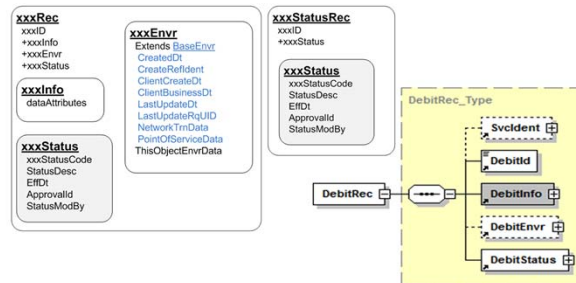




IFX Object Fundamentals

Introducing the Objects Data Design Pattern



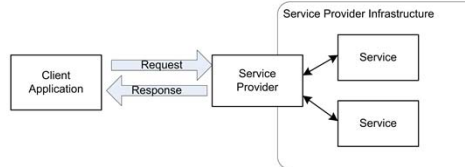
Welcome to the continuing series of IFX Orientation presentations.

In this presentation we will begin to dig a bit deeper into the IFX Object Model.

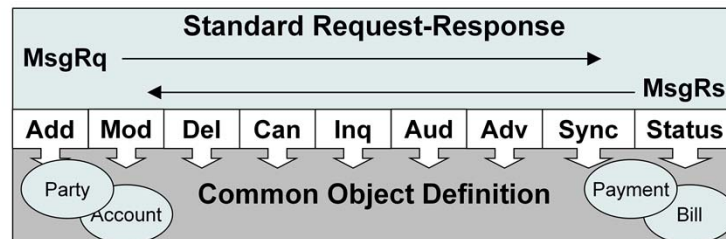
IFX Messages exchange data

The fundamental underlying principle of the IFX Framework is this simple:

Service Providers offer Services to Clients.



- The IFX Message Specification standardizes a means of exchanging data and managing the state and content of data objects



In other presentations we have discussed the underlying principle of the IFX Framework : Service Providers offer Services to Clients and that the IFX Message Model provides a mechanism to exchange data between service providers.

We also discussed the fact that IFX Messages are always addressed to a specific type of object – and usually act on a single, specific instance of an object.

With that background we can now delve into how data is structured in the IFX standard to facilitate consistent representation of – and effective management of – financial data .

In the IFX framework, data is consistently represented as objects, and all objects are structured according to well-defined rules.

So let's dig in!

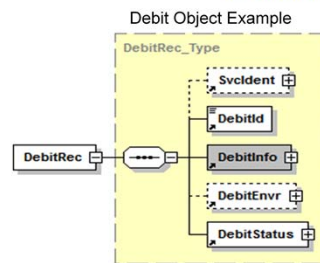
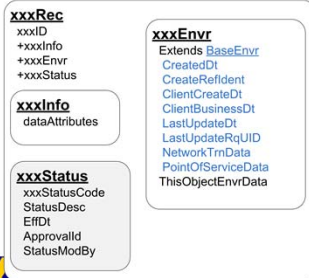
IFX Objects are data

Key Concept

- All IFX Objects adhere to a basic pattern
- This pattern is enforced in the Standard Repository (BMS DB)
- The BMS DB can generate schema for an object for immediate use in your code

Object Elements and Structures

[xxxAuth](#)
[xxxAuthRec](#)
[xxxEnvr](#)
[xxxD](#)
[xxxInfo](#)
[xxxKeys](#)
[xxxRec](#)
[xxxRef](#)
[xxxRsMsg](#)
[xxxRgMsg](#)
[xxxSel](#)
[xxxStatus](#)
[xxxStatusRec](#)
[xxxUpdElements](#)
[xxxNewElements](#)



This slide shows several ways to visualize IFX Objects.

1. On the upper right portion of the slide is a complete list of elements that will be found in IFX Objects. (the xxx represents an object name)
2. The lower right shows how an object looks in an XML Schema; and
3. The lower left shows a container hierarchy to help visualize data 'packaging'

There is a lot of detail packed into this slide and by the end of this presentation you should find that you have a pretty solid understanding of how it represents these key facts:

- Every IFX object adheres to this exact same design pattern
- The names of the data elements in every object are constructed exactly the same way. You have only to substitute the name of the object for the xxx pattern in these lists to arrive at the exact tag name for specific group of data elements. (Notice how the word "Debit" is combined with the element name in the schema example.)
- This consistent application of rules and patterns makes it easy to extend the standard and makes it clear where your data extensions will reside
- In the same spirit, this consistency shows up in the schema generated from the BMS making it easy for developers to find data and work with the standard.

Types of Objects - examples

Type	Example(s)
People, Organizations, Roles	Party, Person (Party), Org (Party), Biller
Things	Card, (ATM) Device, Bill
Transactions	Payments, Transfer, Debit, Credit
Relationships*	Party-to-Party, Party-to-Account
Financial records	Account, Passbook

* IFX often represents the relationships between objects as objects in their own right. Such objects always name the things being related and appends "Rel" to signify the object represents a relationship (e.g. PartyAcctRel).

