Disclaimer

The IFX Forum makes no warranties whatsoever with respect to the contents of this specification. Without limitation, the IFX Forum makes no warranty (i) that the information contained in the specification is accurate, error-free or describes a practically realizable product or service, or (ii) that the product or service described in the specification can be produced or provided without infringing third-party rights or violating applicable laws or regulations.

RESERVATION OF RIGHTS: The contents of this specification are protected by copyright and other intellectual property laws. The IFX Forum expressly reserves all rights in such content.

Table of Contents

The TOC describe the entire guide, but only selected content is available in this sample document.

In this sample document we use highlighted ellipses (...) to indicate that the Implementation Guide goes into further detail. This sample document is intended to illustrate that the guide is a robust document with valuable information about the underlying concepts of the specification and significant detail about how to use the IFX specification to implement real-world use-cases.

1 Introduction ..................................................................................................................................... 5
2 Intended Use.................................................................................................................................... 5
3 Common Vocabulary........................................................................................................................ 5
4 Web Services Concepts .................................................................................................................... 7
5 IFX Service Architecture / Framework ............................................................................................. 7
  5.1 Overall Framework ....................................................................................................................... 7
  5.2 Object Framework ........................................................................................................................ 7
    5.2.1 Object Representation ............................................................................................................. 8
    5.2.2 Object Extension ...................................................................................................................... 8
  5.3 Message Framework ........................................................................................................................ 9
  5.4 Service Provider and Service Objects ............................................................................................... 9
    5.4.1 Service Provider Object ......................................................................................................... 10
    5.4.2 Service Object ........................................................................................................................ 11
    5.4.3 Summary ................................................................................................................................ 13
6 Implementation Process ................................................................................................................ 14
  6.1 Pre-Requisites ................................................................................................................................ 14
  6.2 Business Process Flow .................................................................................................................... 14
  6.3 Service Scope Definition ................................................................................................................ 15
    6.3.1 Reusability ............................................................................................................................. 15
    6.3.2 Responsibility ......................................................................................................................... 15
    6.3.3 Transactionality ....................................................................................................................... 15
    6.3.4 Maintainability ....................................................................................................................... 15
  6.4 Determine Applicable IFX Messages ............................................................................................. 16
6.5 Service Definition

6.5.1 Status Messages and Error Handling

6.5.2 Service Versioning

6.6 Map IFX Message Content to System of Record

6.6.1 Mapping Documentation

6.6.2 Extending the IFX Schema

6.6.3 Namespaces

6.7 Define the Web Service Interface / WSDL

6.8 Generate XML Bindings

6.9 Implementation & Deployment

7 Use Cases

7.1 Create and Fund a New Account

7.1.1 Use Case Description

7.1.2 Applicable IFX Objects

7.1.3 System of Record Mapping

7.1.4 Service Definition

7.2 Online Banking

7.2.1 Use Case Description

7.2.2 Applicable IFX Objects

7.2.3 System of Record Mapping

7.2.4 Service Definition

7.3 New Card Setup Using BIAN Service Domains

7.3.1 Use Case Description

7.3.2 Service Scope Definition

7.3.3 Applicable IFX Messages

7.3.4 Service Definition

7.3.5 System of Record Mapping

8 Appendix A – IFX BMS Object Search

9 Appendix B – IFX Messages

10 Appendix C – Data Representation

10.1.1 Data Types

10.1.2 Abstract vs. Concrete Aggregates

10.1.3 Keys (xxxKeys)

10.1.4 Reference (xxxRef)

10.1.5 Selection (xxxSel)
1 Introduction
The Interactive Financial eXchange (IFX) Business Message Specification is developed and maintained as a cooperative industry effort among major financial institutions, service providers, and information technology partners to achieve a single, open financial services industry standard. It provides a comprehensive message set for developing new financial industry services and software.

The concepts in this Guide rely upon the currently published version 2.x of the IFX Business Message Specification (BMS) which is available at this URL. http://bms.ifxforum.org/rel2. In many cases, there are hyperlinks from this document to the BMS for convenient reference to details in the specification.

2 Intended Use – SAMPLE DOCUMENT
This document is an Implementation Guide for parties interested in building an interoperable software application for financial message processing, using the business messages of the Interactive Financial eXchange (IFX) Specification.

