Implementing BIAN Service Domains using the IFX Business Message Specification

BIAN & IFX Forum, Inc.
Proof of Concept Report

A Joint Effort of Banking Industry Architecture Network and IFX Forum, Inc.

December 2013

DISCLAIMER

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

RESERVATION OF RIGHTS

The contents of this report are protected by copyright and other intellectual property laws. IFX Forum, Inc. and BIAN expressly reserve all rights in such content.

Copyright

Table of Contents

Diagrams and Illustrations

In order to provide a readable narrative and workflow, the body of this report is presented entirely in portrait page orientation. In the interest of presenting diagrams and illustrations in a more readable form for detailed reference, some diagrams are repeated in the appendices in full-page landscape rendering.

Management Summary

The collaboration of BIAN and IFX on this Proof of Concept (POC) was initiated with the intent to prove:

- that BIAN-defined service domains could be mapped to, and implemented using, a pre-existing service oriented messaging standard – in this case, the IFX Business Message Specification; and,
- that the IFX service framework could be used to implement a pre-existing view of standard business services – in this case BIAN service landscape and service domains.

These concepts, once proven, can be viewed more generally to conclude that either standard could be utilized within any banking infrastructure and set the stage to provide many benefits of standardization.

The underlying assumption for this proof of concept is that a bank with a desire to implement solutions based on a Service Oriented Architecture (SOA) must do so within the context of existing IT systems, business procedures and technology governance. Of necessity, then, adoption of SOA solutions is likely to be incremental and evolutionary. In order to be successful, the SOA solution(s) must not disrupt existing systems and must be flexible enough to adapt to the bank's environment while at the same time moving the systems forward with respect to standardization and value-added functionality. Otherwise, the benefits of such implementations will be questionable if not completely negated.

In this report we will demonstrate that the BIAN and IFX standards can be mapped effectively to the bank's current IT landscape when necessary, and that they can be used together to achieve breakthrough efficiencies in the software development lifecycle.

Looked at from the view of the BIAN standard, we will demonstrate that BIAN service domains can be aligned with existing business capabilities and implemented using the bank's current SOA technology.

Viewed from the perspective of IFX messaging, we will demonstrate that the standard can be adapted to implement services as defined within the bank's own view of service definitions.

Most importantly, though, we show that banks can achieve much greater value using both standards as points of reference. Banks can begin to optimize their business capabilities both strategically and tactically in such a way as to:

a) accelerate the development lifecycle for new offerings from concept to implemented capability by utilizing the standards;
b) clearly define the scope of services offered to, or provided by, external partners;
c) isolate legacy systems from external changes by defining them as services within the overall services landscape.

About IFX Forum and BIAN

IFX Forum and BIAN are two industry-leading Standards Development Organizations addressing Service Oriented Architecture (SOA) in financial services. The two organizations have complementary solutions to many of the architectural challenges facing the banking industry. Both organizations address areas where banks can gain significant economies and efficiencies by deploying standard, interoperable solutions.

- The IFX standard is designed to meet the business requirements of the global financial services industry. IFX is usable in many types of business and operating environments; extensible to cover many types of transactions; and global in perspective, leveraging broad participation from Forum members around the world. The Forum promotes interoperability of industry standards by working cooperatively with other standards organizations and consortia. The IFX Forum has demonstrated its commitment to work with other standards development organizations, including ANSI X12, ASCX9, ISO 20022, BIAN and others.
- The success of BIAN, the Banking Industry Architecture Network, mirrors the increasing importance of standard software in a maturing industry. The BIAN services landscape is a blueprint for the logical components of a bank's IT environment. Leveraging this blueprint can significantly accelerate architecture initiatives – be it in the planning of change initiatives, in the procurement of components, or the benchmarking of an existing landscape against best practices.

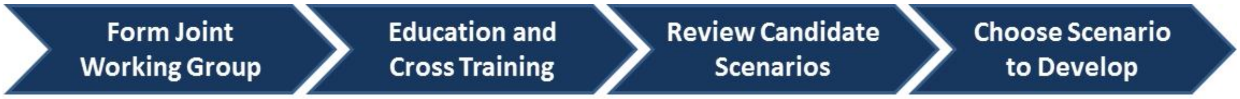
BIAN and IFX Forum have strength and expertise in different areas. BIAN's main focus has been on the business services and the services landscape, whereas IFX Forum's main focus has been on financial services messaging, message components and operational methods.

Both frameworks are technology-neutral, which should be complementary when used together to provide a powerful, standardized, interoperable solution for financial institutions. Therefore, both organizations believe that the possibility to leverage both standards could create an impact in the banking industry and could bring mutual benefits to both organizations, now or in the future.

1 About the Proof of Concept

1.1 Setting the Stage

In June 2013, BIAN and IFX Forum brought together experts from both organizations with the charter to demonstrate that the work each organization does in defining standards for service oriented solutions in the banking industry is complementary, compatible and valuable.

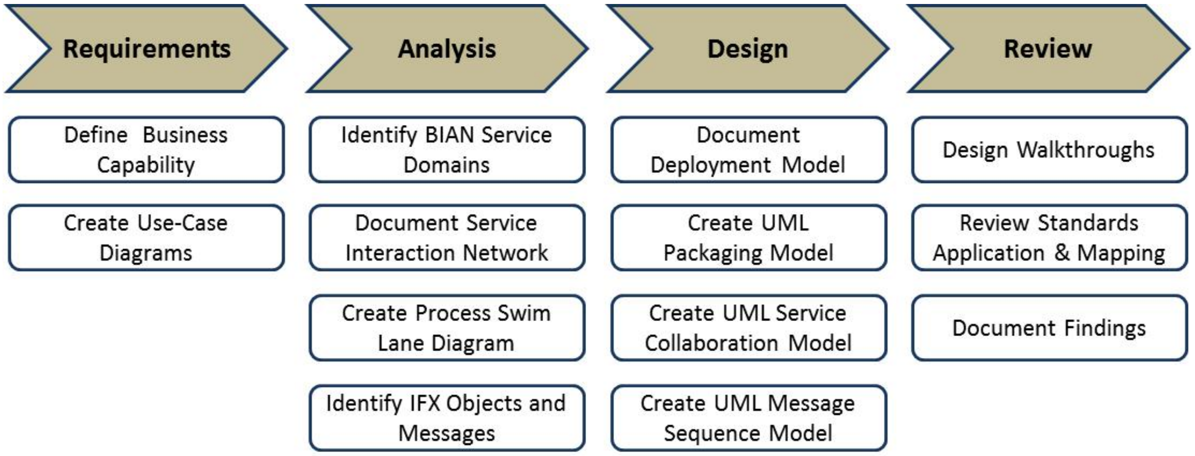

Form Joint Working Group
Education and Cross Training
Review Candidate Scenarios
Choose Scenario to Develop

At that time it was decided that the best approach to accomplish the goal was to design a solution for a real-world business capability, referencing both standards as part of the solution and providing sufficient detail to prove that credible implementations could be realized.

The members of the work group shared subject matter expertise about the standards and each group's respective approach to service oriented architecture. This was followed by a series of detailed discussions of the best candidates for demonstration. Ultimately a prime candidate was chosen for more detailed design and illustration. Specifically, the process chosen was *New Card Setup,* which occurs when a new prospect applies for a credit card from the bank.

1.2 Step-by-step

With a business scenario chosen, it became a straightforward matter to plan the tasks that the work group would need to accomplish, treating it as a typical challenge faced every day in IT departments in any typical bank. The final review steps are specific to the nature of this Proof of Concept.


Requirements
Define Business Capability
Create Use-Case Diagrams
Analysis
Identify BIAN Service Domains
Document Service Interaction Network
Create Process Swim Lane Diagram
Identify IFX Objects and Messages
Design
Document Deployment Model
Create UML Packaging Model
Create UML Service Collaboration Model
Create UML Message Sequence Model
Review
Design Walkthroughs
Review Standards Application & Mapping
Document Findings

Given the requirements for *New Card Setup*, we intentionally went on to design a solution using BIAN Service Domains and the IFX Business Messaging Standard as given constraints.

1. We itemized the particulars of the common banking business capability – *New Card Setup* – for implementation. The scenario involves product offerings, customer setup, account setup and various documentation and communications.
2. We described a typical, step-by-step set of processes often used by banks to provide such a service.
3. We created UML Use-Case diagrams of the capability.
4. We aligned the capability process steps to the BIAN service domains.
5. We identified the IFX objects involved in the processes.
6. We identified the IFX messages that would be required to implement the desired functionality.
7. We defined a typical deployment model in order to illustrate the capacity of the two standards to be used in a distributed environment and/or collaboratively across company boundaries.
8. We describe the message semantics for each step of the process in UML sequence diagrams.
9. We conducted a variety of reviews and captured key concepts of how to map capabilities and implementation techniques between the two standards.

When this work was complete, we reviewed the discoveries made during the POC and summarized our findings. We identified the challenges we encountered and offer advice to the reader about how best to address them. We provide the benefit of our learning in a manner that we hope will smooth the path for future implementers of both standards. Last, but not least, our conclusions are being shared with the respective work groups and architecture committees in BIAN and IFX in the interest of continuous improvement for both standards.

No two banks are exactly alike. Therefore, we have chosen to diagram two different use-case models typical of variations that might be found in different banks.

> ***No two banks are exactly alike. For the POC to be meaningful, we concluded that it would be important to illustrate that the BIAN and IFX standards could be adapted to more than one implementation model of the business scenario.***

Following naturally from that approach, we illustrate how the collaboration between services can be partitioned differently and messages sequenced in accordance with the different models.

Ultimately, the approach taken for this proof of concept was not markedly different from the approach any bank might take to provide or improve upon a given business capability. It involved all typical business analysis and software development lifecycle steps (short of physical implementation and testing), as follows.

- Define the requirements
- Document use-cases for a business capability
- Identify existing capabilities
- Consult the enterprise architects and other subject matter experts for design guidance
- Design a solution that creates the capability and can be implemented within the existing environment.

The parallels to standard business and technology practices are important. Beyond the steps in the Design Review Phase that are specific to reporting on the Proof of Concept, the activities we engaged in are not specialized to either standard and are typical of the skills exercised in any IT department in any bank.

2 Proof of Concept

2.1 The Business Scenario

The scenario we chose to illustrate the viability of using the two standards to create a business capability is outlined below:

1. A new prospect applies for a credit card from the bank. The branch sales representative initiates the offer process, first checking that this is indeed a new customer.
2. Next, the details provided by the prospect are used to initiate a credit check.
3. A customer master agreement is completed, followed by a product-specific terms and conditions agreement that includes making the necessary disclosures and eligibility checks.
4. The offer is accepted and the new card facility setup is initiated.
5. The underlying customer accounts are set up and a credit card ordered.
6. The prospect is captured as a new customer and a welcome pack is later sent with the supporting documents and guidelines for the customer.

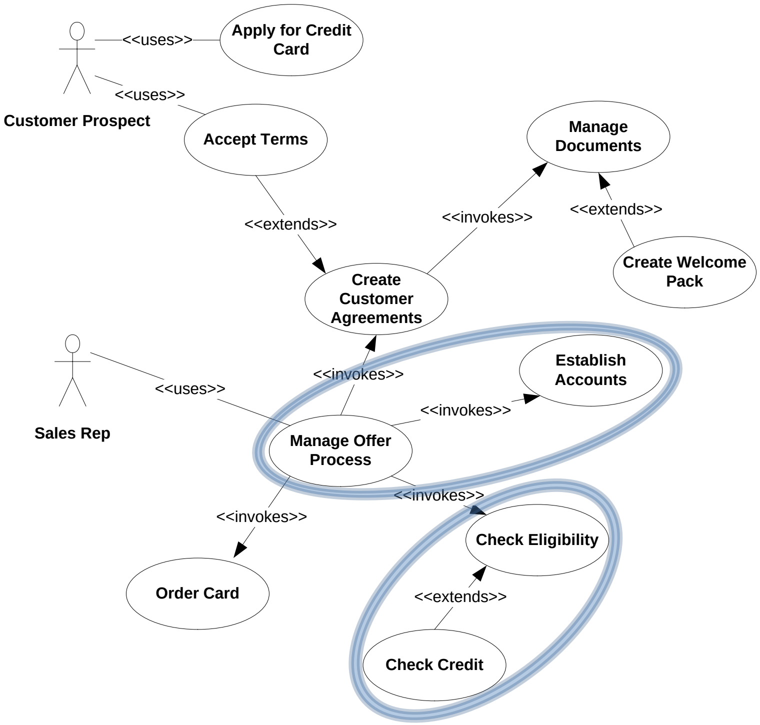
2.2 Use Case Models

No two banks are identical. Therefore we have chosen to diagram two implementation models.

2.2.1 Alternative 1

This diagram illustrates one possible view of the business scenario. In this model the sales rep invokes a single process -- the *Manage Offer Process*.

