SMARTVISTA EXCHANGE PROTOCOL WITH OMNICHANNEL (SVXP OMNI ISS)

API developer reference

April 2018

Contents

[1 PREFACE 3](#_Toc511213327)

[1.1 Revision history 3](#_Toc511213328)

[1.2 Document purpose 3](#_Toc511213329)

[2 SMARTVISTA INTEGRATION SERVICES OVERVIEW 3](#_Toc511213330)

[2.1 General concepts 3](#_Toc511213331)

[2.2 Data types, Occurrence, Dictionaries 3](#_Toc511213332)

[3 CUSTOMER FILE STRUCTURE 4](#_Toc511213333)

[3.1 Overview 4](#_Toc511213334)

[3.2 References 4](#_Toc511213335)

[3.3 List of elements 5](#_Toc511213336)

[4 MERCHANT FILE STRUCTURE 9](#_Toc511213337)

[4.1 Overview 9](#_Toc511213338)

[4.2 References 9](#_Toc511213339)

[4.3 List of elements 9](#_Toc511213340)

[5 DICTIONARIES 12](#_Toc511213341)

1. PREFACE
   1. Revision history

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Revision | Date | Author | | Details |
| 1.0 | 16.04.2018 | | Kolodkina Y. | Initial version |

* 1. Document purpose

SVXP OMNI is a reference manual for developers who are implementing API of the SmartVista solution. This document is written for internal use of BPC. The document describes the content and the structure of the API.

It is supposed document users to be familiar with financial transactions, communications and XML data format.

1. SMARTVISTA INTEGRATION SERVICES OVERVIEW
   1. General concepts

SmartVista exchange protocol with OmniChannel (SVXP OMNI hereafter) provides a description of the file formats of information upload into OmniChannel (OMNI) from SmartVista. File format xml. For each format will be described in this document, XML Schema Definition language (XSD) and provided examples.

* 1. Data types, Occurrence, Dictionaries

For SVXP OMNI methods the standard XML data types are used. Those are fully described in the following document**XML Schema Part 2: Datatypes Second Edition and** t can be found here: [***http://www.w3.org/TR/xmlschema-2/***](http://www.w3.org/TR/xmlschema-2/)

Within the current document all the SVXP OMNI messages are described in the table structure below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tag name | Data type | Length | Occurrence (min-max) | Description |
| Card | | | | |
| **card\_id** | string | 200 | 1-1 | Internal card instance identifier(UID) |
| inst\_id | int | 4 | 1-1 | An institution ID |
| card\_number | string | 24 | 1-1 | A card number(PAN). It must be unique in the scope of an institution. It can be masked. |
| card\_mask | string | 24 | 0-1 | Masked card number |
| card\_type | int | 4 | 0-1 | Card type: Visa Gold, Visa Classic etc. Dictionary value |
| category | string | 8 | 0-1 | Category of card. Value from dictionary CRCG |
| Lang | String | 7 | 0-1 | Language. Please refer to LANG dictionary |

**Data Type:** SVXP OMNI tags can be of Primitive XML Data Types (string, long, boolean, etc ) or Complex Data Types (Aggregates).

**Occurrence**: This field defines if the field is mandatory or optional (first number) as well as maximum number occurrences of this tag in the message (last digit)

e.g. 1-1 = minOccurs="1" maxOccurs="1"

Documentation is provided along with Examples of the Request messages for all of the methods described below.

1. CARD FILE STRUCTURE
   1. Overview

The file contains information on cards. Including card data and card services data. The information about cards is synchronized between the SmartVista and OmniChannel.

The direction of the file may be only OUTGOING.

Tag CARDS is root tag and it include itself one or more tags CARD.

* 1. References

Format of card file described by XSD file.

SVXP OMNI XSD: svxp\_omni\_iss\_card.xsd

Example of xml document:

svxp\_omni\_iss\_card.xml – cards information.

* 1. List of elements

| Tag name | Data type | Length | Is mandatory | Description |
| --- | --- | --- | --- | --- |
| cards | | | | It is root tag and it include itself one or more tags card |
| file\_id | long | 16 | 0-1 | Unique identifier of outgoing file |
| file\_type | string | 8 | 1-1 | Type of outgoing file. Describe the purpose of data in file. Dictionary FLTP. Value FLTPCSIF. |
| inst\_id | int | 4 | 0-1 | An institution ID |
| card | complex |  | 1-\* | Card data. |
| card | | | | |
| card\_id | string | 200 | 1-1 | Internal card instance identifier(UID) |
| card\_number | string | 24 | 1-1 | A card number(PAN). It must be unique in the scope of an institution. It can be masked. |
| card\_type | int | 4 | 0-1 | Card type: Visa Gold, Visa Classic, MasterCard Gold etc. |
| customer | complex |  | 1-1 | Customer-owner of card. Should contain an attribute customer\_id |
| contract | complex |  | 1-1 | Contract by which card was registered. Should contain an attribute contract\_id |
| available\_services | complex |  | 1-\* | Available services of card |
| attached\_services | complex |  | 0-\* | Attached services of card |
| customer | | | | |
| customer\_id | Long | 12 | 1-1 | Unique identifier of customer in SVBO. It can be used for search and update/insert data it OMNI |
| customer\_number | String | 200 | 1-1 | An external ID of a customer. It must be unique in the scope of an institution |
| contract | | | | |
| contract\_id | Long | 12 | 1-1 | Unique identifier of contract in SVBO. It can be used for search and update/insert data it OMNI |
| contract\_number | String | 200 | 1-1 | A contract number. It must be unique in the scope of an institution |
| available\_services | | | | |
| service\_type | string | 8 | 1-1 | Service type |
| service\_type\_name | string | 200 | 1-1 | Service name |
| service\_external\_code | string | 200 | 0-1 | Service external code |
| service\_number | string | 200 | 0-1 | Service number |
| attached\_services | | | | |
| service\_type | string | 8 | 1-1 | Service type |
| service\_type\_name | string | 200 | 1-1 | Service name |
| service\_external\_code | string | 200 | 0-1 | Service external code |
| service\_number | string | 200 | 0-1 | Service number |