Interactive Financial Exchange

Version 1.0.1

Public Review Draft

Business Message Specification
January 3, 2000

Table of Contents

1 Overview 1-1
1.1 Introduction 1-1
1.1.1 The IFX Business Message Specification 1-1
1.1.2 IFX Implementation Specifications 1-1
1.2 Design Principles 1-1
1.3 Benefits to Financial Institutions and Service Providers 1-3
1.4 Benefits to Consumers and Small Businesses 1-3
2 Structure 2-5
2.1 Definitions 2-5
2.1.1 Biller 2-5
2.1.2 Biller Payment Provider (BPP) 2-5
2.1.3 Biller Service Provider (BSP) 2-5
2.1.4 Client 2-5
2.1.5 Customer 2-5
2.1.6 Customer Payment Provider (CPP) 2-5
2.1.7 Customer Service Provider (CSP) 2-5
2.1.8 Financial Institution (FI) 2-6
2.1.9 Provider 2-6
2.1.10 Server 2-6
2.1.11 Service 2-6
2.1.12 Service Provider (SP) 2-6
2.1.13 May, Should, and Must 2-7
2.2 Documentation Conventions 2-7
2.3 Data Types 2-8
2.3.1 Character 2-9
2.3.2 Narrow Character 2-9
2.3.3 Binary 2-9
2.3.4 Boolean 2-9
2.3.5 YrMon, Date, Time, DateTime, and Timestamp 2-10
2.3.6 Phone Number 2-14
2.3.7 Decimal 2-14
2.3.8 Currency Amount 2-15
2.3.9 Enum 2-16
2.3.10 Long 2-16
2.3.11 Identifier 2-16
2.3.12 Universally Unique Identifier (UUID) 2-16
2.3.13 URL 2-17
2.4 Building Blocks 2-17
2.4.1 Element 2-18
2.4.2 Aggregate 2-18
2.4.3 Field 2-18
2.4.4 Message 2-18
2.4.5 Service 2-23
2.5 Naming Convention 2-24
2.5.1 Rules for Naming in the IFX Specification 2-25
2.6 International Support 2-25
2.6.1 Country Codes 2-25
2.6.2 Character Sets 2-25
2.6.3 Reporting Foreign Exchange Transactions 2-25
2.6.4 Languages and Dialects 2-26
2.6.5 Times and Time Zones 2-26
2.6.6 Country-Specific Tags 2-26

2.7 Versioning and Specification Evolution2-26
2.7.1 Extension and Customization 2-26
2.7.2 Mandatory and Optional Elements 2-27
3 Common Elements and Aggregates3-28
3.1 Common Elements 3-28
3.1.1 Edit Masks <AcctMask>, <SecretMask> 3-28
3.1.2 Customer Login ID <CustLoginId> and Customer Permanent ID <CustPermId> 3-29
3.1.3 Request Identifier <RqUID> 3-29
3.1.4 Customer Service Provider Reference Identifier <CSPRefId> and Service Provider Reference Identifier <SPRefId> 3-30
3.1.5 Service Provider Name <SPName> 3-30
3.1.6 Organization <Org> 3-30
3.1.7 Token <Token> 3-31
3.2 Common Aggregates 3-31
3.2.1 Customer Data 3-31
3.2.2 Organizational Contact Aggregate <OrgContact> 3-33
3.2.3 Biller Data 3-34
3.2.4 Postal Addresses 3-35
3.2.5 Account Data 3-35
3.2.6 Account Balances 3-41
3.2.7 Term Data 3-44
3.2.8 Selection Criteria 3-44
3.2.9 Recurring Model Data 3-45
3.2.10 System Message Data 3-46
3.2.11 Industry Identifier Aggregate <IndustId> 3-49
3.2.12 Secrets Aggregates 3-50
3.2.13 Payment Instrument Aggregate <PmtInst> 3-51
4 Security 4-53
4.1 Channel Level Security vs. Application Level Security 4-53
4.2 Application Level Security in the IFX Specification 4-53
4.3 Security Common Aggregates 4-53
4.3.1 Customer Password Aggregate <CustPswd> 4-53
4.4 Authentication 4-54
4.4.1 Description 4-54
4.4.2 Signon 4-54
4.4.3 Signoff 4-57
5 The Base Service <BaseSvc> 5-58
5.1 Base Service Message Summary 5-58
5.2 Base Service Common Elements and Aggregates 5-60
5.2.1 Service Profile Common Elements 5-60
5.2.2 Service Profile Common Aggregates 5-63
5.2.3 Customer Profile Common Aggregates 5-64
5.3 Service Profile 5-69
5.3.1 Description 5-69
5.3.2 Service Profile Inquiry Message 5-70
5.3.3 Holiday Inquiry Message 5-70
5.4 Customer Profile 5-71
5.4.1 Description 5-71
5.4.2 Customer 5-74
5.4.3 Customer/Service Link 5-82
5.4.4 Service/Account Link 5-88
5.4.5 Disclosure 5-95
5.5 Base Service Profile <BaseSvcProfInfo> 5-97
6 The Banking Service <BankSvc> 6-100
6.1 Description 6-100
6.1.1 Accounts 6-100