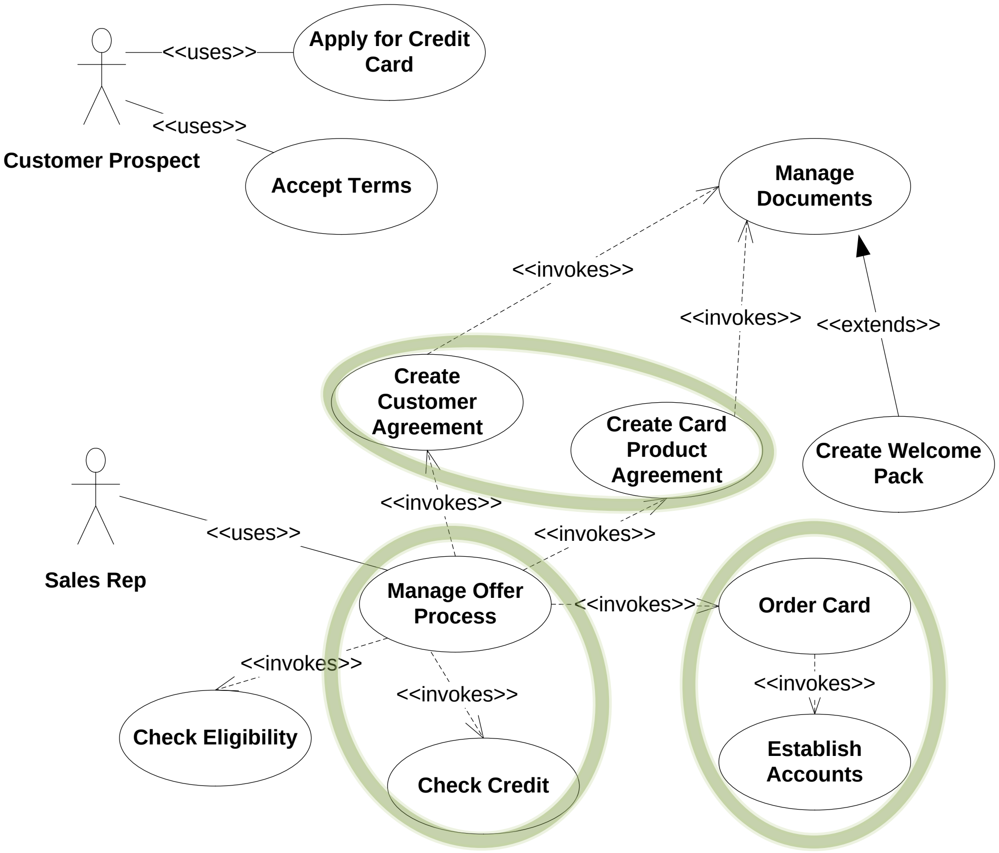
Manage Offer Process directly invokes several other processes including *Establish Accounts* and *Order Card*. *Check Credit* is viewed as an extension of the overall Check Eligibility process.



2.2.2 Alternative 2

In this alternative the activities are broken down to a finer level of detail, corresponding more closely to the "elemental" service domains that BIAN seeks to define. By defining the lowest level business capabilities, BIAN hopes to achieve canonical partitions (business capabilities that can be consistently interpreted by any bank).

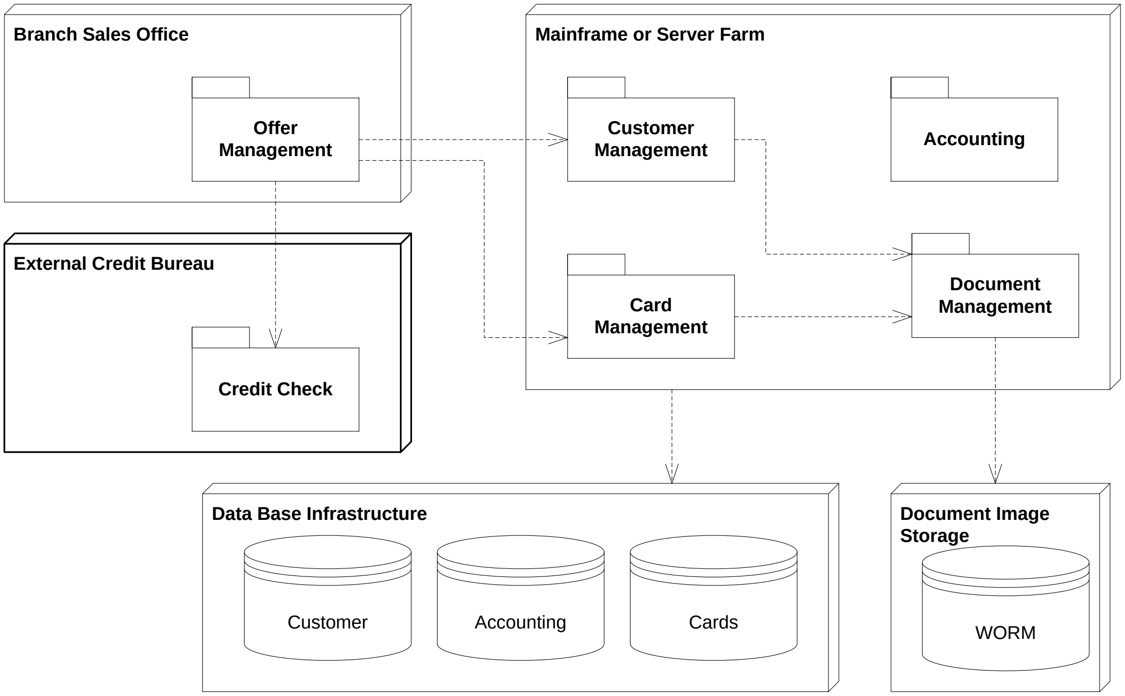
Again the sales rep invokes a single process -- the *Manage Offer Process*. However, in this model *Check Credit* is a separate service not viewed as an extension of *Check Eligibility*. Also, *Card Facility* manages the account setup and card order. Lastly, this model shows that there are separate Customer and Card Product Agreements.



2.3 Execution Environment

We assume a somewhat typical configuration of hardware and software packaging as follows. This deployment model will vary from bank to bank, some using a more centralized mainframe-centric deployment architecture, others a more distributed client-server architecture.

Although message routing and other processes will be affected by the physical environment, this design is not directly concerned with those considerations, so alternatives will not be shown.



2.4 BIAN Service Domains

The BIAN Service Domains define capabilities with implied boundaries**.** Each domain's service operations define available Interfaces – or services – available to the "outside world". The BIAN Standard does not define which service domains may call upon the services of other service domains nor does it prescribe how such collaborations are to be implemented.

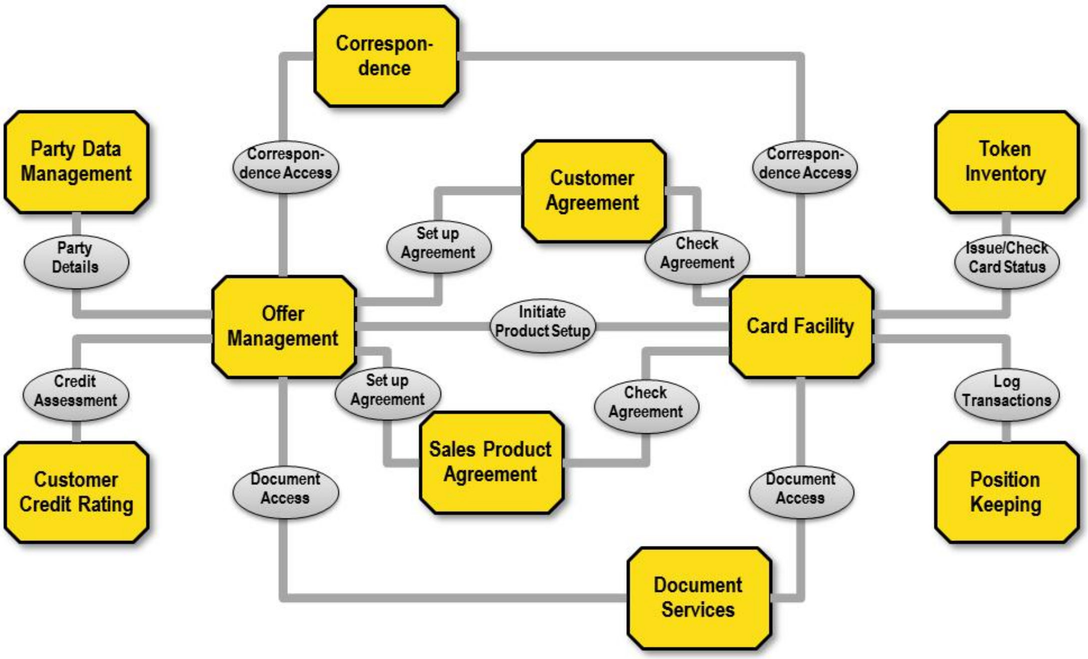
For each service domain BIAN also defines an associated **control record**. The control record describes functional activities for particular entities that are within the scope of the service domain. Control records describe the different state and state transitions that an instance of the object or entity passes through as the BIAN Service Domain's function is executed through the complete life cycle.

More details can be found in Appendix A.1.

2.4.1 Identifying BIAN Candidate Service Domains for this Business Scenario

Based on prevailing behaviors, modeled using business scenarios, the typical connections or collaborations between domains can be represented in a network diagram. The diagram below illustrates the service domains identified for this use case. The rectangular boxes represent BIAN Service Domains; the ovals characterize the service dependency between the domains. Many other connections between the domains are also supported but are not shown here as they are not pertinent to this use case.

As would be expected, the Offer Management and Card Facility service domains are the most heavily involved in this use case.

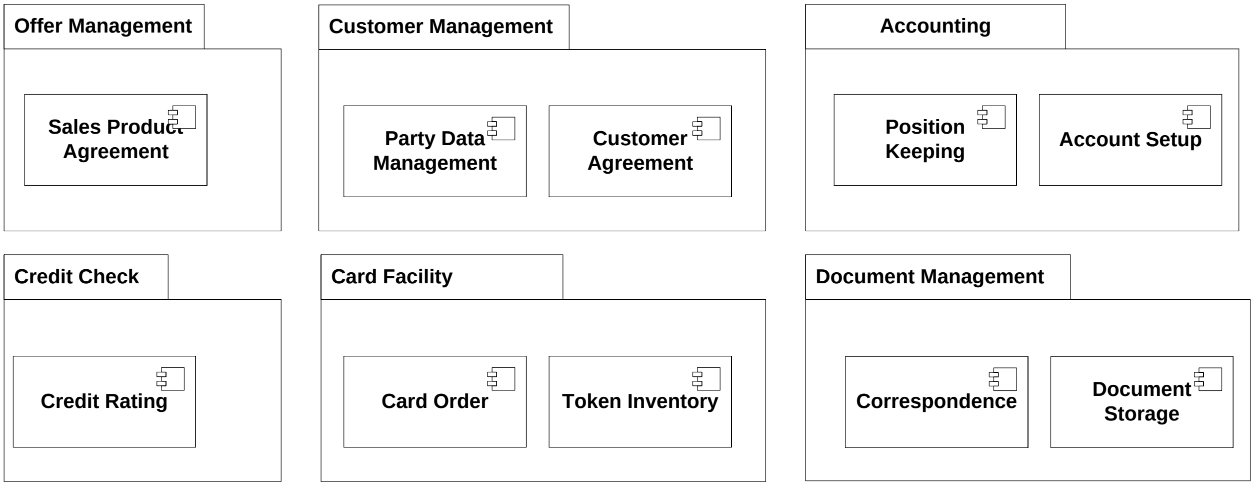


2.4.2 Capability Packaging

Having identified the service domains involved in our scenario, it is possible to create a packaging model that fits within the implementation environment.

The BIAN design approach to service domains is intended to identify elemental business capabilities that enable canonical partitioning. By limiting the scope of a service domain to perform a particular function on a particular entity, the BIAN standard implicitly defines service domain boundaries. However, this does not govern how to package the capabilities (service domains) for convenient deployment and on-going maintenance.

