

Wire Data Model and Services 3.2

# Document Status

status: Request for Comment (valid values are < Request for Comment, Preliminary Review, Public Review, Architectural Review, Final Review, Published, Deprecated)

This version: **Assembla**.com. Files Tag = CUFX\_3.2\_RFC\_Active

Previous Version: **Assembla**.com. Files Tag = CUFX\_3.1\_RFC\_Archive

# Change Log

|  |  |  |
| --- | --- | --- |
| Version | Date | Changes |
| 3.0 | **12/19/2013** | * Initial Document |
| 3.0 | **01/14/2014** | * Updated JSON Examples and Use Case |
| 3.0 | **04/02/2014** | * Updated JSON Examples and data attributes for XSD updates. |
| 3.1 | **07/17/2015** | * Updated to release 3.1 |
| 3.2 | **05/10/2016** | * Updated to release 3.2 |

# Overview of Specification

The CUFX Wire Data Model and Service provides a standard description for wire payment definitions and instances of those payments. The CUFX Wire Data Model and Services provide the information necessary to create, read, update or delete Wire transactions between credit union data systems.

# Any know Errors in the document

|  |  |
| --- | --- |
| **Error Description** | Status of Error |
|  |  |

Table of Contents

[Document Status 1](#_Toc376855691)

[Change Log 1](#_Toc376855692)

[Overview of Specification 1](#_Toc376855693)

[Any know Errors in the document 1](#_Toc376855694)

[Document Conventions 2](#_Toc376855695)

[Definitions related to the specification 2](#_Toc376855696)

[Data Elements 3](#_Toc376855697)

[Filters used when accessing the WIRE data 3](#_Toc376855698)

[Use Case 4](#_Toc376855699)

[Services 4](#_Toc376855700)

[Authentication 4](#_Toc376855701)

[Wire-transfer Resource Based Create, Read, Update, Delete 5](#_Toc376855702)

[REST-JSON add Wire-transfer 5](#_Toc376855703)

[REST-JSON Read Wire-transfers 6](#_Toc376855704)

[REST-JSON update Wire-transfer 7](#_Toc376855705)

[REST-JSON Delete pending wire-transfer 8](#_Toc376855706)

[General Error handling For All Services 8](#_Toc376855707)

[Bibliography 8](#_Toc376855708)

# Document Conventions

List any document conventions such as what bold and italics mean and how the document is intended to be read.

“Within this specification, the key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” are to be interpreted as described in W3 Working Group (W3C)]. However, for readability, these words do not appear in all uppercase letters in this specification.

At times, this specification recommends good practice for authors and user agents. These recommendations are not normative and conformance with this specification does not depend on their realization. These recommendations contain the expression “We recommend …”, “This specification recommends …”, or some similar wording.”

All formatting in this document utilize Word Styles.

All Citations must utilize Word Citations to automatically show at the end of the document.

All updates after the initial creation must be performed using Tracking Changes turn on and Accepted by the Architecture committee.

# Definitions related to the specification

Account Data

The CUFX account Data model and Services defines the account data model and Services used by all specifications. Accounts are an abstract summary of loan, share and investment balances at a high level. Any financial product stored in a financial services platform optionally including any transactions associated with that account.

# Data Elements

## Filters used when accessing the WIRE data

Refer to Security Services documentation to understand what may be contained the header and processed by security procedures. When accessing the data include **MessageContext.xsd** so that the service can determine the scope of the request. Refer to recent CUFX messageContext Data and CUFX Security Services for use of MessageContext.xsd. Include any filter variables related to the request. See **WireFilter.xsd.**

## Wire Data attributes

All CUFX fields related to a card are defined in card.xsd. A summary of the attributes is listed here for reference.

Note: Fields not listed in the calling specification are not to be returned to the calling specification. i.e. If the field occurrenceMemo is not listed in the calling specification, then do not return the data field to alleviate issues with unexpected information and bloat of information being returned to light weight applications.