This applies whenever related objects have persistent attributes (e.g. account owner,) or have many-many relationships (e.g. party has many accounts).



Before we get too deep into the object structure, let's take a quick look at how the IFX Object model maps to the real world.

The table is mostly self explanatory. In the left-hand column are types of objects typically found in the financial services domain. On the right are some specific examples. (By the way, the IFX implementation guide – available to members and subscribers – actually provides a fairly detailed list of financial objects, their synonyms and the IFX object used to represent them.)

It's worth a special note to recognize that 'Relationship' objects adhere to all of the same rules and patterns as other IFX objects. The naming of relationship objects requires us to name the 2 object types being related and, for clarity, we add the suffix "REL" to indicate the object contains data about the relationship rather than the objects themselves. Data base modelers will recognize this pattern as an intersection table in 3rd normal form data models which also supports many-to-many relationships.

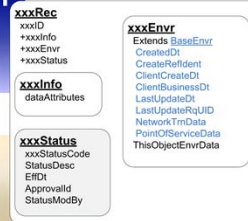
What is an IFX Object?

Key Concepts

- A set of naturally related data attributes that can be uniquely identified
- A normalized, implied data structure to which you map your own physical representations of data

Implications

- IFX does not define the physical format, actual record structure, storage technology, etc
- A server must be able to assemble a proper representation of the object as described in the specification in order to respond to IFX messages.



So what is an IFX Object? Essentially, an IFX object is **an implied record structure** that represents a set of well defined data usually stored as a unit on a server with persistent storage. **It is not assumed that the object is stored exactly as shown in the object model.**

IFX defines the data contents necessary to represent the object, but IFX does not define the physical format, actual record structure, storage technology, etc.

This is an important point, because it is at this juncture that you may recognize that you can and must map your own physical data structures to the IFX object model in order to ensure smooth implementation of the standard.

The IFX assumption, therefore, is that **a server is able to assemble a proper representation of the object.** This capability is necessary of course in order to respond appropriately to all messages about an object.

Core Data Aggregates

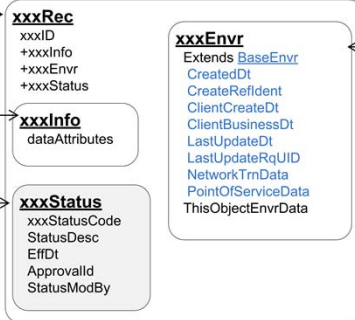
Key Concept

- Groups of closely related data elements are represented in the IFX specification as 'Aggregates'
- Every IFX Object has the same set of core aggregates

The Object Record aggregate contains all of the object data.

The Object Info aggregate contains all of the object data naturally managed by applications and users

The Status segment contains the current state of the object



The Object Envr aggregate contains environmental data that is fixed by the server



© Copyright IFX Forum, Inc. 2015

Slide 6

Groups of closely related data elements are represented in the specification as 'Aggregates'

The core aggregates of every IFX object include:

- The **Info** segment which contains the properties of the object that are directly maintainable by a client
- The **Envr** segment which contains environmental data that is fixed by the server
- The **Status** segment which contains the current state of the object.

There are other important core aggregate elements as well, but understanding this set of **mandatory** aggregates is sufficient for now.

Object Identification

Key Concept

- IFX Objects have unique ids – always named xxxID; always the same simple data type (NC-36)
- IFX Objects support alternate identifiers such as business keys – always itemized in the xxxIDENT aggregate
- Each object is 'owned' by the service that creates it (named in SvclIdent) and the creator assigns the unique ID

Implications

- The ID is assigned by the service provider (SP) and the SP is always considered to have the 'official' state of the object, though others may have copies
- The object id is unique within the service provider's domain.
- IDENTs take on many forms and often are consistent with other standards and conventions (e.g. IBAN, UPIC, CardMagData, Account Numbers, etc)



IFX assumes that each object is uniquely identified by its owner. In the IFX Framework, the **service** that creates an object **assigns the object ID** and is considered the '**owner**' of the object.

As a practical matter, this may simply mean that the Service Provider uses an arbitrary database key as the object ID. But the model supports more interesting use-cases as well. For example, a financial institution may have subsidiaries with specialized service capabilities, or rely upon outsourcing vendors for services; or specific servers within a bank infrastructure can be identified as Service Providers.

The SvclIdent, which acts as a qualifier for IDs, also ensures that messages can be routed back to the owner to ascertain an object's status and that the context for the unique id is clear.