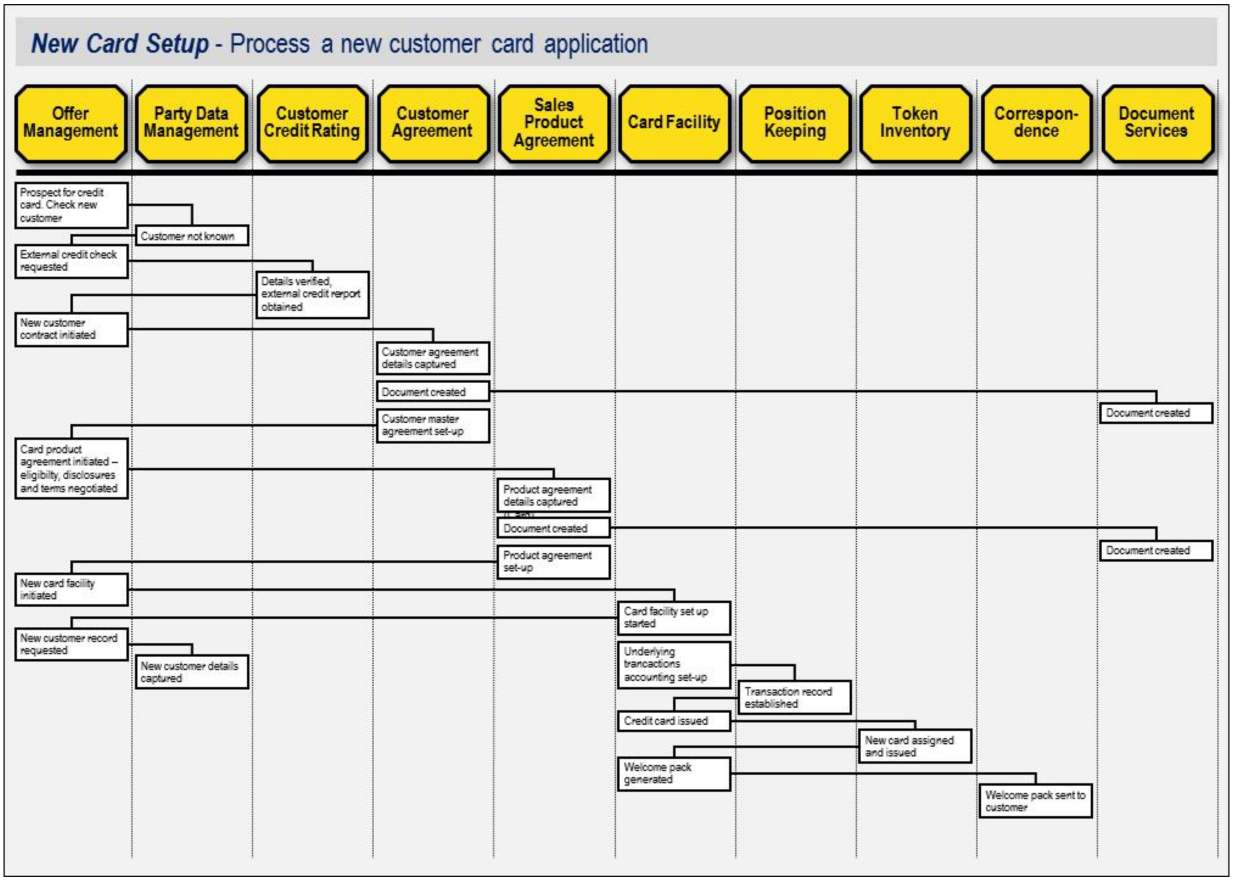
The packaging strategy for these capabilities and functionality within a particular bank may be composed differently than we have illustrated here, but notice that all of the BIAN service domains are represented. This packaging strategy is particular to the deployment model we assume for the POC.



2.4.3 Organizing the Collaboration between Service Domains

The BIAN standard does not define how service domains call upon the services of other service domains or how such collaborations are implemented. This allows implementers to model the interaction in a manner suitable to the business practices of their organizations.

The following diagram illustrates the *New Card Setup* scenario's activities as collaborating service domains defined in the BIAN service landscape. As can be seen, most of the interaction between service domains involves the *Offer Management* and *Card Facility* domains in this model. Of the two Use-Case alternatives presented in Section 2.2, this aligns most closely with Alternative 2.


New Card Setup - Process a new customer card application
Offer Management
Party Data Management
Customer Credit Rating
Customer Agreement
Sales Product Agreement
Card Facility
Position Keeping
Token Inventory
Correspon-dence
Document Services
Prospect for credit card. Check new customer
Customer not known
External credit check requested
Details verified, external credit rerport obtained
New customer contract initiated
Customer agreement details captured
Document created
Document created
Customer master agreement set-up
Card product agreement initiated – eligibilty, disclosures and terms negotiated
Product agreement details captured
Document created
Document created
Product agreement set-up
New card facility initiated
Card facility set up started
New customer record requested
Underlying trancactions accounting set-up
New customer details captured
Transaction record established
Credit card issued
New card assigned and issued
Welcome pack generated
Welcome pack sent to customer

2.5 IFX Objects and Messages

An IFX object can be visualized as a set of data organized according to a consistent, repeatable pattern. IFX Objects are constructed from basic building blocks of data Elements and data Aggregates, where elements are single pieces of information with defined data types and aggregates are groups of related elements identified by a single name for convenience.

IFX objects support a well-defined set of operations (or methods) which cause IFX objects to be created, modified and destroyed.

More details can be found in Appendix B.

2.5.1 Identifying the IFX Objects

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.

It is a straightforward exercise to discover IFX objects involved in a use-case. It is also a simple matter to associate IFX objects with the BIAN entities that are the subject of defined services. Using typical business and systems analysis techniques we identified the following business objects and located them in the standards hierarchy of each standard.

BIAN Domain	BIAN Control Record	IFX Object	Data Attributes
Party Data Management	Party REGISTRATION	Party	Close alignment as defined
Offer Management	Customer Offer EXECUTION	Offer	Close alignment as defined
Customer Agreement	Customer AGREEMENT	Party	There is a "customer established date" data attribute in IFX PartyInfo. See also Document Services and Correspondence below.
Sales Product Agreement	Sales Product AGREEMENT	Party	Similar to Customer Agreement.
Customer Credit Rating	Customer Credit QUALIFICATION	Acct	IFX links the rating to a specific Account for a customer
Card Facility	Credit/Charge Card FULFILLMENT	CardOrd	IFX maps different Card fulfillment capabilities to the card entity. See also Token Inventory and Object Relationships below.
Position Keeping	Transaction Record TRACKING	Acct	Maps to the Acct entity of IFX.
Token Inventory	Token INVENTORY	Card	IFX handles the card inventory as a property of the card capability overall
Document Services	Document Handling OPERATION	Party	IFX can capture form images, metadata and other agreement details in the Correspondence aggregate.

BIAN Domain	BIAN Control Record	IFX Object	Data Attributes
Correspondence	Correspondence OPERATION	Party	IFX handles correspondence as a property/extension of the party to which it is a subject
Object Relationships	N/A The IFX relationship objects have no direct equivalent in BIAN	PartyAcctRel, PartyCardRel, CardAcctRel	IFX uses XxxYyyRel objects to capture the attributes of relationships between objects X and Y. These objects can be used to fulfill some of the capabilities identified in the Card Facility service domain.

2.5.2 Activity Cross Reference

At this point in the analysis it is possible to associate the activities of the use-case with BIAN service domains and the IFX messages used to accomplish the necessary actions.

BIAN Service Domain	Action	IFX Message(s)	IFX Object(s)
Party Data Management	Identify Prospect	PartyAdd PartyInq	Party
Offer Management	Offer Product	OfferAdd	Offer
Customer Agreement	Execute Customer Agreement	PartyMod	Party
Document Services	Record Customer Agreement	PartyMod	Party
Sales Product Agreement	Execute Product Agreement	PartyCardRelAdd	PartyCardRel
Document Services	Record Product Agreement	PartyMod	Party
Customer Credit Rating	Qualify Customer	AcctAdd AcctMod	Acct
Position Keeping	Setup card transaction tracking	AcctAdd PartyAcctRelAdd CardAcctRelAdd	Acct PartyAcctRel CardAcctRel
Card Facility	Card Order	CardOrdAdd	CardOrd
Token Inventory	Create Card	CardAdd	Card
Correspondence	Send Welcome Pack	PartyMod	Party

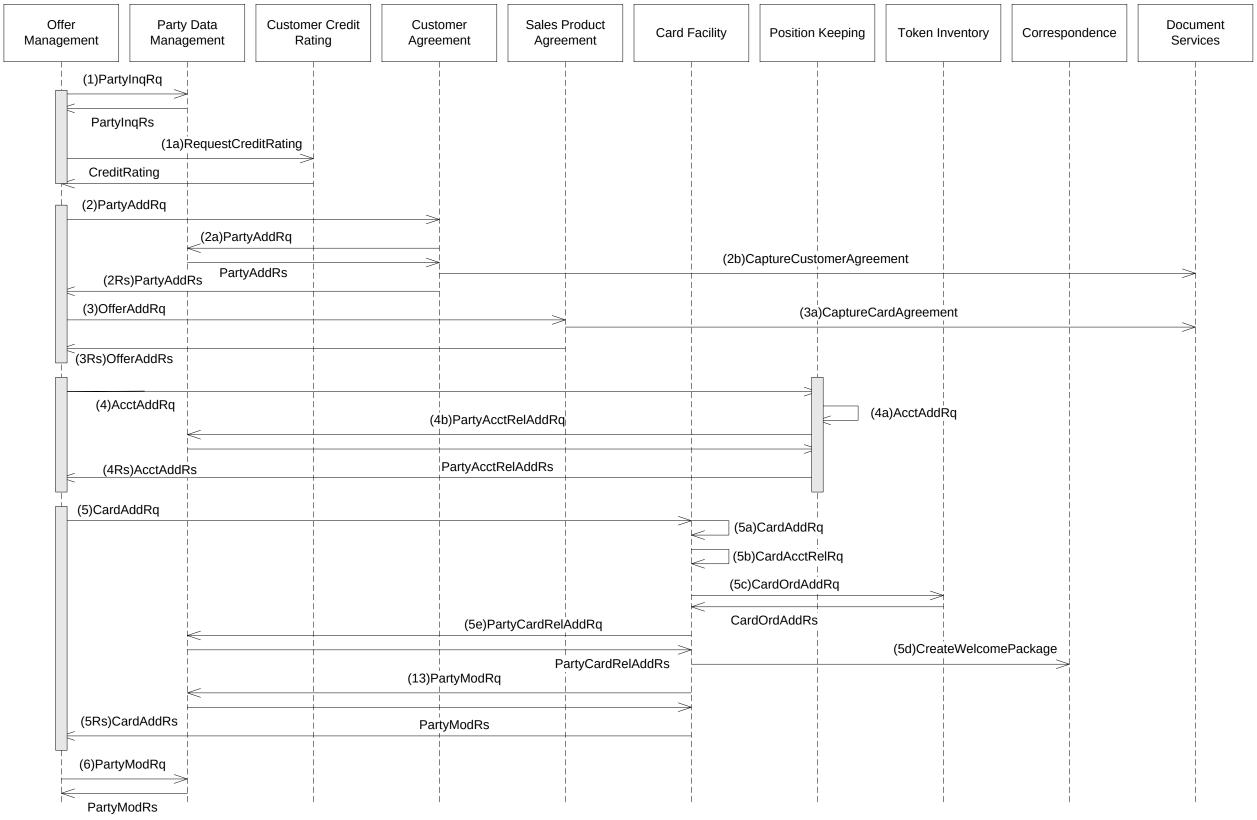
2.6 Message Sequence Diagrams

In the examples that follow, the service domains retain the BIAN service names as shown in the previous section. The two alternatives illustrate different collaboration between services and different demarcations of responsibility that align with the two alternative use case diagrams.

The table on the pages following the sequence diagrams provides additional details about interactions.

2.6.1.1 Alternative 1

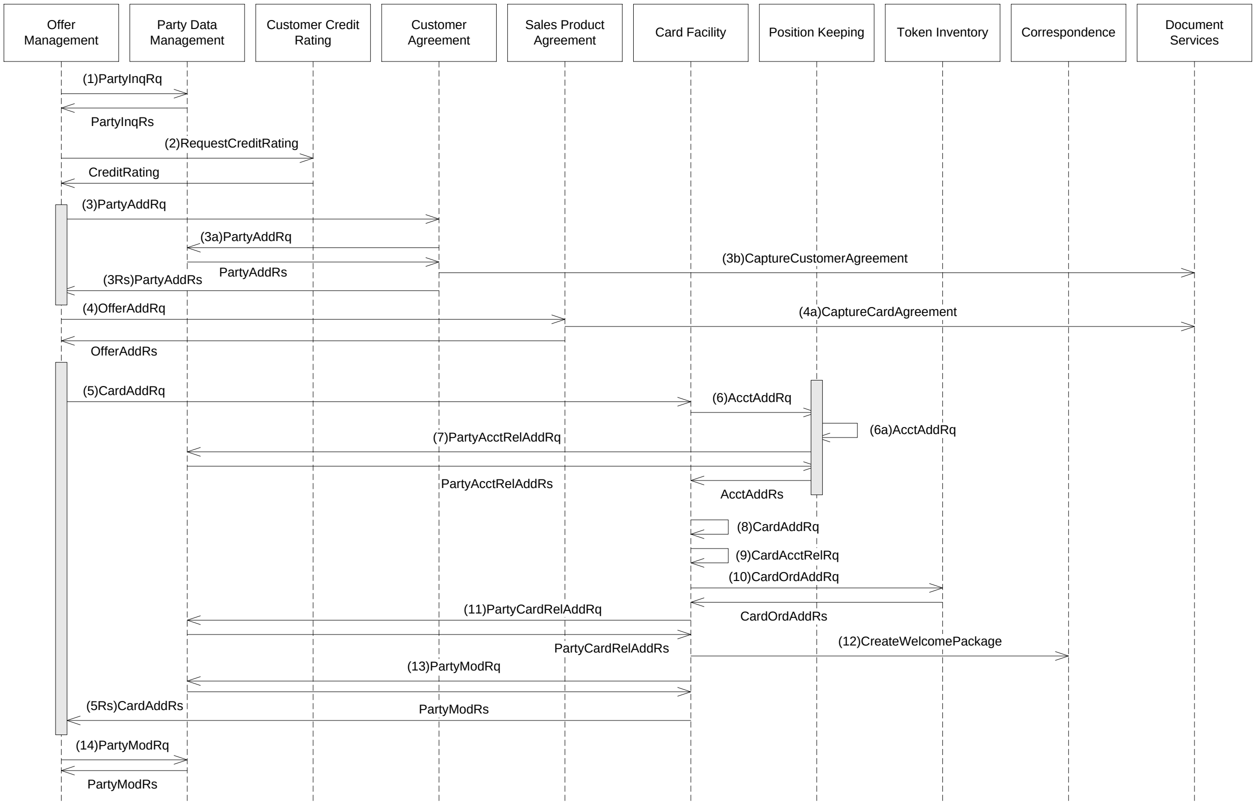
This message sequence diagram aligns with the first use case diagram in Section 2.2.1. The interactions in this diagram break encapsulation boundaries for *Offer Management* from the point of view of the BIAN control records. That is, *Offer Management* (rather than *Card Facility*) is initiating the account setup. This illustrates that implementing specific BIAN Services in a particular banking environment can be re-organized to match constraints within the bank.