In this sample document we use highlighted ellipses (…) to indicate that the Implementation Guide goes into further detail. This sample document is intended to illustrate that the guide is a robust document with valuable information about the underlying concepts of the specification and significant detail about how to use the IFX specification to implement real-world use-cases.

3 Common Vocabulary
The IFX Business Management Specification (BMS) utilizes several terms that may have somewhat different meanings for users versed in particular Web Services standards and tools. The purpose of this section is to provide context to several terms to clarify concepts within the IFX Messaging Standard.

Within this document, these terms will be prefixed with IFX when referring to the term as defined in the IFX BMS in order to avoid ambiguity.

Object
IFX Objects can be somewhat simplistically viewed as organized sets of data of a particular type. As in any typical banking environment, the IFX Objects are subject to action in more than one service interaction. Further discussion of IFX Objects can be found in Appendix A – IFX BMS Object Search

Message
IFX Messages are defined to affect the state and content of IFX objects. The standard does not define implementation details, but IFX Messages …

To summarize, the IFX BMS defines a set of objects and messages. The messages and objects adhere to consistent design patterns that simplify understanding and provide an obvious framework for extensions. A set of messages represents an IFX Service. These services can then be realized using Web Service technologies and standards. The IFX Message Specification places no restriction on the organization of operations or services, nor on the manner in which these are implemented.
4 Web Services Concepts
This document assumes basic understanding of Web Services concepts and the artifacts involved in
design and development of a Web Service, and will draw correlation, when possible, to the W3C
Working Group Notes for Web Services Architecture (http://www.w3.org/TR/ws-arch/).

- Web Services Description Language (WSDL) – The standard for describing web services (e.g., the
  messages that are exchanged between provider and agent)
- Service Oriented Architecture Protocol (SOAP) – The standard framework for packaging and
  exchanging XML messages

The Web Services Supplement, a companion publication to this guide, offers specific guidance on using
the IFX messages in a Web Services implementation including working code examples. The Web
Services Supplement is not available in Sample Form.

<table>
<thead>
<tr>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
</tbody>
</table>

5 IFX Service Architecture / Framework
The IFX Framework consists of an Object Framework, outlining the structure and common elements of
message content and a Message Framework, which provides the structure of IFX compliant messages
consumed and produced by service implementations.

5.1 Overall Framework
The IFX Business Message Specification is designed to operate in stateless, multi-tiered, service-oriented
environments. The framework consists of Common Object Definitions with well-defined data semantics
and a Request-Response message protocol, where each message is ...

5.2 Object Framework
An IFX Object is a set of data that is organized according to a consistent pattern and supports a well-
defined set of operations. IFX Messages cause IFX Objects to be created, modified, and destroyed. IFX
Objects are constructed from basic building blocks ...
For more specific information on data typing and object representation, see Appendix C – Data Representation

5.3 Message Framework
The messages comprising the IFX Specification are listed in Appendix B – IFX Messages. The specification supports implementation of a subset of messages based on the services that are going to be offered.

5.4 Service Provider and Service Objects
A fundamental assumption underlying the IFX Specification is that one or more services are being offered by a Service Provider.

Specific services are not defined by the IFX Forum. Rather, Service Providers define Services at a level of granularity and functionality convenient to them. These may be very broad sets of capabilities or ...

The diagram below describes the relationship between clients, Service Providers, and Services in their simplest form.

- Request Messages are directed to the Service Provider by the client
- The Service Provider uses one or more Services to satisfy the request
- The client receives a response

5.4.1 Summary
The IFX Standard is based on Service Oriented Architecture.

1. A Service Provider is uniquely identifiable. IFX mandates ...
2. Each Service is uniquely identifiable within the Service Provider’s domain. This, in effect, ...
3. IFX Objects are “owned” or managed by the Service that creates or instantiates the object.
4. IFX Messages are directed to a Service Provider, who may in turn redirect the message(s) ...
5. Although it is typical for networks, mainframes, servers, and databases to be involved in supporting a service offering, IFX makes no assumptions about the technical infrastructure used to provide services.

© 2014, by the Interactive Financial eXchange Forum, Inc.
6 Implementation Process
This section provides an outline of the steps usually taken during an implementation. The document will provide more in-depth detail of the steps, while taking each use scenario from use case to implementation.