6.1.2 Client-Initiated Actions 6-100
6.2 Banking Service Message Summary 6-101
6.3 Banking Service Common Aggregates 6-104
6.3.1 Deposit Account Record Aggregate <DepAcctRec> 6-104
6.3.2 Credit Card Account Record Aggregate <CCAcctRec> 6-104
6.3.3 Common Loan and LOC Account Aggregate <LoanInfoCommon> 6-105
6.3.4 Loan Account Record Aggregate <LoanAcctRec> 6-105
6.3.5 Line of Credit Account Record Aggregate <LOCAcctRec> 6-106
6.3.6 Mortgage Account Record Aggregate <MortAcctRec> 6-106
6.3.7 Bank Account Transaction Record Aggregate <BankAcctTrnRec>6-106
6.3.8 Deposit Account Transaction Record Aggregate <DepAcctTrnRec> 6-107
6.3.9 Credit Card Account Transaction Record Aggregate <CCAcctTrnRec> 6-108
6.3.10 Selection Range Check Number Aggregate <ChkRange> 6-109
6.3.11 Stop Check Record Aggregate <StopChkRec> 6-109
6.3.12 Foreign Exchange Rate Information Aggregate <ForExRateInfo>6-110
6.3.13 Transfer Record Aggregate <XferRec> 6-110
6.3.14 Check Order Record Aggregate 6-112
6.3.15 Deposit Book Order Record Aggregate <DebBkOrdRec> 6-114
6.3.16 Transfer Profile Aggregate <XferProf> 6-114
6.4 Statement and Account Inquiry 6-115
6.4.1 Balance Inquiry 6-115
6.4.2 Account Inquiry 6-116
6.4.3 Deposit Account Statement Inquiry 6-118
6.4.4 Credit Card Statement Inquiry6-119
6.4.5 Deposit Account Transaction Inquiry 6-121
6.4.6 Credit Card Account Transaction Inquiry 6-123
6.4.7 Bank Account Transaction Image Inquiry 6-124
6.4.8 Interest Rate Inquiry 6-125
6.4.9 Bank Account Taxation Inquiry 6-127
6.4.10 Foreign Exchange Rate Inquiry 6-128
6.5 Stop Check 6-129
6.5.1 Stop Check Add 6-129
6.5.2 Stop Check Cancel 6-130
6.5.3 Stop Check Inquiry 6-131
6.5.4 Stop Check Audit 6-133
6.5.5 Stop Check Sync 6-134
6.6 Single Transfer 6-135
6.6.1 Funds Transfer Add 6-135
6.6.2 Funds Transfer Modify 6-136
6.6.3 Funds Transfer Cancel 6-137
6.6.4 Funds Transfer Status Modify 6-138
6.6.5 Funds Transfer Inquiry 6-139
6.6.6 Funds Transfer Audit6-141
6.6.7 Funds Transfer Sync 6-142
6.7 Recurring Transfer Model 6-144
6.7.1 Recurring Transfer Model Add 6-144
6.7.2 Recurring Transfer Model Modify 6-145
6.7.3 Recurring Transfer Model Cancel 6-146
6.7.4 Recurring Transfer Model Inquiry 6-147
6.7.5 Recurring Transfer Model Audit 6-148
6.7.6 Recurring Transfer Model Sync 6-150
6.8 Customer Communications 6-151
6.8.1 Check Order6-151
6.8.2 Deposit Book Order 6-152
6.9 Banking Service Profile <BankSvcProfInfo> 6-153
7 The Pay Service <PaySvc> 7-155