Message names ending in *Rq* and *Rs* are IFX request and response messages. The other messages are outside the scope of the IFX Business Message Specification.

2.6.1.2 Alternative 2

This UML Sequence diagram illustrates the collaboration resulting from the Use Case model Alternative 2. Note that *Card Facility* manages the account setup through interactions with *Position Keeping*. This is a tighter alignment with the BIAN service definitions.



Message names ending in *Rq* and *Rs* are IFX request and response messages. The other messages are outside of the scope of the IFX Business Message Specification.

2.6.2 Message Sequence Details

The following table describes the messages for ***both*** alternative sequence diagrams. This is possible for two important reasons that highlight the modularity of both standards.

1. The IFX messages are intentionally designed to operate effectively in multi-tiered service environments. The messages can be invoked from any service and can be reliably forwarded to any other service regardless of physical or logical partitions.
2. The BIAN service domains are defined at a level of granularity that allow for precise operation. The services can be flexibly arranged for effective collaboration between processes.

> ***A key finding of this Proof of Concept is that both standards have the built-in modularity to adapt to the constraints of existing platforms without sacrificing integrity. This simplifies adoption regardless of whether the implementer is starting from a defined messaging standard or the architectural constraints of defined service definitions.***

Alt 1 Step	Alt 2 Step	Process Description	Message Name	Message Contents	Notes
0	0	Start	NA	NA	Prospect for new card account engages sales rep
1	1	Verify that this is not an existing customer	PartyInqRq	Certain identifying information such as SSN, EIN, Name, Address	The expected response is “No match found.” IFX message responses handle a wide variety of status returns including: “[1120] No Records Match Selection Criteria”, which is the expected condition in this case. An alternate implementation might expect to get a Party Record in return that will be updated to convert a prospect to a customer, or add products to an existing customer.
1a	2	Request Credit Rating	Request Credit Rating	Certain identifying information such as SSN, EIN, Name, Address	The process with the external credit rating agency is "out-of-band" with respect to the bank’s systems. The credit rating will be captured and included in PartyInfo.

Alt 1 Step	Alt 2 Step	Process Description	Message Name	Message Contents	Notes
2	3	New customer contract initiated	PartyAddRq	Identifying information Credit Rating from Step 2	This begins the process of converting the prospect to a card carrying customer.
2a	3a	Create the Customer record	PartyAddRq	Forwarding message from Offer Management. The IFX standard readily facilitates delegation of this sort. The PartyAddRs received in the Customer Agreement service will be routed back to Offer Management. (Step 3Rs).	In this illustration, responsibility for Party management is defined in the *Party Data Management* service. Note that the IFX standard indicates that the system of record for any object assigns the object ID that returned in the response to the Add message for the object – in this case PartyID.
2b	3b	Customer Agreement details are captured	CaptureCustomer Agreement	May be in the form of scanned documents or properly transcribed details	
3	4	Card Product Agreement initiated	OfferAddRq		
3a	4a	Card Agreement Details are captured	CaptureCard Agreement	May be in the form of scanned documents or properly transcribed details	
4	5a*	Add account	AcctAddRq	PartyId	The request to add an account that will be associated with the card is initiated. *In Alt2 this is initiated by the *Card Facility* rather than as an independent operation.
5	5	Initiate Card facility	CardAddRq	PartyId (from Step 2a or 3a)	This begins a multi-step process in the background that ends with card(s) ordered, accounts set up and welcome packages mailed.

Alt 1 Step	Alt 2 Step	Process Description	Message Name	Message Contents	Notes
4a	5b	Add account			In this illustration the *Position Keeping* service "owns" accounts (shown as a message to self) and thus responds to the AcctAddRq from the *Card Facility* with the AcctID it assigned in Step 4a/5b.
4b	5c	Record relationship of party to the account	PartyAcctRelAdd Rq	PartyID AcctId (from Step 4a/5b)	Record the Owner of the account. In this scenario it is also the responsibility of the *Position Keeping* service to record party-to-account relationships. In this illustration it is shown as collaboration with the Party Data Management Service. Optionally, this message could be repeated for other authorized users, co-signers, etc.
5a	5d	Create the card record	CardAddRq		Create the Card Record in system of record and assign a CardID.
5b	5e	Record the relationship of the card to the account	CardAcctRelAdd Rq	CardID (from Step 5a/5d) AcctId (from Step 4a/5b)	In this scenario it is shown as a responsibility and capability of the *Card Facility* service.
5c	5f	Order the physical card	CardOrdAddRq	CardId	This capability is owned by the *Token Inventory* service.
5d	5g	Record the relationship of the party to the card	PartyCardRelAdd Rq	PartyId CardId	Optionally, this may be repeated for additional cards issued to the same party
5e	5h	Generate a Welcome pack	CreateWelcome Package		
5f	5i	Record the fact that a welcome package has been sent	PartyModRq		

Alt 1 Step	Alt 2 Step	Process Description	Message Name	Message Contents	Notes
5Rs	5Rs	Signal that all back end processes are complete	CardAddRs	Return new card id and other pertinent data	After all steps that were initiated after the CardAddRq have been completed (or initiated in the case of correspondence and document capture), the process is complete.
6	6	Record the fact that the prospect is now a happy customer	PartyModRq	PartyId Customer Satisfaction rating	This attribute would require customization of the IFX PartyInfo.

2.6.3 Process Error Handling

Message and process error handling is not illustrated here. However, it is worth noting how process and system errors might be intercepted and handled.

- Every IFX message includes a status code, status severity and status description in its response. Therefore, at any point in the process when the status indicates an error, it is possible for the logic of the managing application to react accordingly.
- Because each message is intended to act on a single object, when errors occur the impact is isolated.
- Any client application that initiates an activity via a message request is anticipating a single response. Regardless of how many subsequent collaborating services are involved to fulfill a request, it is a straightforward operation for each service to evaluate the responses it gets from its delegate services and translate those results into an appropriate response for its direct clients.

  For example, in Alternative 2 Step 5 there is a single CardAddRq from Offer Management to Card Facility. But there are many business operations and collaborating services in Steps 5a-5i where an error may occur. If an error occurs at step 5b (adding the account), the Card Facility service can abort further operation and return an appropriate status code to its direct client, Offer Management.
- With services defined as a single purpose capability, as is the case with BIAN service domains, unintended side-effects are easily avoided.

The combined power of the IFX object-oriented, request-response message protocol coupled with well-defined, granular services as defined by BIAN makes it easy to isolate errors and to manage the reversal of previously successful steps if necessary.

2.7 The IFX Service Definition

To complete this example, we create a Service called *NewCardSetup.* Note that the IFX definition of *NewCardSetup* essentially aggregates a select subset of the functionality of several BIAN Service Domains. This exposes a useful business function to the bank sales representative.

The scope of any IFX Service definition is limited to itemizing the messages. The business rules and user interface(s) that tie the messages together are not defined as part of the IFX Service – the IFX standard is a Business *Message* Specification.

A *Service* is an encapsulated, discrete, unique business capability supported by the *Service Provider*. IFX Services are essentially defined as a list of messages that support a practical capability or functional partition. The service definition may also include a processing schedule, disclosures about the service, version indicators, status and more. A service must have a unique name within the context of the service provider.

The table below summarizes the service we have defined here. The table is not exhaustive but does describe the essential components.

Object	Attribute Hierarchy	Contents	Note
Service Provider <SvcProvider>	Service Provider Name <SvcProviderInfo>/<SvcProviderName>	http://org.ifx-bian	A service provider name is a unique url. Messages can be sent from any place on the globe to that url.
Service <Svc>	Service Name <SvcInfo>/<SvcName>	NewCardSetup	The name of the service is unique within the context of org.ifx-bian.

Object	Attribute Hierarchy	Contents	Note
Service <Svc>	Messages Supported <SvcInfo>/<MsgSupt>	PartyInqRq PartyAddRq PartyModRq OfferAddRq AcctAddRq CardAddRq PartyAcctRelAddRq CardAddRq CardAcctRelAddRq CardOrdAddRq PartyCardRelAddRq	MsgSupt is a repeating element that itemizes all messages available in the service. The order is arbitrary.
Service <Svc>	Operations Supported <SvcInfo>/<OperSupt>	PartyCardAcctRelOperRq	IFX provides for chaining a sequence of messages together so that they may be invoked as one message unit.
IFX Operation	PartyCardAcctRelOper An alternative approach to handling the relationships between party, card and account. Rather than expose the individual messages, the implementer may choose to create an operation to handle the associations.	PartyAcctRelAddRq CardAcctRelAddRq PartyCardRelAddRq	The individual messages that are used to compose the IFX Operation are shown.

2.8 Design Summary

There are probably dozens of ways to implement the *New Card Setup* business capability. Within any given bank there will be legacy system constraints, corporate policies, partnerships, enterprise architecture governance, security considerations and many other influences on the design of the *New Card Setup* service.

In this paper we illustrate two different views of the business scenario. Doing so demonstrates that a typical bank business capability can be assembled from:

- specific, granular service definitions
- object-oriented representation of data and processes
- message-oriented communications between collaborating services

- services provided by external partners, such as credit rating agencies
- support services such as document management

It also demonstrates that application of the standards does not unnecessarily constrain the technical implementation

While it was not within the scope of this POC to physically implement the capability, the high level design presented here is sufficiently detailed to demonstrate that it is viable using a deployment and packaging strategy typical of many organizations.

Most importantly, this proof of concept demonstrates that the IFX Business Message Specification can be used in conjunction with BIAN service definitions to fully realize critical business capabilities. The clear definition of services and service patterns provided by BIAN can be used to identify the boundaries of legacy systems and newly implementable capabilities.

3 Proof-of-Concept Discoveries

3.1 Significant Observations

During this exercise some key differences in how the two organizations approach the definition and documentation of services and objects became apparent. If these differences are recognized in advance, then implementers of the standards will encounter few difficulties identifying how the standards work together and how to map them to existing systems.

<u>Observations:</u>

- BIAN Service Domains define carefully scoped business capabilities.
- IFX Objects are generally data-centric representations of common business entities. It is usually quite apparent how to map IFX objects to existing data stores.
- For each service domain, the BIAN control record describes a function to be performed on a type of object.

<u>Implications:</u>

- ➢ IFX Objects will often be the subject of several different BIAN service domains; specific objects are involved in many business capabilities.
- ➢ BIAN Service Domains effectively imply which data attributes will be involved in an IFX Message.

  Example: The IFX PartyModRq message will be used in several Service Domains.

  In the *Party Data Management* service domain, PartyModRq will update a subset of the attributes of the Party object regarding reference details.

  In the *Customer Agreement* domain, the PartyModRq message will be used to update contract details.

<u>Observations:</u>

- BIAN describes functional patterns that characterize business activities –
  e.g. Plan, Manage, Enroll, Track.
- IFX describes functional patterns in terms of data management –
  e.g. Add, Modify, Delete, Synchronize.
- IFX Messages usually result in state changes or updates to the objects targeted by the messages.

<u>Implications:</u>

- ➢ Generally there is not a direct correlation between an IFX function (message) and a BIAN function (business operation).
- ➢ IFX messages enable implementation of, and tracking the results of, business functions described in the service domain.
- ➢ An implementer will use one or more IFX messages to accomplish the work within scope of the BIAN service.

Observations:

- Generally a single IFX object is likely to be the subject of a particular BIAN control record. However, IFX represents certain relationships between objects as a first class objects in their own right. These have no direct parallel in the BIAN service landscape or control records.
- In some cases IFX Objects have strong parallels to BIAN control records. When the control record defined by BIAN is "elevated" to the level of an object, there is more likely to be a one-to-one mapping between an IFX Object and a BIAN service domain.

Implication:

- ➢ Mapping objects and messages between the two standards will sometimes require bank-specific decisions about whether to favor the feature-functionality of one approach over the other.