At a high level, implementing a Web Service using the IFX Message Specification can be broken down into the following steps:

1. Documenting the Business Process Flow
2. Defining the scope and intent of service(s)
3. Determining appropriate IFX Messages
4. Defining IFX Service(s)
5. Mapping IFX Message content to elements of System of Record (SOR)
6. Generating the Supporting Schema
7. Defining the Interface / WSDL
8. Generating XML Bindings

The following section provides a definition of each of these steps, and experiences of participants in the working group. The subsequent sections provide walkthroughs of realizing an IFX Service based on common use cases.

6.1 Map IFX Message Content to System of Record
It is important to understand SOR data model with its data types, required elements, and description. Data type compatibility also needs to be taken into consideration when performing data mapping, to ensure that proper type conversion is implemented when necessary, and to enforce compatibility with the SOR.

6.1.1 Mapping Documentation
Generally, these mappings are kept as spreadsheets. The resulting artifact can be used as a specification between Business Analysts and the developers. The artifact typically allows for specification of translation between the two formats (required fields, type conversions etc.). An IFX Request Message must contain all relevant data required by the SOR, so the required elements of the message must be documented. First, create a data dictionary of the SOR.

<table>
<thead>
<tr>
<th>Entity info</th>
<th>Type</th>
<th>Length</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer.Id</td>
<td>Number</td>
<td>12</td>
<td>Y</td>
<td>Primary key</td>
</tr>
<tr>
<td>Customer.FullName</td>
<td>Varchar</td>
<td>150</td>
<td>Y</td>
<td></td>
</tr>
</tbody>
</table>
Then, map the service inputs and outputs. Most implementations use an XPath style notation to represent the fully qualified data path.

<table>
<thead>
<tr>
<th>Business Reference</th>
<th>XPath element reference</th>
<th>Type</th>
<th>Usage</th>
<th>Table Column</th>
<th>Type</th>
<th>Length</th>
<th>Req’d</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Service Input</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Party Identifier</td>
<td>PartyInqReq/PartySel/PartyKeys/PartyId</td>
<td>NC-36</td>
<td>Req’d</td>
<td>Customer.Id</td>
<td>Num</td>
<td>12</td>
<td>Yes</td>
<td>Primary key</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Service Output</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Party Identifier</td>
<td>PartyInqRs/PartyRec/PartyId</td>
<td>NC-36</td>
<td>Req’d</td>
<td>Customer.Id</td>
<td>Num</td>
<td>12</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Party full name</td>
<td>PartyInqRs/PartyRec/PersonPersonInfo/PersonData/PersonName/FullName</td>
<td>C-96</td>
<td>Req’d</td>
<td>Customer.FullName</td>
<td>Varchar</td>
<td>150</td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>

There are additional aspects to consider when mapping that are beyond the scope of this document:

- Aggregation – Mapping multiple columns to one element
- Decomposition – Splitting a value within an element into multiple fields
- Validation – Validating input within elements of the request
- Translation – Translating or formatting of internal codes for abstraction, presentation or language translation
- Derivation – Creating new fields for input or output, based on the request or response message.

### 6.1.2 Extending the IFX Schema

In the process of modeling the target SOR to IFX, it is likely that there are elements within the SOR that have no direct mapping into IFX. It would be impossible for the IFX Specification to cover every aspect of a financial SOR, so there will inevitably come a time where adjustments need to be made to the schema to support unique characteristics of the underlying system. There are several strategies for handling...

### 6.1.3 Namespaces

Managing extensions to the IFX Schema requires consideration of the context in which the Specification is being used. The approach to creating or using a namespace...

### 6.2 Define the Service Interface

In this step, the interface is created that can be used to generate code stubs for implementation, and can be used by clients to create their interface to the service. See the Web Services Supplement for...

### 6.3 Generate XML Bindings

This step captures the process to transform the defined schema into object bindings for the implementation language to be used to realize the service. See the Web Services Supplement for...
6.4 Implementation & Deployment
After completing the preceding steps, all that remains is implementation and deployment of the service. These aspects of development are highly specific to each business. For reference, see the Web Services Supplement for an example...

7 Use Cases
The following Use Cases are presented as common scenarios where a Service Oriented Architecture may be leveraged in this domain. The goal is to present situations where the design fulfilling the scenario could be varied between an implementation using a set of fine-grained services or one delivering a set of composite services that combine fine-grained services.