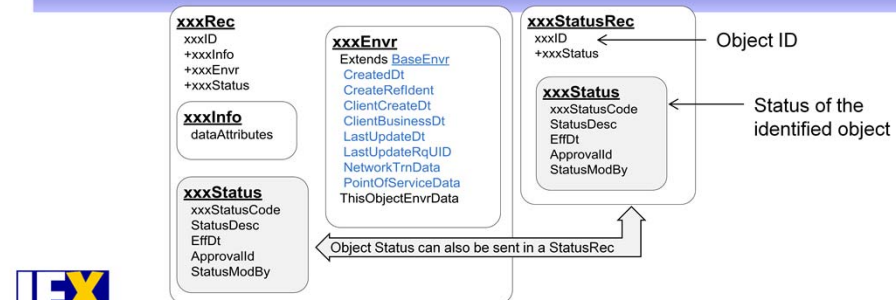
Finally, any object may have alternate keys such as account numbers, card ids and such. IFX supports and identifies these as optional attributes of the object's IDENT structure. As the name implies, these are identifying attributes but they are distinct from the ID assigned by the service that created the object.

One final note, there is actually no prohibition against using a business key such as account number as the object id; the only requirement is that the SP can ensure it is entirely unique within its domain.

Object Status

Key Concepts

- IFX Messages cause actions to be taken on objects – effectively changing the state of the object.
- The current state of an object is found in the Status aggregate
- The Status segment can be independently serialized and transmitted in a Status Record



© Copyright IFX Forum, Inc. 2015


Slide 8

Every persistent object in IFX is required to maintain Status.

Importantly, the standard supports direct communication of an object status through the StatusRec.

This allows direct inquiry of object status and a simple mechanism to return the status in response to such an inquiry. As shown here, it is a simple matter of including the object's Status aggregate in a StatusRec along with the object ID.

Objects in the BMS repository



DB 2.x

Home

List IFX Objects

List Response Codes

List Interfaces

List Operations

List IFX Tag Names

Lookup Tag by name:

Download XSD

List Document Pages

List Changed Tags

List Tag Review History

List Tag Review

Comments

Editor

Download DB Snapshot

Edit Context

Change Password

Logout

© IFX 2000-2013

IFX Object - Debit

Short Description - Debit

Description -

A debit can either be performed by itself or require a prior authorization using the object.

For a complete discussion of Credits, Debits and Authorizations refer to [Debit and Credit Objects in IFX](#).

[Follow this link to edit the descriptions](#)

Object Elements and Structures

DebitEnvr
DebitId
DebitInfo
DebitKeys
DebitRec
DebitRef
DebitRstMsg
DebitRqMsg
DebitSel
DebitStatus
DebitStatusRec

Object Methods

Add	Rq	Rs	Status Codes
Advise	Rq	Rs	Status Codes
Aud	Rq	Rs	Status Codes
Can	Rq	Rs	Status Codes
Inq	Rq	Rs	Status Codes
Mod	Rq	Rs	Status Codes
Rev	Rq	Rs	Status Codes
StatusMod	Rq	Rs	Status Codes
Sync	Rq	Rs	Status Codes

© 2000-2013 IFX Forum, Inc. All rights reserved.

Version: 2.x

The BMS data base (repository) facilitates easy navigation of the IFX standard objects and messages.

It also creates and enforces all of the base pattern requirements when a new object is defined.

Now that we have a good understanding of the underlying concepts of IFX objects – their fundamental structure, how they represent data, how they are identified, how data processing in response to messages impacts object state, and how that state can be communicated, we can examine how the IFX BMS Data Base represents them and how to navigate the UI.

In the menu on the left side of this page, you can select “List IFX Objects” and then choose any object of interest.

Shown here is a straightforward example of an IFX Object – the Debit object. Note that in addition to itemizing the data structure this page also shows exactly which methods, or messages are implemented in standard for this object. It’s also an easy matter to navigate to any object data or related message from this point, but you are likely to find that you frequently jump to the Info segment where all of the user managed data resides.

Conclusion

There's a lot more to be learned about the object model and data representation in the IFX Specification. We have only covered the fundamentals here.

Look for other orientation materials to learn more about

- Other common data structures such as xxxRef, xxxAuth
- How to use the RecSelect, FieldSelect and IFXPath to scope inquiries and limit the data elements being impacted by message.
- The role of AbstractAggregates in the IFX Specification
- Details about various types of usage – Required, Optional, Repeating, OR-blocks and XOR blocks



© Copyright IFX Forum, Inc. 2015

Slide 10

There's a lot more to be learned about the objectmodel. We have only covered the fundamentals here.

Look for other orientation materials to learn more about these topics.

Additional Resources

There is much more to be learned using the resources shown below.

Direct Inquiries To		
Organization	Name	
IFX Forum	Judith Vanderkay, Director, Public Relations	info@ifxforum.org
IFX Community Forum	http://www.ifxforum.org/standards/discussion/	Available to Guests, Subscribers and Members

Reference Documents		
Topic	Type of Document	Reference to Document
IFX Standards	IFX Standard Online	http://www.ifxforum.org/standards/standard/
IFX SOA Implementation Guide	Sample content freely available	http://www.ifxforum.org/standards/implementation_guide/
Complete guide available to members and subscribers		



© Copyright IFX Forum, Inc. 2014

Slide 11

Thank you for your attention, your interest and your support of the IFX Standard.

More information can be found at these locations and at www.ifxfourm.org.