3.2 Gaps

Some gaps were found during the mapping analysis. Resolutions to these gaps have been discussed and agreed between IFX and BIAN. The results of the discussion are recorded below:

Gap	Resolution
There are some BIAN service interactions that do not require responses from the collaborating service. The IFX message specification always calls for responses.	This is not an impediment to implementation. Option 1: The message responses can be ignored by the initiating client. Option 2: Implementers may choose to disregard the standard requirement to provide response messages. Within an infrastructure that is completely within the control of the implementer, this may not be viewed as a material deviation from the standard.
There are some capabilities called for in the BIAN Service Domain that have no counterpart in the IFX standard (Document Services, for example).	This condition is to be expected in the normal course of implementation. Option 1: Extend the IFX messages to build custom objects, messages or services. IFX Forum provides guidance on how this is to be accomplished. Option 2: Substitute in-house solutions.
IFX provides for some capabilities that are not defined in the BIAN Service Domain (Card-Account relationships, for example).	This condition is to be expected in the normal course of implementation. Option 1: Enhance the service domain to provide the additional capability. Option 2: Drop the requirement for the undefined capability Option 3: Create a separate Service Domain to manage the additional capability.

4 Conclusions

The alignment of the BIAN and IFX approaches to SOA standardization for this use case was very strong. We were able to quite readily illustrate two different ways to implement *New Card Setup* using the BIAN service domain definitions and IFX messaging. The design of practical implementations was straightforward and there were few gaps to address.

These key design principles were demonstrated in this POC.

1. The IFX messages are intentionally designed to operate effectively in a multi-tiered, service oriented architecture. Messages can be invoked from any service and can be reliably forwarded to other services regardless of physical or logical partitions.
2. The BIAN service domains are defined at a level of granularity that allows for flexible assembly and sequencing of processes.

The modularity and complementary nature of the standards proved flexible enough to support these conclusions.

- The BIAN-defined service domains can be mapped to, and implemented using, a pre-existing service oriented messaging standard – in this case, the IFX Business Message Specification.
- The IFX message and service framework can be used to implement a pre-existing view of standard business services – in this case as defined in the BIAN service landscape and service domains.

It is reasonable to conclude that these results can be generalized beyond the scope of this proof of concept.

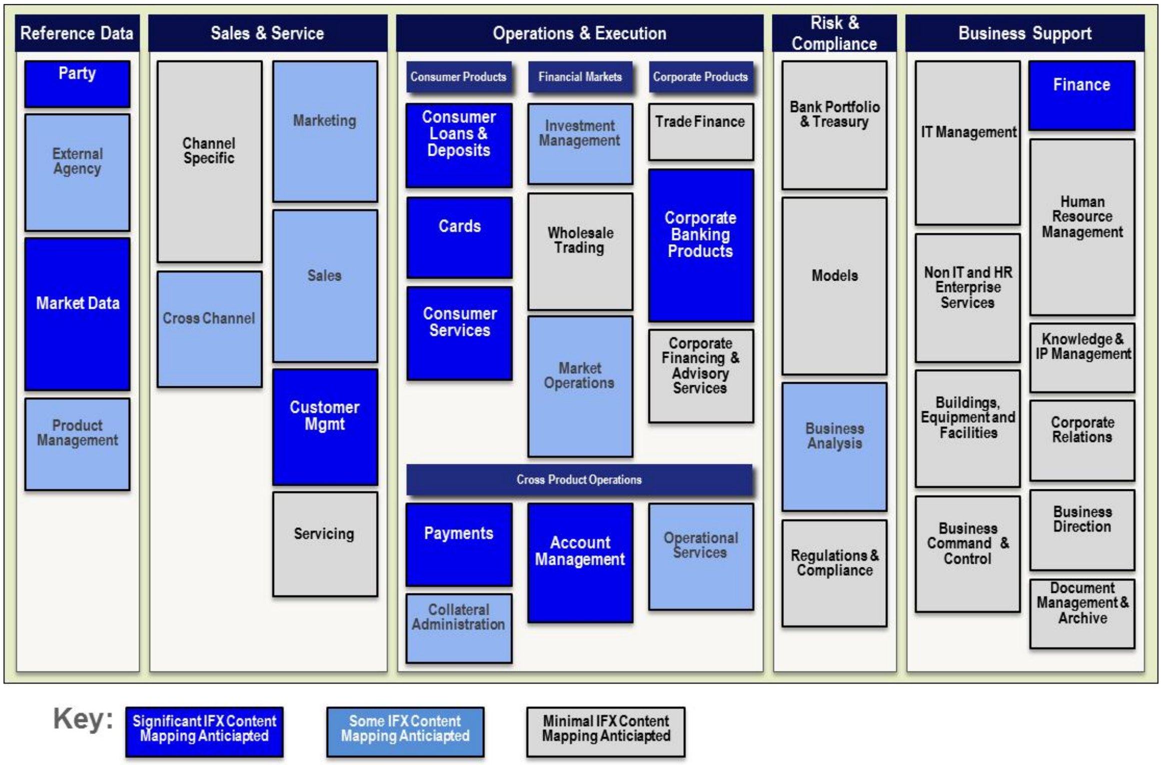
- Both standards are technology neutral.
- The business objects involved were immediately evident in the IFX standard.
- The business capabilities were easily recognized in the BIAN standard.
- Neither standard was forced to "bend the rules" in any significant way in order to accommodate the other.

These two standards can each be adapted to existing banking systems, setting the stage to introduce and leverage the benefits of Service Oriented Architecture. Using the two together can significantly reduce the analysis effort of building properly defined, supportable services in any financial institution.

4.1 Mapping BIAN and IFX Capabilities

Based on the promising nature of the results, IFX and BIAN reviewed the two standards at a high level to assess where it is likely that the capabilities of the two standards can be readily leveraged within banks today. We believe that it will be worth the effort to create a more detailed mapping of IFX object and messages to BIAN Service Domains in the future.

The diagram below illustrates the BIAN Service Landscape V2.5 with IFX v2.3 coverage indicated



Note this is an initial mapping for information purposes only – a more formal analysis is required to complete the message to Service Domain mapping described in this paper.

Appendices

Appendix A Service Model Comparisons

A.1 BIAN Service Domains

The BIAN Service Domains define capabilities with implied boundaries. Each domain's service operations define available Interfaces – or services – available to the "outside world". The BIAN Standard does not define which service domains may call upon the services of other service domains nor does it prescribe how such collaborations are to be implemented. .

A BIAN Service Domain represents a discrete business capability partition. The business purpose or role of a BIAN Service Domain combines an instance of some object or entity and a type of function performed on that instance. For example, the object could be something tangible like a piece of equipment, and possible functions could be to maintain or operate the equipment. The object can also be something intangible such as a customer relationship, and suitable functions might be to maintain a governing agreement or to define and implement a development plan for a customer relationship.

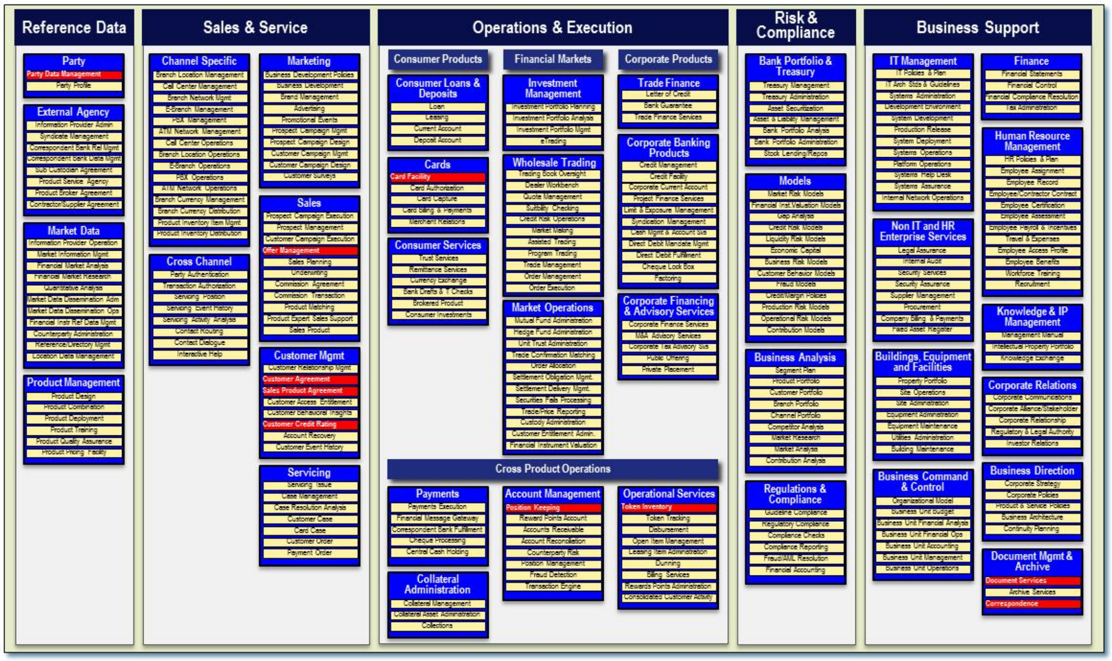
A BIAN Service Domain also has an associated "Control Record" that manages the different state and state transitions that an instance of the object or entity passes through as the service domain's function is executed through the complete life cycle.

Each service domain performs a single function on a specific object. In order to define canonical standards, the business capability partitions represented by BIAN Service Domains need to be elemental in nature. BIAN has developed techniques to correctly scope a service domain that combines a deterministic decomposition of the objects or entities that make up a bank and applying a standard list of allowed functions performed in the execution of business.

Each BIAN Service Domain has a defined collection of service operations that provide all necessary access. Any business activity or event can be modeled as a combination of service exchanges between different service domains. BIAN Service Domains define the capability partitions and supported service operations but do not constrain the pattern of interactions that may occur between service domains in response to a particular business event. As a result, the same service domains and service operations may be orchestrated to support different execution patterns as may be represented in different business processes that might be implemented. This flexibility to interpret that same BIAN conceptual design in different implementations is demonstrated by applying mapping two different processes or implementation patterns to the same selection of BIAN Service Domains.

A.1.1 Service Landscape

The BIAN Service Domains are organized in a hierarchical representation – the BIAN Service Landscape. As illustrated, the Service Domains highlighted in red were identified as participants in the POC use case. A more legible version of this diagram is available at www.bian.org. The point of this illustration is simply to show that a particular business capability is likely to involve several service domains from different portions of the overall landscape.



Reference Data

Party
Party Data Management
Party Profile

External Agency
Information Provider Admin
Syndicate Management
Correspondent Bank Rel Mgmt
Correspondent Bank Data Mgmt
Sub Custodian Agreement
Product Service Agency
Product Broker Agreement
Contractor/Supplier Agreement

Market Data
Information Provider Operation
Market Information Mgmt
Financial Market Analysis
Financial Market Research
Quantitative Analysis
Market Data Dissemination Adm
Market Data Dissemination Ops
Financial Instr Ref Data Mgmt
Counterparty Administration
Reference/Directory Mgmt
Location Data Management

Product Management
Product Design
Product Combination
Product Deployment
Product Training
Product Quality Assurance
Product Pricing Facility

Sales & Service

Channel Specific
Branch Location Management
Call Center Management
Branch Network Mgmt
E-Branch Management
PBX Management
ATM Network Management
Call Center Operations
Branch Location Operations
E-Branch Operations
PBX Operations
ATM Network Operations
Branch Currency Management
Branch Currency Distribution
Product Inventory Item Mgmt
Product Inventory Distribution

Cross Channel
Party Authentication
Transaction Authorization
Servicing Position
Servicing Event History
Servicing Activity Analysis
Contact Routing
Contact Dialogue
Interactive Help

Marketing
Business Development Policies
Business Development
Brand Management
Advertising
Promotional Events
Prospect Campaign Mgmt
Prospect Campaign Design
Customer Campaign Mgmt
Customer Campaign Design
Customer Surveys

Sales
Prospect Campaign Execution
Prospect Management
Customer Campaign Execution
Offer Management
Sales Planning
Underwriting
Commission Agreement
Commission Transaction
Product Matching
Product Expert Sales Support
Sales Product

Customer Mgmt
Customer Relationship Mgmt
Customer Agreement
Sales Product Agreement
Customer Access Entitlement
Customer Behavioral Insights
Customer Credit Rating
Account Recovery
Customer Event History

Servicing
Servicing Issue
Case Management
Case Resolution Analysis
Customer Case
Card Case
Customer Order
Payment Order

Operations & Execution

Consumer Products

Consumer Loans & Deposits
Loan
Leasing
Current Account
Deposit Account

Cards
Card Facility
Card Authorization
Card Capture
Card billing & Payments
Merchant Relations

Consumer Services
Trust Services
Remittance Services
Currency Exchange
Bank Drafts & T Checks
Brokered Product
Consumer Investments

Financial Markets

Investment Management
Investment Portfolio Planning
Investment Portfolio Analysis
Investment Portfolio Mgmt
eTrading

Wholesale Trading
Trading Book Oversight
Dealer Workbench
Quote Management
Suitability Checking
Credit Risk Operations
Market Making
Assisted Trading
Program Trading
Trade Management
Order Management
Order Execution

Market Operations
Mutual Fund Administration
Hedge Fund Administration
Unit Trust Administration
Trade Confirmation Matching
Order Allocation
Settlement Obligation Mgmt.
Settlement Delivery Mgmt.
Securities Fails Processing
Trade/Price Reporting
Custody Administration
Customer Entitlement Admin.
Financial Instrument Valuation