7.1 Create and Fund a New Account

7.1.1 Use Case Description
In this scenario, an established party (customer) wishes to create and fund a new account. This scenario is executed in a branch office. The branch office Customer Service Representative (CSR) verifies the identity of the party and confirms that he/she is a customer of the financial institution. The CSR then requests the set of products that are available to the party. The party selects a product, and the CSR provides the party with a list of funding options. The party selects one of these options. The CSR completes the transaction by submitting the selections, with which the system creates the account, associates the account with the party, and executes the transactions necessary to fund the account.

The process can be depicted as shown in Figure 1.

Figure 1 - Open and Fund Account Use Case

<table>
<thead>
<tr>
<th>BUSINESS PROCESS FLOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. CSR validates existing Customer.</td>
</tr>
<tr>
<td>2. CSR presents available products.</td>
</tr>
<tr>
<td>3. Customer selects product.</td>
</tr>
<tr>
<td>4. CSR presents funding options.</td>
</tr>
<tr>
<td>5. Customer selects funding.</td>
</tr>
<tr>
<td>6. CSR adds account.</td>
</tr>
<tr>
<td>7. CSR adds Party Account Relationship.</td>
</tr>
<tr>
<td>8. CSR adds Funding Transaction</td>
</tr>
</tbody>
</table>
7.1.2 Applicable IFX Objects
In this case, the following elements would be considered entities:

Each of these entities requires unique identification and has a lifespan within the System of Record. For these entities, the IFX BMS was used to locate similar objects (see Appendix A – IFX BMS Object Search).

<table>
<thead>
<tr>
<th>Entity</th>
<th>IFX Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
<td>Party</td>
</tr>
<tr>
<td>(Financial) Product</td>
<td>ProdIntRate</td>
</tr>
<tr>
<td>Transaction</td>
<td>Credit, Xfer</td>
</tr>
</tbody>
</table>

7.1.3 System of Record Mapping
Mapping an IFX Message to the SOR data model is the process of creating logical links between IFX Message elements and distinct data model attributes.

<table>
<thead>
<tr>
<th>XML Schema elements</th>
<th>SOR model entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyRec/PartyId</td>
<td>Customer</td>
</tr>
<tr>
<td>PersonPartyInfo/PersonData/PersonName/FullName</td>
<td>FullName</td>
</tr>
<tr>
<td>PersonPartyInfo/PersonData/Contact/PhoneNum/Phone</td>
<td>PhoneNumber</td>
</tr>
<tr>
<td>PersonPartyInfo/PersonData/Contact/EmailAddr</td>
<td>Email</td>
</tr>
</tbody>
</table>

The core of the mapping document includes a map from the column to the fully qualified aggregate within IFX. Typically included are the type mapping, usage constraints, and notes related to validation or conversion between IFX and the SOR.

<table>
<thead>
<tr>
<th>Business Reference</th>
<th>XPath element reference</th>
<th>Type</th>
<th>Usage</th>
<th>Table Column</th>
<th>Type</th>
<th>Length</th>
<th>Req’d</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Input</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Party Identifier</td>
<td>PartyInqRq/PartySel/PartyKeys/PartyId</td>
<td>NC-36</td>
<td>Req’d</td>
<td>Customer.Id</td>
<td>Num</td>
<td>12</td>
<td>Yes</td>
<td>Primary key</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service Output</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Party Identifier</td>
<td>PartyInqRs/PartyRec/PartyId</td>
<td>NC-36</td>
<td>Req’d</td>
<td>Customer.Id</td>
<td>Num</td>
<td>12</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Party full name</td>
<td>PartyInqRs/PartyRec/PersonPartyInfo/PersonData/PersonName/FullName</td>
<td>C-96</td>
<td>Req’d</td>
<td>Customer.FullName</td>
<td>Varchar</td>
<td>150</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Home phone number</td>
<td>PersonPartyInfo/PersonData/Contact/PhoneNum/Phone</td>
<td></td>
<td>Required</td>
<td>Varchar</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
7.1.4 Service Definition
One way to describe a high-level service that implements particular business capabilities is to describe the interaction of the underlying technical services necessary to realize the desired result. The UML sequence diagram below represents this use case.

Figure 2 - Open and Fund Account Message Sequence
Appendix A – IFX BMS Object Search

This appendix will help the reader locate a BMS Object that may be abbreviated or just have a different, but synonymous, name for the topic being researched. The online BMS is a good tool for doing research but, if the topic has been abbreviated in the BMS, the topic may be difficult to locate. If the reader is a member of the IFX Forum, a pdf file containing all of the BMS documentation is available for download from the Architecture area of the members-only website. The pdf can be used to search for a keyword that will point to an object of the BMS. This appendix is valuable if the pdf is not available. To use this appendix effectively, search using a keyword such as Mortgage, Payment, Statement, etc.

