accountType": "Residential" }</pre>
RESPONSE
Status:201 Content-Type: application/json Content-Location: " <a href="http://serverlocation:port/customerManagement/customerAccount/ca1234">http://serverlocation:port/customerManagement/customerAccount/ca1234</a> " <pre>{   "id": "ca1234",   "lastModified": "2013-06-19T00:00:00-04:00",   "name": "sample account number 1",   "accountType": "Residential",   "status": "Active" }</pre>

## DELETE CUSTOMERMANAGEMENT/CUSTOMERACCOUNT/{ID}

Note customer account will be deleted via DELETE /customerAccount/{ID} and will not allow deletion of customers if no id is provided.

Description:

- This operation will delete the customer Account with the specified ID
- The return will be response code and will not have any resource.
- Attribute selection is disabled.
- The ID may be a string (or a string containing numbers).

Behavior :

- What status and exception codes are returned.

- 404 Not found when the supplied ID doesn't match a known customerAccount.
- Returns HTTP/1.1 status code 204 if the request was successful.
- 500 – Internal server Error

The following example shows deletion of customer account with filtration criteria based on id.

<b>REQUEST</b>
DELETE /customerManagement/customerAccount/ca1234 Accept: application/json
<b>RESPONSE</b>
Status:204

#### GET customerManagement/paymentMean/{ID}

Note that collections can be retrieved via GET /customerManagement /paymentMean with no {ID}

Description :

- This operation returns all paymentMean, unless an ID is specified in which case a specific paymentMean resource would be returned.
- The resource instance being returned is a paymentMean or an array of paymentMean if the query returns multiple resources
- Filtering is enabled on all paymentMean attributes.
- Attribute selection is enabled.
- The ID may be a string (or a string containing numbers).

Behavior :

- What status and exception codes are returned.
  - 200 OK – paymentMean was retrieved
  - 404 Not found when the supplied ID doesn't match a known paymentMean.
  - 500 Internal server Error
- Returns HTTP/1.1 status code 200 if the request was successful.
- Any other special return and/or exception codes.

Examples: Get request with one payment mean including all attributes in the response. Filtering and attribute selection is described in the examples following.

<b>REQUEST</b>
----------------

GET /customerManagement/paymentMean/{ID}/?{fields=attributes}&{filtering expression}

Accept: application/json

## RESPONSE

```
{
  "id": "45",
  "href": "http://serverlocation:port/customerManagement/paymentMean/45",
  "name": "My favourite payment mean",
  "validFor": {
    "startDateTime": "2013-04-19T16:42:23-04:00",
    "endDateTime": "2014-04-19T16:42:23-04:00"
  },
  "paymentMeanType": "BankAccountDebit",
  "relatedParty": {
    "id": "1",
    "role": "customer",
    "name": "Gustave Flaubert",
    "href": "http://serverlocation:port/partyManagement/individual/1"
  },
  "bankAccount": {
    "BIC": "PSSTFRPPPAR",
    "domiciliation": "LaBanquePostale-75900ParixCedex15",
    "IBAN": "FR4620061009010835927F33098",
    "accountHolder": "Mr.GustaveFlaubert"
  }
}
```

## PUT customerManagement/paymentMean/{ID}

PUT is not supported for paymentMean as any modification can be handled through PATCH API.

## PATCH CUSTOMERMANAGEMENT/PAYMENTMEAN/{ID}

Description :

- This Uniform Contract operation is used to partially update the representation of a payment mean.
- Resource represents the payment mean.
- Behavior :
  - Update of paymentMean will be based on identifier( id)
  - Update of paymentMean is allowed to all attributes except attributes which are set by backend and are read-only (id, href, etc.)
  - Update of name, validFor is allowed through PATCH API.

Behavior :



- Returns HTTP/1.1 status code 200 if the request was successful.
- Returns HTTP/1.1 status code 400 (Bad request) if content is invalid (missing required attributes, ...).
- 500 – Internal server Error

Attribute name	Patchable	Rule
id	N	Cannot be updated from outside
href	N	Cannot be updated from outside
name	Y	
validFor	Y	

Rule name	Rule/Pre Condition/Side Effects/Post Conditions

Eg: PATCH for updating the name.

REQUEST
PATCH /customerManagement/customerAccount/{ID} Content-type: application/json-patch+json <pre>{   "op": "update",   "path": "/customerAccount/paymentMean",   "value": {     "name": "new name for my credit card"   } }</pre>
RESPONSE
Status:200

Content-Type: application/json

{ JSON Resource Representation with every attributes }

## POST CUSTOMERMANAGEMENT/PAYMENTMEAN/{ID}

### Description :

- This Uniform Contract operation is used to create a payment mean.
- Resource represents a managed entity.
- The resource instance being returned is a paymentMean
- Mandatory attributes that must be provided when you create the paymentMean :

### Behavior :

- Returns HTTP/1.1 status code 201 if the request was successful.
- Returns HTTP/1.1 status code 400 (Bad request) if content is invalid (missing required attributes, ...).
- 500 – Internal server Error

The requester cannot generate the id. The id is generated automatically in the back-end.