Corporate Products

Trade Finance
Letter of Credit
Bank Guarantee
Trade Finance Services

Corporate Banking Products
Credit Management
Credit Facility
Corporate Current Account
Project Finance Services
Limit & Exposure Management
Syndication Management
Cash Mgmt & Account Svs
Direct Debit Mandate Mgmt
Direct Debit Fulfilment
Cheque Lock Box
Factoring

Corporate Financing & Advisory Services
Corporate Finance Services
M&A Advisory Services
Corporate Tax Advisory Svs
Public Offering
Private Placement

Cross Product Operations

Payments
Payments Execution
Financial Message Gateway
Correspondent Bank Fulfilment
Cheque Processing
Central Cash Holding

Collateral Administration
Collateral Management
Collateral Asset Administration
Collections

Account Management
Position Keeping
Reward Points Account
Accounts Receivable
Account Reconciliation
Counterparty Risk
Position Management
Fraud Detection
Transaction Engine

Operational Services
Token Inventory
Token Tracking
Disbursement
Open Item Management
Leasing Item Administration
Dunning
Billing Services
Rewards Points Administration
Consolidated Customer Activity

Risk & Compliance

Bank Portfolio & Treasury
Treasury Management
Treasury Administration
Asset Securitization
Asset & Liability Management
Bank Portfolio Analysis
Bank Portfolio Administration
Stock Lending/Repos

Models
Market Risk Models
Financial Inst.Valuation Models
Gap Analysis
Credit Risk Models
Liquidity Risk Models
Economic Capital
Business Risk Models
Customer Behavior Models
Fraud Models
Credit/Margin Policies
Production Risk Models
Operational Risk Models
Contribution Models

Business Analysis
Segment Plan
Product Portfolio
Customer Portfolio
Branch Portfolio
Channel Portfolio
Competitor Analysis
Market Research
Market Analysis
Contribution Analysis

Regulations & Compliance
Guideline Compliance
Regulatory Compliance
Compliance Checks
Compliance Reporting
Fraud/AML Resolution
Financial Accounting

Business Support

IT Management
IT Policies & Plan
IT Arch Stds & Guidelines
Systems Administration
Development Environment
System Development
Production Release
System Deployment
Systems Operations
Platform Operations
Systems Help Desk
Systems Assurance
Internal Network Operations

Non IT and HR Enterprise Services
Legal Assurance
Internal Audit
Security Services
Security Assurance
Supplier Management
Procurement
Company Billing & Payments
Fixed Asset Register

Buildings, Equipment and Facilities
Property Portfolio
Site Operations
Site Administration
Equipment Administration
Equipment Maintenance
Utilities Administration
Building Maintenance

Business Command & Control
Organizational Model
Business Unit Budget
Business Unit Financial Analysis
Business Unit Financial Ops
Business Unit Accounting
Business Unit Management
Business Unit Operations

Finance
Financial Statements
Financial Control
Financial Compliance Resolution
Tax Administration

Human Resource Management
HR Policies & Plan
Employee Assignment
Employee Record
Employee/Contractor Contract
Employee Certification
Employee Assessment
Employee Payroll & Incentives
Travel & Expenses
Employee Access Profile
Employee Benefits
Workforce Training
Recruitment

Knowledge & IP Management
Management Manual
Intellectual Property Portfolio
Knowledge Exchange

Corporate Relations
Corporate Communications
Corporate Alliance/Stakeholder
Corporate Relationship
Regulatory & Legal Authority
Investor Relations

Business Direction
Corporate Strategy
Corporate Policies
Product & Service Policies
Business Architecture
Continuity Planning

Document Mgmt & Archive
Document Services
Archive Services
Correspondence

A.2 IFX Service Model

The most fundamental assumption underlying the IFX standard is that one or more services are being offered by a service provider. A service provider may manage these services directly or may choose to partner with others to provide the service. Businesses (typically financial institutions) that adopt the IFX standard to provide services to customers (clients) are known as Service Providers. Service Providers are identified by a URL which is also the access point for clients to invoke the services offered. All client messages are directed to the Service Provider at the specified URL.

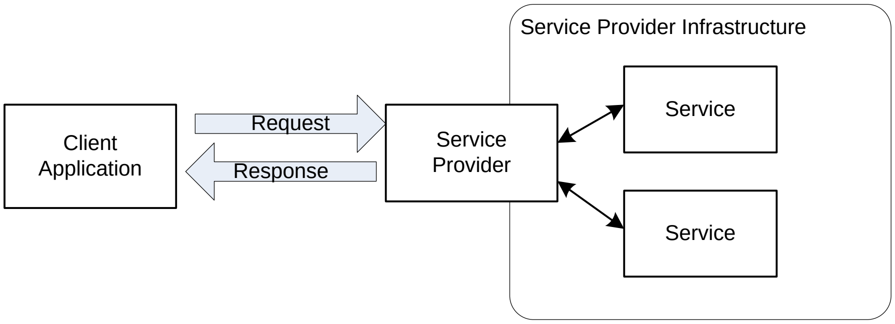
Specific services are not defined by IFX Forum. Rather, Service Providers define services at a level of granularity and functionality convenient to them. These may be very broad capabilities or very specific functional capabilities within a larger context.

The diagram below describes the relationship between clients, service providers and services in their simplest form.

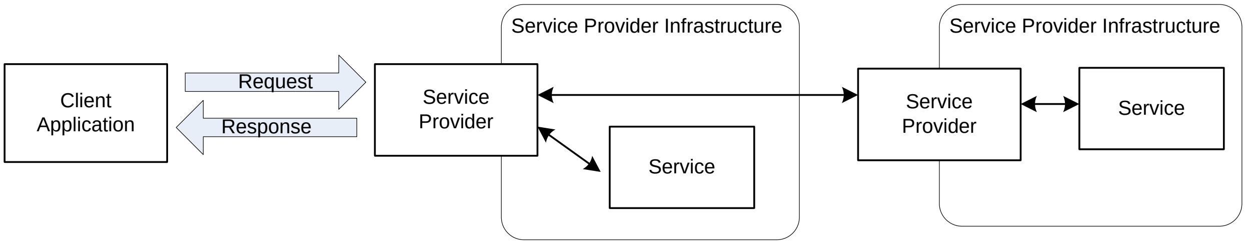
Request Messages are directed to the Service Provider by the Client.

The Service Provider uses one or more Services to satisfy the request.

The Client receives a response.



Service Providers may invoke services provided by a business partner (another Service Provider) to fulfill a particular capability. The client has a single point of access to invoke a service and this secondary relationship is not known to the client. The service provider known to the client handles all message requests and message responses on behalf of the client, routing requests to any other partner or system desired and routing responses back to the client. The diagram below illustrates this slightly more complex model.



A.3 BIAN and IFX Service Models Comparison

The Service Model Mapping was based on IFX Object version 2.3 and BIAN Service Definition version 2.5. The result of the mapping is listed below:

BIAN	IFX	Findings
	Service Provider	An IFX Service Provider (SP) offers to meet the needs of a business or customer with a focus on particular business capabilities. An SP may be an organization or an entry point into a bit of infrastructure designed to provide a set of capabilities. The boundaries of a service provider are typically defined by the same criteria described by BIAN Business Area and Business Domain.
Service Domain	Service	A practical capability or functional partition that can be service-enabled as a discrete and unique business capability. A Service Provider may offer one or more Services associated with Service Domain(s) as defined by BIAN.
Service Operation	Interface Operations Messages	In order to provide the capabilities identified for a Service, certain messages and/or operations are employed by the Service Provider. IFX Supports discrete messages, a sequenced set of messages as an operation and a design-time mechanism to organize messages and operations as an interface to a service.
Control Record		The BIAN standard breaks business capabilities down to the finest/elemental partitions in order to define canonical service operation. The control record describes functional activities for particular entities that are within the scope of the service domain. Control records describe the different state and state transitions that an instance of the object or entity passes through as the BIAN Service Domain's function is executed through the complete life cycle.

Appendix B IFX Messages and Objects

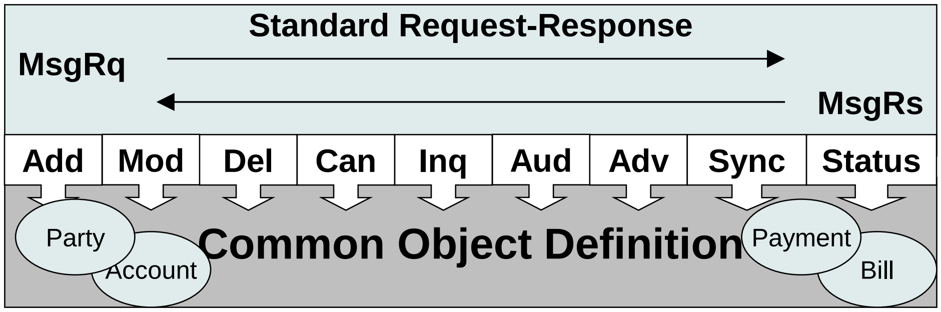
B.1 Discussion

An IFX Object is a set of data that is organized according to a consistent pattern. IFX Objects are constructed from basic building blocks of data Elements and data Aggregates, where elements are single pieces of information with defined data types and aggregates are groups of related elements identified using a single name for convenience.

IFX Objects support a well-defined set of operations (or methods) that cause IFX objects to be created, modified and destroyed.

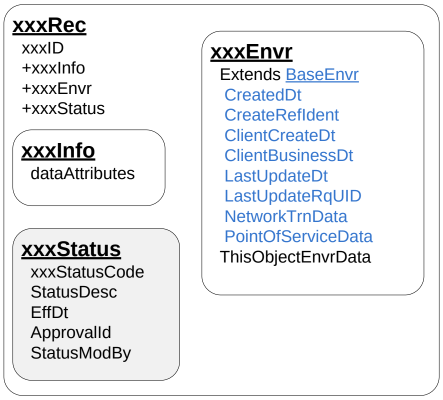
The IFX Business Message Specification is designed to operate in stateless, multi-tiered, service-oriented environments. The framework consists of Common Object Definitions with well-defined data semantics and a Request-Response message protocol, where each message is targeted to act upon one object and each response indicates success or failure, to ensure common understanding of object states after each attempted or successful message operation.

The framework is depicted below with some examples of defined IFX Objects.



B.2 Object Representation

As depicted in the figure at right, each instance of an IFX object is a record (xxxREC) with a unique ID, a set of client-managed data properties contained in an INFO aggregate, a set of environmental (ENVR) data properties managed by the server and a STATUS aggregate describing the object's current state.



Detailed information for these object representations can be found in the Object Framework section of the BMS at http://bms.ifxforum.org.

B.3 BIAN Interaction Patterns and IFX Message Protocols

BIAN interaction names do not describe machine-machine protocols; they describe service dependencies. Nevertheless, it is possible to illustrate a rough approximation of how the BIAN interaction patterns can be manifest in a message protocol such as IFX.

The IFX Message standard is based on a Request-Response protocol. In certain implementations the response messages are either ignored or not sent at all. There are also situations where the processing required to satisfy a request is not expected in real-time. IFX supports such requests with a response that acknowledges receipt of the valid message, a token that can be used to inquire about process results at a later time and, optionally, a timeframe when the results will be available.

The chart below describes how to implement some of the BIAN interaction patterns using the IFX Request-Response design.

BIAN Interaction Pattern	**IFX Protocol**	**Notes**
Request & Hold - Synchronous	Request-Response	Response is expected to report success or failure.
Handoff	Request-Response	The only Response expected is that the message was received; it is either ignored or not sent depending on implementation decisions.
Request & Monitor – asynchronous	Request-Response with AsyncRsData	Response is not immediately expected. The message response code will be [900] with a severity [Warn] and a token <AsyncRqUID> that can be used in later messages to retrieve results and processing status. See AsyncRsData in IFX BMS.
NA (Multi-step process request)	xxxOperRq- xxxOperRs	The IFX standard facilitates the desire to bundle a sequence of messages with *IFX Operations*. Such a request will include a list of messages along with instructions regarding processing – i.e.,<OperRules>. The operation rules indicate how to handle errors and warnings when processing the sequence of messages. (Abort, Continue, ReverseAll, ReverseProcessed.)
Make announcement	NA	Allows for subscribers to ingest a service report. It is possible to send IFX messages to any number of recipients and ignore responses, but there is no inherent Publish-Subscribe protocol in the IFX standard.