7.1 Description 7-155
7.1.1 Payees 7-155
7.1.2 Payments 7-157
7.2 Pay Service Message Summary 7-161
7.3 Pay Service Common Aggregates 7-164
7.3.1 Standard Payee Record <StdPayeeRec>7-164
7.3.2 Customer Payee Record Aggregate <CustPayeeRec> 7-165
7.3.3 Payment Record Aggregate <PmtRec> 7-167
7.3.4 Payment Authorization Record Aggregate <PmtAuthRec> 7-171
7.3.5 Remittance Record Aggregate <RemitRec> 7-172
7.3.6 Invoice Information Aggregate <InvoiceInfo> 7-174
7.4 Standard Payee 7-175
7.4.1 Standard Payee Inquiry 7-175
7.5 Customer Payee 7-177
7.5.1 Customer Payee Add 7-177
7.5.2 Customer Payee Modify 7-177
7.5.3 Customer Payee Type Modify 7-178
7.5.4 Customer Payee Delete 7-179
7.5.5 Customer Payee Inquiry 7-180
7.5.6 Customer Payee Audit 7-182
7.5.7 Customer Payee Synchronization 7-183
7.6 Single Payment 7-185
7.6.1 Payment Add 7-185
7.6.2 Payment Modify 7-186
7.6.3 Payment Status Modify 7-187
7.6.4 Payment Cancel 7-188
7.6.5 Payment Inquiry 7-189
7.6.6 Payment Audit 7-193
7.6.7 Payment Sync 7-194
7.7 Payment Authorization 7-196
7.7.1 Payment Authorization Add 7-196
7.7.2 Payment Authorization Modify 7-196
7.7.3 Payment Authorization Cancel7-197
7.7.4 Payment Authorization Inquiry 7-198
7.7.5 Payment Authorization Audit 7-199
7.7.6 Payment Authorization Synchronization 7-200
7.8 Remittance 7-201
7.8.1 Remittance Add 7-201
7.8.2 Remittance Modify 7-202
7.8.3 Remittance Status Modify 7-203
7.8.4 Remittance Delete 7-204
7.8.5 Remittance Inquiry 7-205
7.8.6 Remittance Audit 7-206
7.8.7 Remittance Synchronization 7-207
7.9 Recurring Payment Model 7-208
7.9.1 Recurring Payment Model Add 7-208
7.9.2 Recurring Payment Model Modify 7-209
7.9.3 Recurring Payment Model Cancel 7-210
7.9.4 Recurring Payment Model Inquiry 7-211
7.9.5 Recurring Payment Model Audit 7-213
7.9.6 Recurring Payment Model Sync 7-214
7.10 Pay Service Profile <PaySvcProfInfo> 7-216
8 The Bill Presentment Service <PresSvc> 8-218
8.1 Description 8-218
8.1.1 Biller Directory 8-218
8.1.2 Activate Bill Delivery 8-219