Attribute name	Mandatory	Default	Rule
id	N	Automatically generated	If not given, the id is generated by the system. It is also possible to add an ID in the POST request, which is then used
href	N	Automatically generated	Cannot be set from outside
name	Y		
paymentMeanType	Y		
relatedParty	Y		
bankAccount	N		
creditCard	N		

## Rules:

Rule name	Rule/Pre Condition/Side Effects/Post Conditions
bankAccount	Mandatory if paymentMeanType is different from Credit card
creditCard	Mandatory is paymentMeanType is equal to Credit card

Create a customer account only with mandatory attributes:

REQUEST
POST customerManagement/paymentMean/{ID} Content-type: application/json <pre> {   "name": "My favourite payment mean",   "validFor": {     "startDateTime": "2013-04-19T16:42:23-04:00",     "endDateTime": "2014-04-19T16:42:23-04:00"   },   "paymentMeanType": "BankAccountDebit",   "relatedParty": {     "id": "1",     "role": "customer",     "name": "Gustave Flaubert",     "href": "http://serverlocation:port/partyManagement/individual/1"   },   "bankAccount": {     "BIC": "PSSTFRPPPAR",     "domiciliation": "LaBanquePostale-75900ParixCedex15",     "IBAN": "FR4620061009010835927F33098",     "accountHolder": "Mr.GustaveFlaubert"   } }           </pre>

**RESPONSE**

Status:201

Content-Type: application/json

```
{
  "id": "45",
  "href": "http://serverlocation:port/customerManagement/paymentMean/45",
  "name": "My favourite payment mean",
  "validFor": {
    "startDateTime": "2013-04-19T16:42:23-04:00",
    "endDateTime": "2014-04-19T16:42:23-04:00"
  },
  "paymentMeanType": "BankAccountDebit",
  "relatedParty": {
    "id": "1",
    "role": "customer",
    "name": "Gustave Flaubert",
    "href": "http://serverlocation:port/partyManagement/individual/1"
  },
  "bankAccount": {
    "BIC": "PSSTFRPPPAR",
    "domiciliation": "LaBanquePostale-75900ParixCedex15",
    "IBAN": "FR4620061009010835927F33098",
    "accountHolder": "Mr.GustaveFlaubert"
  }
}
```

**DELETE CUSTOMERMANAGEMENT/PAYMENTMEAN/{ID}**

Note payment mean will be deleted via DELETE /paymentMean/{ID} and will not allow deletion of paymentMean if no id is provided.

Description:

- This operation will delete the paymentMean with the specified ID
- The return will be response code and will not have any resource.
- Attribute selection is disabled..
- The ID may be a string (or a string containing numbers).

**Behavior :**

- What status and exception codes are returned.
  - 404 Not found when the supplied ID doesn't match a known paymentMean.
  - Returns HTTP/1.1 status code 204 if the request was successful.
  - 500 – Internal server Error

The following example shows deletion of paymentMean with filtration criteria based on id.

<b>REQUEST</b>
DELETE /customerManagement/customerAccount/45 Accept: application/json
<b>RESPONSE</b>
Status:204

## 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 FOR CUSTOMERMANAGEMENT

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.

<b>REQUEST</b>
POST /customerManagement/customer/hub Accept: application/json  <pre>{ "callback": "http://in.listener.com" }</pre>
<b>RESPONSE</b>
Status:201 Content-Type: application/json Content-Location: <a href="http://serverlocation:port/customerManagement/customer/hub/42">http://serverlocation:port/customerManagement/customer/hub/42</a>  <pre>{"id":"42","callback":"http://in.listener.com","query":null}</pre>
<b>REQUEST</b>

POST /customerManagement/customerAccount/hub

Accept: application/json

```
{ "callback": "http://in.listener.com" }
```

#### RESPONSE

Status:201

Content-Type: application/json

Content-Location: <http://serverlocation:port/customerManagement/customerAccount/hub/42>

```
{ "id": "42", "callback": "http://in.listener.com", "query": null }
```

#### REQUEST

POST /customerManagement/paymentMean/hub

Accept: application/json

```
{ "callback": "http://in.listener.com" }
```

#### RESPONSE

Status:201

Content-Type: application/json

Content-Location: <http://serverlocation:port/customerManagement/paymentMean/hub/42>

```
{ "id": "42", "callback": "http://in.listener.com", "query": null }
```

## UNREGISTER LISTENER DELETE

### CUSTOMERMANAGMENT/CUSTOMERACCOUNT/HUB/{ID}

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.

DELETE customer hub

<b>REQUEST</b>
DELETE /customerManagement/customer/hub/{id} Accept: application/json
<b>RESPONSE</b>
Status:204

## DELETE customer account hub

<b>REQUEST</b>
DELETE /customerManagement/customerAccount/hub/{id} Accept: application/json
<b>RESPONSE</b>
Status:204

## DELETE payment mean hub

<b>REQUEST</b>
DELETE /customerManagement/paymentMean/hub/{id} Accept: application/json
<b>RESPONSE</b>
Status:204



## RELEASE HISTORY

Release Number	Date	Release led by:	Description
0.a	28-Nov-2013	Arindam Basu (Ericsson) Sudipta Dutta (Ericsson) Brinda Santh M (IBM) Hemant Gour (IBM) Shashank Singh (IBM) Vishal Aggarwal (IBM) Nitish Jaisoor (Infosys) Rajesh Gannaprapu (Infosys) Srinivasan Duraiswamy (Infosys) Veeramani K (Infosys)	Initial Spec prepared in the TMF Conference held at Bangalore, India (Nov 26 to Nov 28)
0.b	5-Mar-2014	Christian Traxler (Infonova) Alexander Sturn (Infonova) Gilbert Scheibelhofer (A1) Josh Salomon (Amdocs) Véronique Mauneau (Orange) Maxime Delon (Orange)	Extension of customer API mgmt. during Spec Jam held in Vienna, Austria

## CONTRIBUTORS TO DOCUMENT

Veronique Mauneau	Orange
Jean-Luc Tymen	Orange
Pierre Gauthier	TM Forum
John Morey	Ciena
Cliff C Faurer	AMKB Cloud