Document History

Authors		
Role	**Name**	**Company**
Author	Richard Urban	IFX Forum, Inc.
Author	Guy Rackham	BIAN
Author	Dave Banko	BIAN

Reference Documents		
Topic	**Type of Document**	**Reference to Document**
IFX Standards	IFX Standard Online	http://www.ifxforum.org/standards/standard/
IFX SOA Implementation Guide	Work in progress	Expected to be published January 2014
BIAN Service Landscape	Version 2.5 landscape	http://bian.org/assets/bian-standards/bian-service-landscape-2-5/

Version History		
Version	**Date of Change**	**Author**
1.0	November 26, 2013	Original Authors

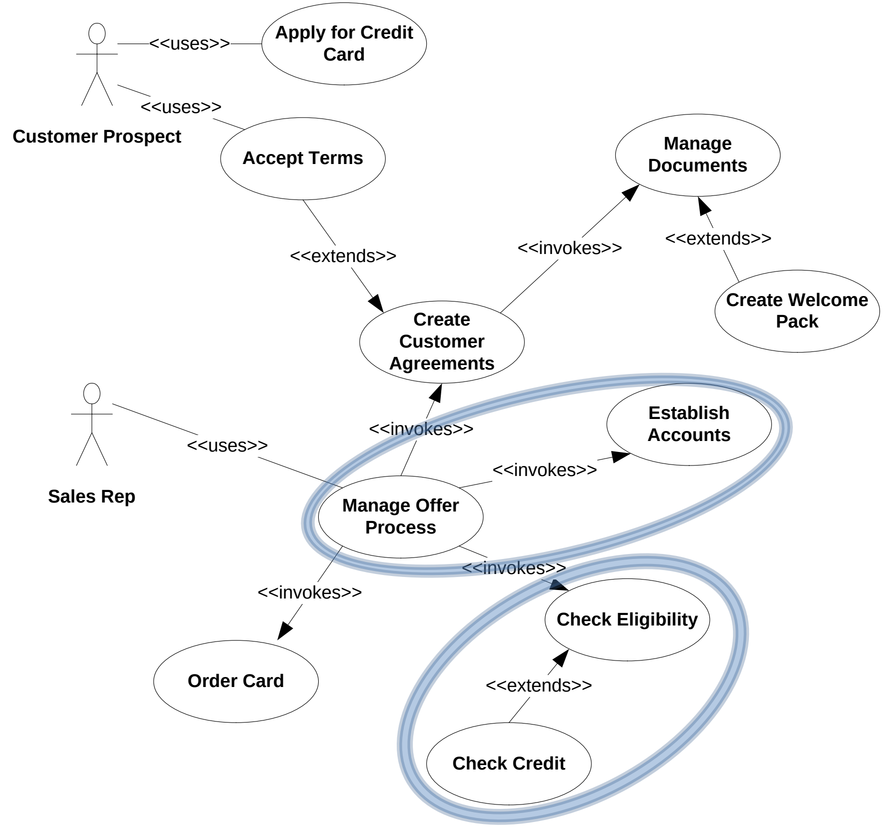
Diagrams and Illustrations

In the interest of providing a readable narrative and workflow, the body of this report is presented entirely in portrait page orientation. In the interest of presenting diagrams and illustrations in a more readable form for detailed reference, many diagrams are repeated in this section in full-page landscape rendering.

Figure 1 - Use Case Alternatives

Use Case Alternative 1

The *Manage Offer Process* directly invokes several other processes, including *Establish Accounts* and *Order Card*. *Check Credit* is viewed as an extension of the overall Check Eligibility process.



Use Case Alternative 2

In this model, *Check Credit* is a separate service not viewed as an extension of *Check Eligibility.* Also, *Card Facility* manages the account setup and card order. Lastly, this model shows that there are separate Customer and Card Product Agreements.

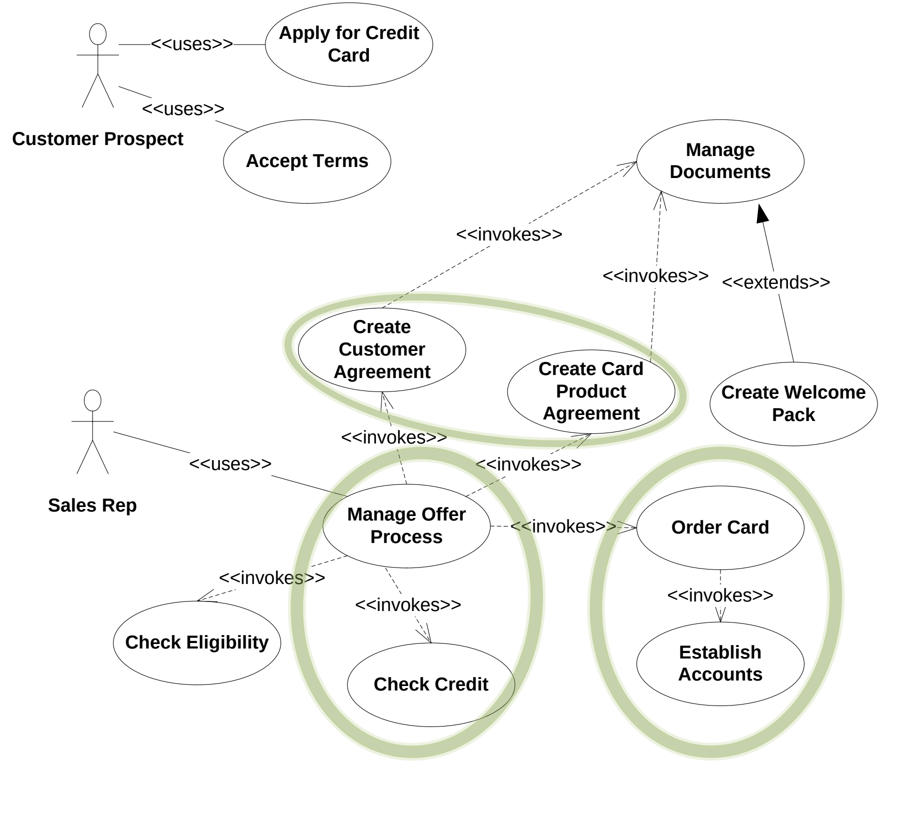


Figure 2 - BIAN Domains Collaboration Sequence

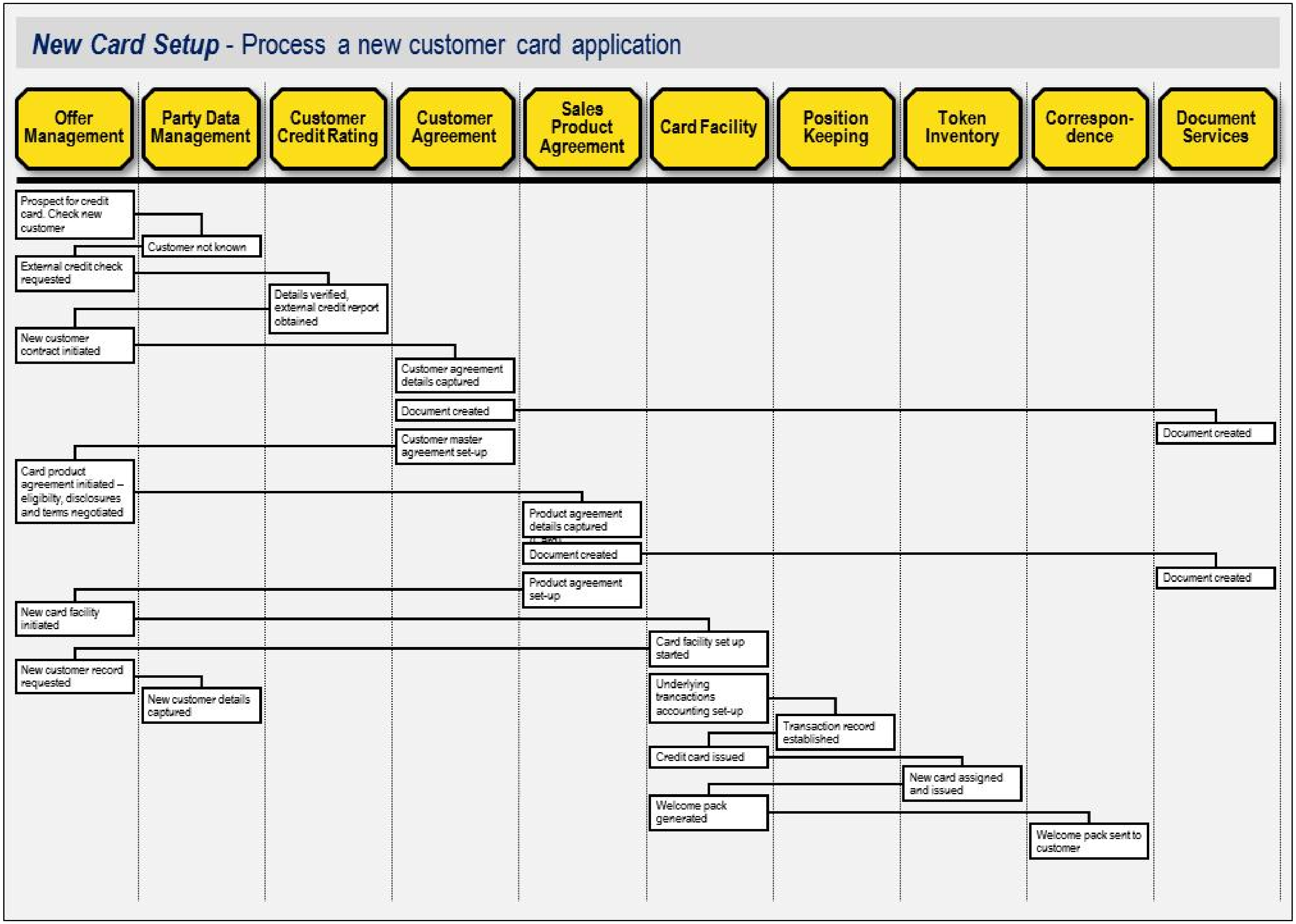

New Card Setup - Process a new customer card application
Offer Management
Party Data Management
Customer Credit Rating
Customer Agreement
Sales Product Agreement
Card Facility
Position Keeping
Token Inventory
Correspon-dence
Document Services
Prospect for credit card. Check new customer
Customer not known
External credit check requested
Details verified, external credit report obtained
New customer contract initiated
Customer agreement details captured
Document created
Document created
Customer master agreement set-up
Card product agreement initiated – eligibilty, disclosures and terms negotiated
Product agreement details captured
Document created
Document created
Product agreement set-up
New card facility initiated
Card facility set up started
New customer record requested
New customer details captured
Underlying trancactions accounting set-up
Transaction record established
Credit card issued
New card assigned and issued
Welcome pack generated
Welcome pack sent to customer