8.1.3 Account Inquiry 8-219
8.1.4 Bill Presentment 8-219
8.1.5 Bill Payment 8-222
8.2 Bill Presentment Service Message Summary 8-222
8.3 Presentment Service Common Aggregates 8-223
8.3.1 Biller Record Aggregate <BillerRec> 8-223
8.3.2 Biller Account Identification Information Aggregate <BillerAcctIdInfo> 8-225
8.3.3 Bill Record <BillRec> 8-226
8.4 Presentment Service Message Detail 8-233
8.4.1 Biller Inquiry 8-233
8.4.2 Bill Inquiry 8-235
8.4.3 Bill Status Modify 8-237
8.5 Bill Presentment Service Profile <PresSvcProfInfo> 8-239

1 Overview

1.1 Introduction

The Interactive Financial Exchange (IFX) Specification provides a robust and scalable framework for the exchange of financial data and instructions independent of a particular network technology or computing platform. The information-sharing potential of IFX has been designed to support communication not only between a Financial Institution and its customers, but also between a Financial Institution and its Service Providers, between Financial Institutions, and eventually directly between customers (e.g., “electronic wallet”). This specification supports existing and emerging financial services and is extensible and customizable for future growth.

The IFX Specification has been developed as a cooperative industry effort among major Financial Institutions, Service Providers, and information technology vendors to these institutions and their customers in the small business and consumer markets. This specification builds on previous industry experience, including the OFX and Gold specifications that are currently implemented by major Financial Institutions and Service Providers to enable the electronic exchange of financial data between them and their customers. The IFX Specification provides a comprehensive specification for new financial industry services and software while providing a common strategic direction for the evolution of existing products and services.

The IFX Specification provides Financial Institutions, their Service Provider vendors, and financial software developers for the small business and consumer markets with a generalized model for financial industry data communications. This generalized model consists of two parts:

1. A business-level Message Specification and its associated Data Dictionary, which are contained in Volumes 1 and 2, and
2. Implementation specifications, which are available separately at the IFX Forum website.

1.1.1 The IFX Business Message Specification

This document provides the financial services industry with a common set of *Business Messages* that may be used to provide message-processing services across multiple organizations and networks. This document defines the information that must be sent in a message request and in a message response from a business perspective and provides message semantics for each request and response pair. This document provides the basis for business-level information flow between Financial Institutions, their customers, and third-party Service Providers.

For example, an individual account holder may bank from home using a secure connection over the public Internet to connect to their Financial Institution. The same request message sent from the personal computer in the home may enter the Financial Institution’s private data network for processing or be routed to a third-party Service Provider that processes that message on behalf of the Financial Institution. Regardless of the organization that actually processes the message or what computing and network architecture that organization has installed, the customer receives a response message with standard semantics.

1.1.2 IFX Implementation Specifications

Complementary documents called *Interactive Financial Exchange Implementation Specifications* are expected to provide additional detail on how the business messages defined in this document may be physically represented. Each IFX Implementation Specification is an agreement between vendors and the financial services industry on how to implement the business messages defined in this document. IFX Implementation Specifications also provide the basis for interworking among products and services from various vendors and custom software developed by Financial Institutions and Service Providers.

1.2 Design Principles

The IFX Specification has been designed to meet the following principles:

Support a Broad Range of Financial Activities—The IFX Specification is envisioned to incorporate the broad feature sets of the existing OFX and Gold specifications. Version 1.0 provides the following functions:

- Bank account balances, account information, statement download for deposit and loan accounts;
- Credit card statement download;
- Funds transfers, including recurring transfers;
- Consumer payments, including recurring payments;
- Business payments, including recurring payments;
- Bill presentment; and
- Customer service.

Support Financial Communications Among a Broad Range of Parties—The IFX Specification supports financial communications among:

- Banks
- Brokerage houses (future)
- Service Providers
- Financial advisors (future)
- Small businesses
- Consumers

Support a Broad Range of Client Devices—The IFX Specification allows Financial Institutions to support customers using a broad range of client devices, including, but not limited to:

- World Wide Web access using any standard Web Browser software,
- Personal Computers with Personal Financial Manager (PFM) software,
- Voice Response Units (VRUs) that provide Bank by Phone services,
- Automated Teller Machines (ATMs), and
- Consumer Handheld Devices such as Personal Digital Assistants (PDAs) or Mobile Telephones with data capabilities.