<table>
<thead>
<tr>
<th>BMS Object Name</th>
<th>Keywords</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acct</td>
<td>Deposit Account, Loan Account, Certificate of Deposit, CD, Time Deposit, Credit Account, Rewards Account, Mortgage Account, Secured Loan Account, Unsecured Loan Account.</td>
</tr>
<tr>
<td>AcctAcctRel</td>
<td>Sweep, Interest Distribution, Overdraft, Notional Pooling.</td>
</tr>
<tr>
<td>AcctHold</td>
<td>Account Hold, Permanent Hold, Check Hold, Court Order Hold, Restriction, ACH Hold, Authorization Hold, Collateral Hold, Secured Account Hold.</td>
</tr>
<tr>
<td>Trn</td>
<td>Transaction, Credit, Debit</td>
</tr>
<tr>
<td>Xfer</td>
<td>Transfer, Credit, Debit</td>
</tr>
</tbody>
</table>

Appendix B – IFX Messages

IFX Message names adhere to a consistent pattern – ObjectActionDirection – where Object is the name of the IFX Object that will be acted on, Action is one of the “verbs” (or methods) shown below and Direction is either Rq (Request) or Rs (Response). It is not necessary to implement...

Two special cases are noteworthy: Authorization and Status. The IFX Specification facilitates inquiry and modification of the Authorization and Status aggregates associated with a specific object by implementing a Record construct that can be targeted by a message. For example, ...

<table>
<thead>
<tr>
<th>Name</th>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Add</td>
<td>The xxxAdd Messages support creating a new instance of the specified xxx object.</td>
</tr>
<tr>
<td>Advise</td>
<td>Advise</td>
<td>The xxx Advise Message is used to notify interested parties that an xxx object was created or modified. The Advise Message set does not...</td>
</tr>
<tr>
<td>Aud</td>
<td>Audit</td>
<td>The xxx Audit Message supports the ability for the client to trace the message history...</td>
</tr>
<tr>
<td>AuthInq</td>
<td>Authorization Inquiry</td>
<td>The xxx Authorization Inquiry Message is used to request an inquiry of authorization information for the specified xxx object.</td>
</tr>
<tr>
<td>AuthMod</td>
<td>Authorization Modification</td>
<td>The xxx Authorization Modify Message is used to change authorization information for the specified xxx object.</td>
</tr>
<tr>
<td>Can</td>
<td>Cancel</td>
<td>...</td>
</tr>
</tbody>
</table>
Appendix C – Data Representation

9.1.1 Data Types
IFX uses common primitive data type representation, as well as definition for types commonly used within IFX Objects (Date, Time, Phone Numbers etc.) Please see the BMS section 3.3 for detailed description of the supported data types.

9.1.2 Abstract vs. Concrete Aggregates
Abstract Aggregates were introduced for ease of modelling and building the BMS. Abstract Aggregates contain characteristics common to groups of data elements. An Abstract Aggregate will never appear ...

9.1.2.1 Aggregate Extension
Non-abstract Aggregates can also be extended to create a more robust model. Extensions of an Object retain the structure of their base object, but ...

9.1.3 Keys (xxxKeys)
The Keys Aggregate contains a set of attributes that, when used together, form a unique identifier of the object. In other words, the Keys of the object can be used to locate a unique/single occurrence of an object in a datastore. The Keys Aggregate may ...

9.1.4 Reference (xxxRef)
A Reference Segment is an indirect reference to another object. On the wire, it will usually contain the ID or Keys of another IFX Object, but ...

9.1.5 Selection (xxxSel)
A Selection Aggregate is an indirect reference to a collection of objects. On the wire, it will usually contain object-supported selection criteria for the target object. The Selection Aggregate is not meant for ad hoc reporting, but rather inquiries ...

< xxxInqRq >
  ...
  < RecSelect >
    (/DebitRec/DebitInfo[DebitType=CashWithdrawal] |
     /DebitRec/DebitInfo[DebitType=CreditCardAdvance]) &
    /DebitRec/DebitInfo/CompositeCurAmt[CompositeCurAmtType=Debit & CurAmt >
     =100.00]/CurAmt
    < /RecSelect >
  < /xxxInqRq >