Figure 3 - Message Sequence Diagram, Alternative 1

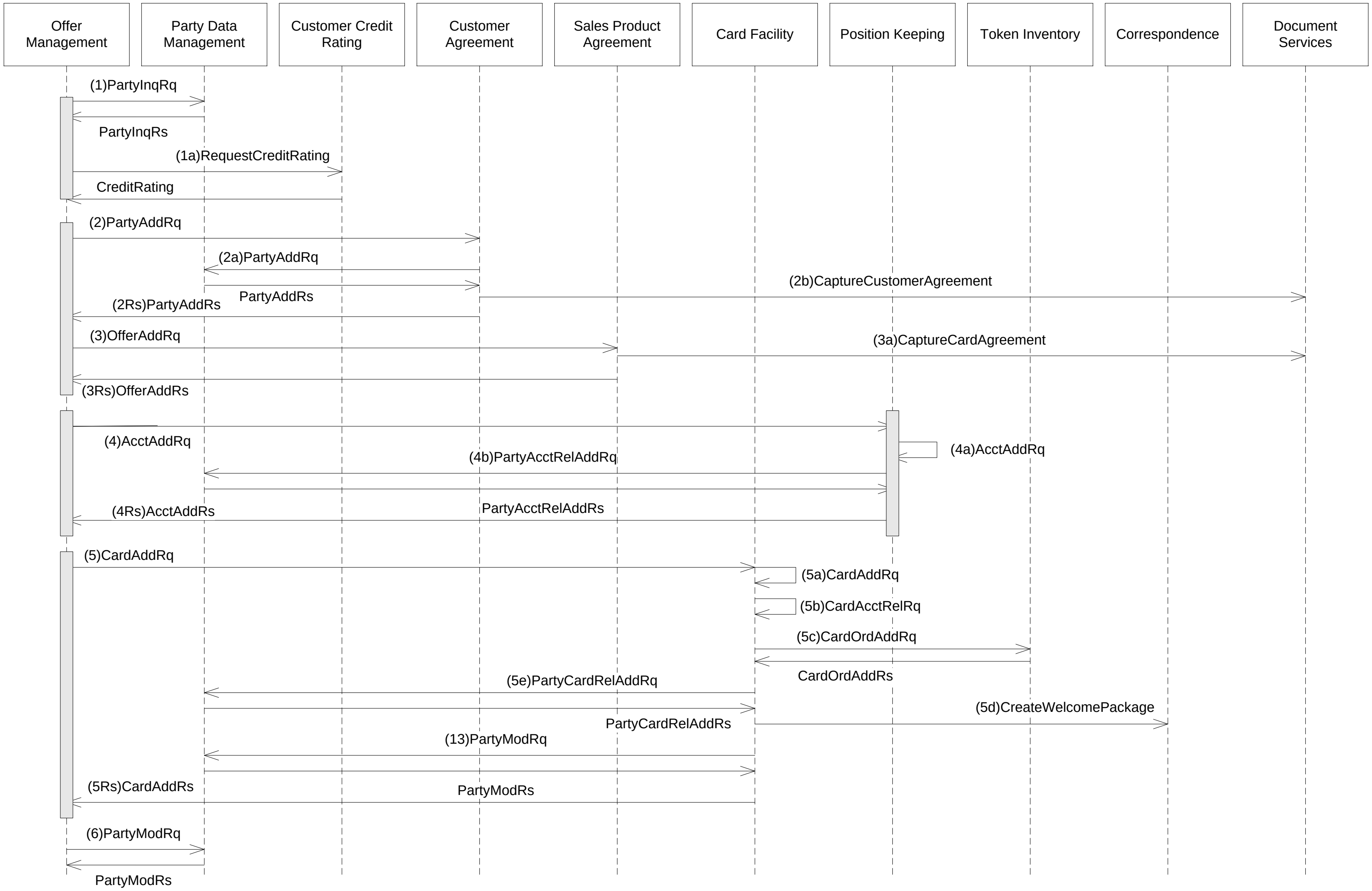


Figure 4 - Message Sequence Diagram, Alternative 2

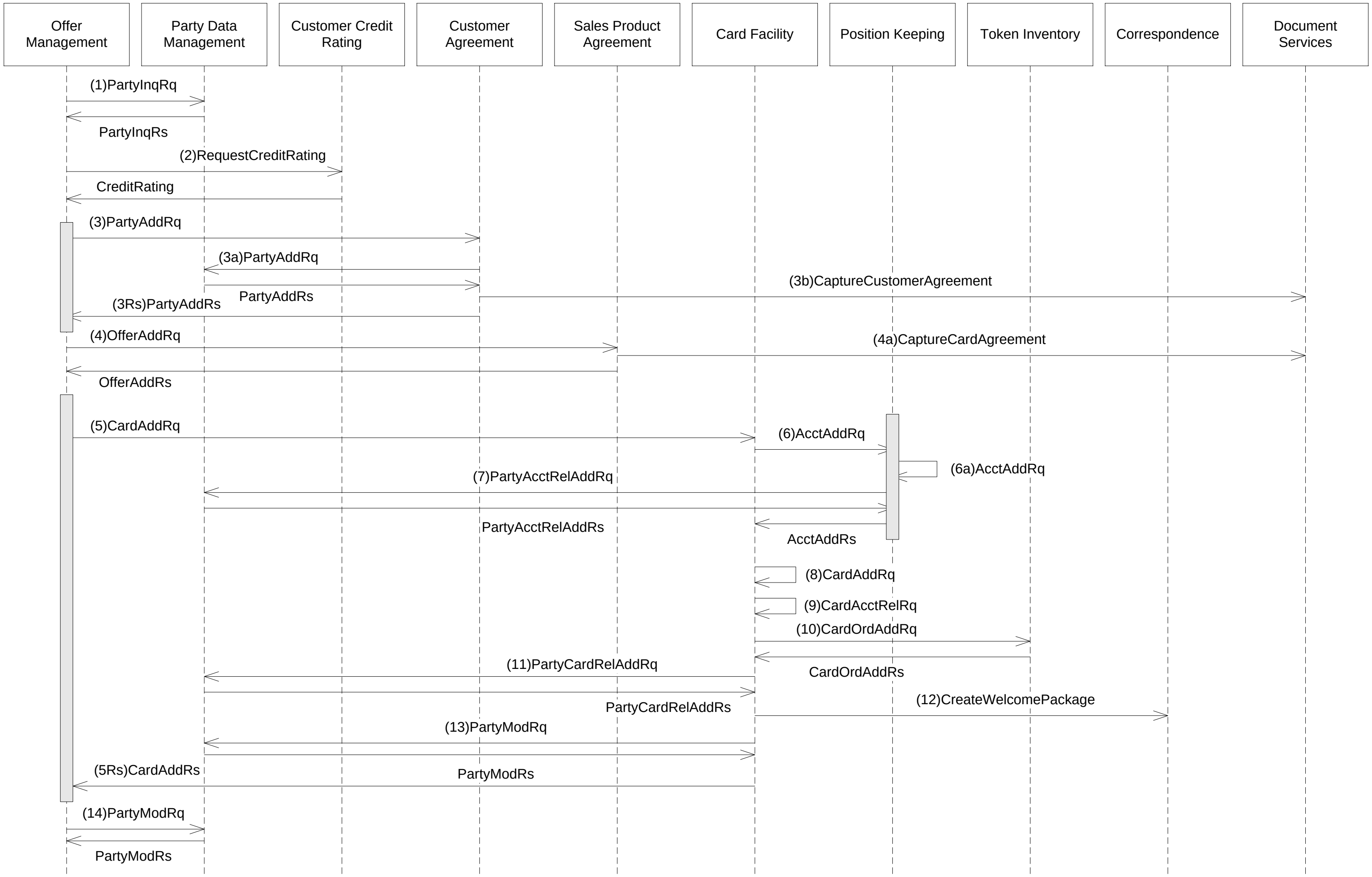


Figure 5 - BIAN Service Landscape

The Service Domains highlighted in red were identified as participants in the POC use case.

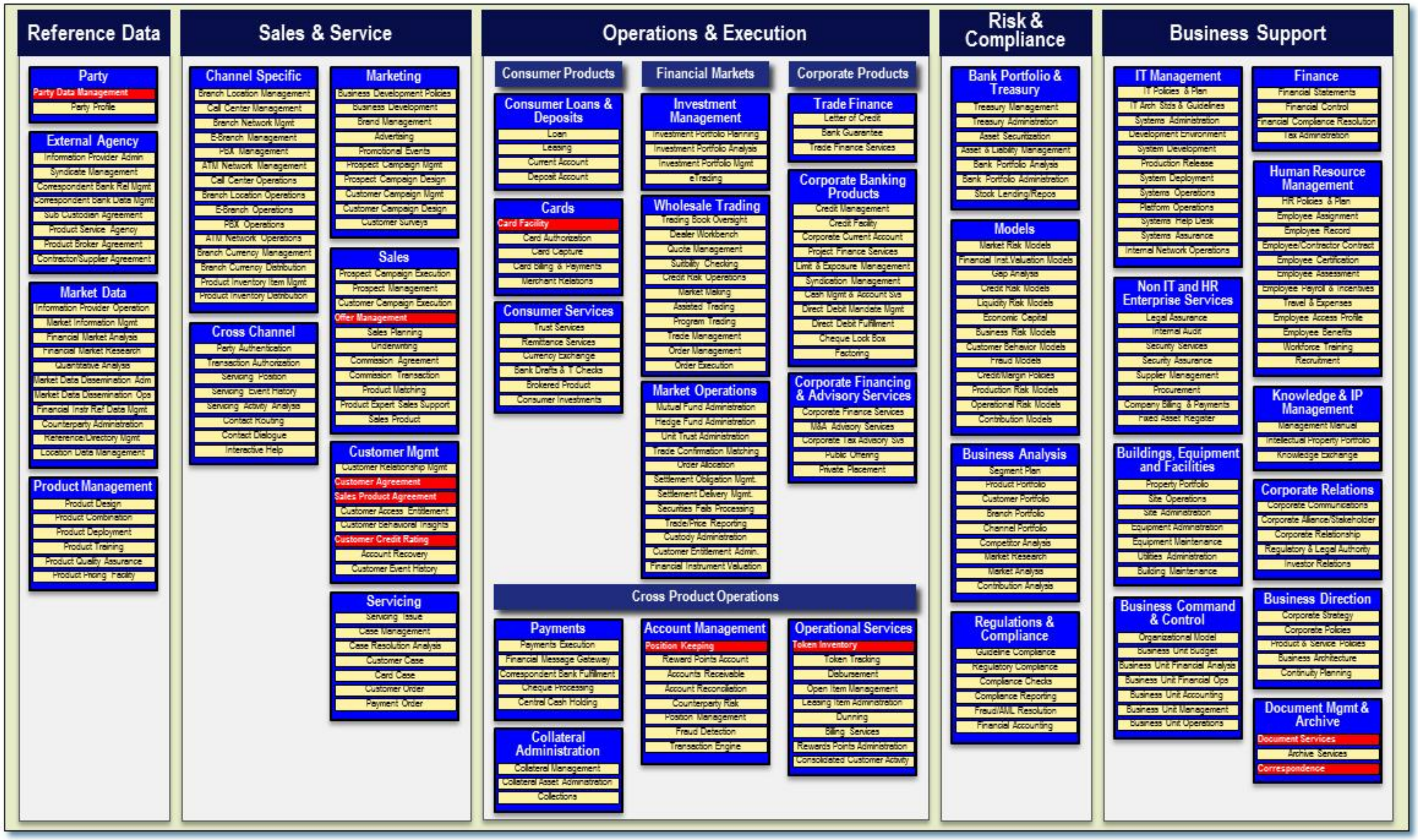
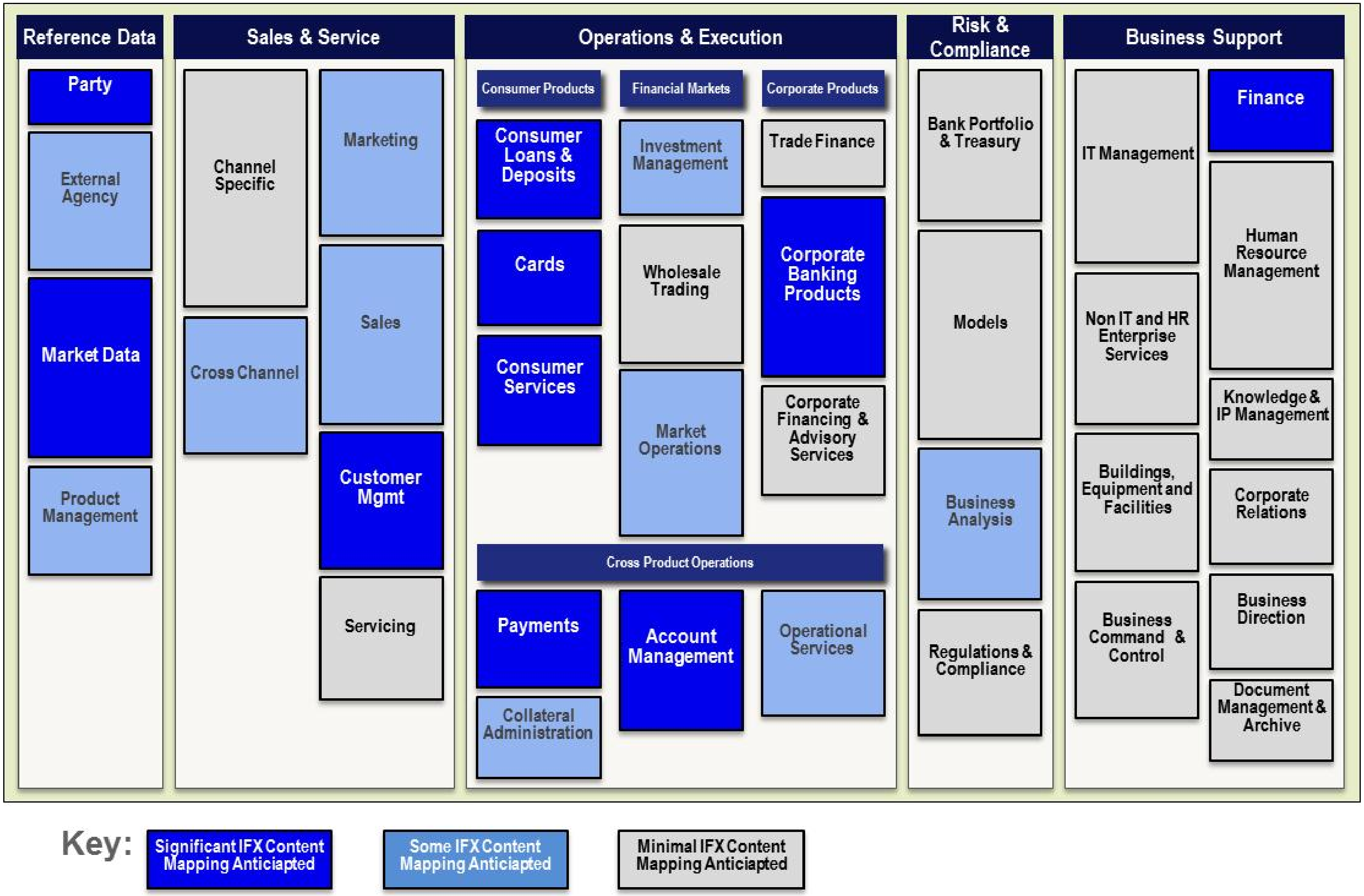


Figure 6 - The BIAN Service Landscape V2.5 with IFX v2.3 Coverage


Reference Data
Party
External Agency
Market Data
Product Management
Sales & Service
Channel Specific
Cross Channel
Marketing
Sales
Customer Mgmt
Servicing
Operations & Execution
Consumer Products
Consumer Loans & Deposits
Cards
Consumer Services
Financial Markets
Investment Management
Wholesale Trading
Market Operations
Corporate Products
Trade Finance
Corporate Banking Products
Corporate Financing & Advisory Services
Cross Product Operations
Payments
Collateral Administration
Account Management
Operational Services
Risk & Compliance
Bank Portfolio & Treasury
Models
Business Analysis
Regulations & Compliance
Business Support
IT Management
Non IT and HR Enterprise Services
Buildings, Equipment and Facilities
Business Command & Control
Finance
Human Resource Management
Knowledge & IP Management
Corporate Relations
Business Direction
Document Management & Archive
Key:
Significant IFX Content Mapping Anticiapted
Some IFX Content Mapping Anticiapted
Minimal IFX Content Mapping Anticiapted