Support Customers Using Multiple Client Devices—The IFX Specification allows a customer to use multiple client devices to interact with a Financial Institution. All devices get the same data for that customer and provide the customer with a consistent experience. The IFX Specification contains requests, which allow an intelligent client to discover what messages the customer has performed using other client devices.

Flexible—The IFX Specification is designed to provide Financial Institutions and Service Providers with the flexibility they need to rapidly develop, test, and deploy new services. The specification is intended to specify the minimum necessary functionality to provide reliable interactions between systems owned and maintained by customers, their Financial Institutions, and their Service Providers.

Customizable—The IFX Specification allows Financial Institutions and Service Providers to add custom elements, aggregates, or entire messages to rapidly deploy new services or add functionality to existing services. Custom elements, aggregates, and messages should be registered with the IFX Specification governance organization to protect against name collisions. However, registration of custom objects does not obligate any organization to use them in order to be in compliance with the specification.

Extensible—The IFX Specification has been designed to allow for constant evolution through the addition of standardized services that may be easily implemented by Financial Institutions and their customers.

Open—The IFX Specification is publicly available. Solutions for financial communications based on the specification may be built by anyone, independent of any specific technology, vendor, or Financial Institution. IFX is currently maintained through cooperative industry effort and will be evolved into a formal standard over time.

***Robust*—**The IFX Specification is used for the execution of important financial messages and the communication of sensitive financial data. The specification provides customers with confirmation that messages have occurred as planned and notification when scheduled messages fail.

Secure—The IFX Specification provides a secure framework for the development of secure online financial services. IFX relies on industry standard mechanisms to provide secure channels between client and server and provides application-level authentication of customers and Financial Institution and Service Provider staff. Note that application-level authentication between different Service Providers for Electronic Bill Presentment and Payment may be provided through alternate means.

Support Batch and Interactive Sessions—The IFX Specification may be implemented using either batch or interactive session management. The business-level messages are not biased towards either batch or interactive sessions.

International Support—The IFX Specification incorporates significant features for international support, including multiple currencies, and languages. Additional requirements for international support will be addressed as the specification evolves and international requirements are identified.

Platform Independent—The IFX Specification makes no assumptions about the hardware or software available as a client or server. IFX may be implemented on any computing platform.

Transport Independent—The IFX Specification makes no assumptions about the network used for the transport of business-level messages. IFX Implementation Specifications provide details on transport for a specific type of network.

1.3 Benefits to Financial Institutions and Service Providers

Customer Acquisition and Retention—Standardization of message sets for financial message processing will remove technical barriers that currently limit an FI's ability to acquire customers for online banking services. These standards will level the competitive playing field for electronic delivery of financial services and allow Financial Institutions to serve new customers for whom appropriate channels have not previously existed. Broader vendor support for these message sets and the associated implementation specifications will stimulate end user adoption of online banking through the availability of a diverse set of client applications that interoperate with any FI that adopts this specification.

Flexibility—Because the semantics and syntax for the request and response message pair are defined end-to-end in IFX and the same message is passed regardless of the network technology, Financial Institutions and Service Providers are not locked into a single vendor or technology. They may find other ways to meet their business objectives if their service is not well supported by a vendor or network provider.

Manageability—Standardization of message sets for financial message processing will remove technical barriers that currently limit an FI's ability to acquire customers for online banking services. This common framework will provide Financial Institutions and their outsourced Service Providers with significant benefits in terms of their ability to manage the development and operation of a portfolio of services for their customers.

Cost Savings—The increased flexibility and manageability of solutions based on the IFX Specification will directly affect the profitability of an online financial service. Financial Institutions and Service Providers that invest in solutions based on IFX will see their investments returned quickly through operational cost savings.