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| occurrenceId | ID to uniquely identify the occurrence of the wire |
| recurringId | Not applicable to wire transactions |
| occurrenceAmount | Amount of the wire |
| occurrenceFromAccountId | The account from which the transfer or payment was taken. |
| occurrenceToAccountId | The account to which the transfer or payment was made. |
| occurrenceStatus | Status of the wire occurrence. |
| fundsWithdrawalType | When will the funds be withdrawn from the account |
| occurrenceMemo | The memo represents a short description that can be applied to the comment of the wire. |
| estimatedProcessDateTime | The date on which to process the wire. |
| createdDateTime | The date that the user submitted the wire. |
| processingStartedDateTime | The date on which the wire processing started. |
| processedDateTime | The date on which the wire was actually processed by the system. |
| completedDateTime | The date on which the core completed the wire. |
| lastUpdatedDateTime | The date of the last known status of the wire. |
| occurrenceConfirmationCode | Unique confirmation code for the wire. |
| queuedForPosting | Indicates whether the wire is pending for posting until system is on-line |
| note | A user's "note to self." |
| batchId | An identifier used to group a set of wires together to process them at the same time |
| customData | Name/data pairs list of fields that are not yet available in the general specification. |
| wireTransferType | Wire transfer type. |
| wireTransferSpecialInstructions | Special instructions for wire transfer. |
| escrowNumber | Escrow number for the wire transfer. |
| escrowOfficerName | Escrow officer name for the wire transfer. |

Fields in blue extended from FundsTransferOccurrenceBase.

# Use Case

CUFX Compliant Application

1

Create Wire Transfer Message

2

Wire Transfer Data Response

Contains Confirmation Code

3

Read Wire Transfer Message

Mapped to Account Id

4

Wire Transfer Data Message Response

Contains confirmation or error

CUFX Core Data Provider

5

Update Wire Transfer Message

6

Wire Transfer Data Message Response

Contains confirmation Code or error

7

Delete Wire Transfer Message

8

Wire Transfer Data Response

Contains Confirmation Code or error

# Services

|  |  |
| --- | --- |
| Definition | WireTransfer Service |
| Overview of Capabilities | Service to create, read, update and delete Wire-Transfer |
| Dependencies | Security Services, messageContext |
| CUFX REST LINK | https://api.dataprovider.com/WireTransfer |
| CUFX WSDL LINK |  |

## Authentication

See Security Services and messageContext for more detail.

## Wire-transfer Resource Based Create, Read, Update, Delete

|  |  |
| --- | --- |
| INPUTS | cufx: WireMessage (which includes)   * [cufx:MessageContext](file:///\\files2\users\CMarjaniemi\Projects\CUFX\MessageContext.html) * cufx:WireFilter (for read, update, delete) * cufx:WireList (for create, update) |
| Outputs | cufx:Transfer (for create, read, update and delete) |
| Return Values | Cufx: Error |
| Side Effects | Creation, read, update or deletion of single wire-transfer. |
| Dependencies | Security Services for authentication and security. |
| Fields used | **Message Headers** : See security services  **messageContext**: See messageContext.xsd  **Filters**: See DepositFilter.xsd  **Attributes: Transfer : See Wires.xsd** |
| Testing Procedures for Certification |  |

### REST-JSON Create Wire-transfer

REQUEST:

**<Security related header parameters... see Security Services>**

Accept: application/json

Accept-Charset: utf-8

Accept-Language: en-us

Content-type: application/json; charset=utf-8

X-API-Version: >=3.0.0

**POST** https://api.dataprovider.com/WireTransfer

{

wireMessage: {

"messageContext" : {<See MessageContext.xsd>

},

"wireList" : {

"Wire”: [{

"occurrenceId": "35464736635",

"occurrenceAmount": {

"value": "234.89",

"currencyCode": "USD"

},

"occurrenceFromAccountId" : account:AccountId,

"createdDateTime" : "2014-01-01T01:01:01Z",

"WireTransferType”: "Escrow",

"WireTransferSpecialInstructions”: xs:string,

"EscrowNumber”: xs:string,

"EscrowOfficerName”: xs:string

}]

}

}

}

RESPONSE

Headers:

Status Code: 200 Ok

Content-type: application/json; charset=utf-8

Content-Language: en-us

Payload:

{

"wireList" : {

"Wire”: [{

"occurrenceId": "35464736635",

"occurrenceAmount": {

"value": "234.89",

"currencyCode": "USD"