1.4 Benefits to Consumers and Small Businesses

Reliability and Performance—Individual consumers and small businesses will receive benefits in reliability and performance through the increased integration of their software with a Financial Institution's systems and networks. The IFX Specification is the product of an unprecedented level of cooperation between developers of consumer and small business financial applications and financial industry high-volume message processing experts.

Consistency of Experience with a Financial Institution—Consumers of online financial services will notice significant improvements in the consistency of their interactions with the Financial Institution through multiple channels as the organizations take advantage of the improved service manageability and flexibility enabled by use of the IFX.

C***ommon Standard Across all Financial Institutions***—Small Businesses and Consumers will also benefit from increased financial industry use of the IFX by their increased ability to manage their relationships with Financial Institutions using off-the-shelf software.

2 Structure

This section describes a number of important foundations of the IFX Specification, including terminology, data types, elements, aggregates, messages, services, authentication realms, naming conventions, versioning and governance, usage rules, and documentation conventions.

Definition of "optional"—A client or server may include optional fields within a message. There is no requirement that the receiver process optional fields. In order to maintain upward compatibility, the receiver of a message must ignore tags that it does not support. Note that it is not unusual for an optional field to be required based on the context. This condition is documented by the words "*but see Description*" in the Usage column.

2.1 Definitions

The following terms are used extensively in IFX Specification documentation and may have very specific meanings within this context.

2.1.1 Biller

A *Biller* is a company or organization that sends a Bill or Statement to a customer, usually a request for payment for a product or service.

2.1.2 Biller Payment Provider (BPP)

A *Biller Payment Provider* (or *BPP*) is an agent (usually a financial institution) of the Biller that originates and accepts payments on behalf of the Biller.

2.1.3 Biller Service Provider (BSP)

A *Biller Service Provider* (or *BSP*) is an agent of the Biller that provides an electronic bill presentment and payment service for the Biller

2.1.4 Client

Client refers to the sender of an IFX Request Message. The client may be a computer system that a Customer is logged into, or it may be some kind of proxy device that is making IFX Requests on behalf of the Customer. This scenario is typical in the case of a Customer using a Web browser to perform financial messages. The Web server may be communicating using IFX Messages to some back end system. In this case, the Web server is considered a proxy client and the back end system is the server. It should be noted that clients might also be Service Providers for certain messages, such as the Pay/No-Pay message and Reverse Pay Decision message.

2.1.5 Customer

A *Customer* is an individual or small business that is a consumer of financial services provided by a Financial Institution.

2.1.6 Customer Payment Provider (CPP)

A *Customer Payment Provider* (or *CPP*) is an agent (usually a financial institution) of the Customer that originates payments on behalf of the Customer.

2.1.7 Customer Service Provider (CSP)

A *Customer Service Provider* (or *CSP*) is an agent of the Customer that provides an interface directly to customers, businesses, or others for bill presentment. A CSP enrolls customers, enables presentment, and provides customer care, among other functions.

2.1.8 Financial Institution (FI)

A *Financial Institution* (or *FI*) is an organization that provides branded financial services to customers. Financial Institutions develop and market financial services to individual and small business customers. Financial Institutions may serve as the processor for their own services or may choose to outsource processing to a Service Provider. In an effort to group the services different providers provide, the Financial Institution is also referred to as a Financial Institution in this document.

2.1.9 Provider

When the term *provider* is used generically, typically qualified by a service name (e.g., Pay provider), it refers to an organization that processes messages that support financial services provided to individual and small business customers. A provider may be either of the following:

- A Financial Institution that has chosen to perform its own message processing in support of its services *or*
- A Service Provider that performs message processing on behalf of a Financial Institution that has decided to outsource this function.