},

"occurrenceFromAccountId" : account:AccountId,

"createdDateTime" : "2014-01-01T01:01:01Z",

"occurrenceConfirmationCode": "occurrenceConfirmationCode1",

"WireTransferType”: "Escrow",

"WireTransferSpecialInstructions”: xs:string,

"EscrowNumber”: xs:string,

"EscrowOfficerName”: xs:string

}]

}

}

### REST-JSON Read Wire-transfers

REQUEST:

**<Security related header parameters... see Security Services>**

Accept: application/json

Accept-Charset: utf-8

Accept-Language: en-us

Content-type: application/json; charset=utf-8

X-HTTP-Method-Override:GET

X-API-Version: >=3.0.0

**POST** https://api.dataprovider.com/WireTransfer

{

wireMessage: {

“messageContext”: { <see messageContext.xsd>

},

“wireFilter”: {

“occurrenceIdList” : {

“occurrenceId” : [ “452344”]

},

“accountIDList” {

“accountId”: [ “1234” ]

},

}

}

}

RESPONSE

Headers:

Status Code: 200 Ok

Content-type: application/json; charset=utf-8

Content-Language: en-us

Payload:

{

"wireList" : {

"Wire”: [{

"occurrenceId": "452344",

"occurrenceAmount": {

"value": "234.89",

"currencyCode": "USD"

},

"occurrenceFromAccountId" : “1234”,

"createdDateTime" : "2014-01-01T01:01:01Z",

"occurrenceConfirmationCode": "occurrenceConfirmationCode1",

"WireTransferType”: "Escrow",

"WireTransferSpecialInstructions”: xs:string,

"EscrowNumber”: xs:string,

"EscrowOfficerName”: xs:string

}]

}

}

### REST-JSON update Wire-transfer

REQUEST:

**<Security related header parameters... see Security Services>**

Accept: application/json

Accept-Charset: utf-8

Accept-Language: en-us

Content-type: application/json; charset=utf-8

X-API-Version: >=3.0.0

**PUT** https://api.dataprovider.com/wireTransfer

{

wireMessage: {

“messageContext”: { <see messageContext.xsd>

},

“wireFilter”: {

"occurrenceIdList": {

"occurrenceId": ["452344"}

},

"Wire”: [{

"occurrenceId": "452344",

"occurrenceAmount": {

"value": "310.89",

"currencyCode": "USD"

},

"occurrenceFromAccountId" : “1234”,

"createdDateTime" : "2014-01-01T01:01:01Z",

"WireTransferType”: "Escrow",

"WireTransferSpecialInstructions”: xs:string,

"EscrowNumber”: xs:string,

"EscrowOfficerName”: xs:string

}]

}

}

RESPONSE

Headers:

Status Code: 200 Ok

Content-type: application/json; charset=utf-8

Content-Language: en-us

Payload:

{

"wireList" : {

"Wire”: [{

"occurrenceId": "452344",

"occurrenceAmount": {

"value": "310.89",

"currencyCode": "USD"

},

"occurrenceFromAccountId" : “1234”,

"createdDateTime" : "2014-01-01T01:01:01Z",

"occurrenceConfirmationCode": "occurrenceConfirmationCode1",

"WireTransferType”: "Escrow",

"WireTransferSpecialInstructions”: xs:string,

"EscrowNumber”: xs:string,

"EscrowOfficerName”: xs:string

}]

}

}

### REST-JSON Delete pending wire-transfer

REQUEST:

**<Security related header parameters... see Security Services>**

Accept: application/json

Accept-Charset: utf-8

Accept-Language: en-us

Content-type: application/json; charset=utf-8

X-HTTP-Method-Override: DELETE

X-API-Version: >=3.0.0

**PUT** https://api.dataprovider.com/wireTransfer

{

wireMessage: {

“messageContext”: { <see messageContext.xsd>

},

“wireFilter”: {

“occurrenceIdList” : {

“occurrenceId” : [ “452344”]

},

“accountIDList” {

“accountId”: [ “1234” ]

},

}

}

}RESPONSE

Headers:

Status Code: 200 Ok

Content-type: application/json; charset=utf-8

Content-Language: en-us

# General Error handling For All Services

Refer to latest CUFX documentation *Error Mapping*.

# Bibliography

W3C. (n.d.). *Key words for use in RFCs to Indicate Requirement Levels [RFC2119].* Retrieved Sept. 8th, 2011, from W3C.