2.1.10 Server

Server refers to a system that receives IFX Request Messages and responds with IFX Response Messages. For clarity in the documentation, the server is assumed to be an endpoint that composes the entire response message and sends it back to the client. In the real world, systems may be implemented with multiple stages of message processing that are transparent to the client. As long as the complete response message is delivered to the client, the protocol will work as designed.

2.1.11 Service

Service specifically refers to a collection of related messages. For example, the Bank service encompasses banking messages such as requesting bank statement, initiating stop checks, etc. This definition correlates to an application-level concept of service that refers to a single function or a collection of similar functions that are branded and marketed to individuals and small businesses by a Financial Institution or Service Provider. Within this specification, each SP maintains a list of the services it supports. Services that are currently defined in this specification are Banking (Bank), Payment (Pay), and Presentment (Pres). Customer Service and a few other functions including Service Profile are grouped together in a chapter called Base Service (Base). Additional services may be defined and implemented by Financial Institutions or Service Providers as extensions to this specification.

2.1.12 Service Provider (SP)

A *Service Provider* (or *SP*) is an organization that provides services to an individual or to other organizations. An example of a Service Provider is a message processor for a Financial Institution that has chosen to outsource its message processing for a particular service. Service Providers typically provide services for multiple individuals or organizations. A Biller may be considered a Service Provider; however, in this document the term "Service Provider" is used to refer generically to an FI, CSP, CPP, BSP and/or BPP. Organizations that provide services to end user customers are referred to specifically as Financial Institutions or CSPs where possible.

> ***NOTE:*** *The labels CSP, CPP, BSP and BPP are used to define a collection of functions and responsibilities. They do not necessarily refer to specific physical or business entities. Any entity wishing to perform a particular role needs to address the issues and responsibilities defined for that role. It is also understood that one entity may perform more than one role, or that an entity may wish to outsource one or more functions of a role to other entities. For example, some Billers may serve as their own BSPs, some Financial Institutions may perform the role of both CSP and CPP, and some CSPs may outsource functions such as customer care to other service providers. For more information about the responsibilities of the various roles, see the Electronic Bill Presentment and Payment Business Practices at http://www.nacha.org/billpay/businesspractices.htm.*

2.1.13 May, Should, and Must

The terms *may*, *should*, and *must* are used frequently within this specification.

May indicates that the described behavior is not required for IFX compliance, nor is there any preferred behavior. Generally, the behavior is described to inform of possible behaviors of which client or server designers and developers should be aware.

Should indicates that, while the described behavior is not required for IFX compliance, it *is* preferred. Generally, the behavior is described to provide a better experience for the communicating parties.

Must indicates that the described behavior is required for IFX compliance.

2.2 Documentation Conventions

Documentation conventions in the specification include:

- All documentation is written and maintained in US English.
- Required elements and aggregates have “Required” in the Usage column.
  - In a request message, “Required” means that the client must include the element.
  - In a response message, “Required” means that the server must return the element if the message is successful, with the exception of <Status>, <CustId> and <RqUID>, which must always be returned.
- Optional elements and aggregates have “Optional” in the Usage column. If there are specific cases where an optional element or aggregate may be required, it is noted by “*but see Description*” in the Usage column, to call attention to this condition.
- For successful messages (i.e., responses with a <Status> <Severity> of Info or Warn), a server must echo all fields provided in the request that are indicated in the response by “Echoed” in the Usage column.
- Some elements and aggregates are required only if the Service Provider indicates this through the Service Profile for the service. These elements and aggregates have “Profiled requirement” in the Usage column.
- Some elements have valid values defined by the Service Profile. These elements have “Profiled values” in the Usage column.
- Some elements and aggregates are only valid if support for them is indicated in the Service Profile. These elements have “Profiled support” in the Usage column.
- Repeating elements and aggregates may appear more than once, and are indicated by “Repeating” in the Usage column.
- An “inclusive or” condition requires at least one of the members of the group to be present, and is indicated with “OR” in the Usage column. A thin dashed line separates the members of the OR, while a thicker dotted line indicates the boundaries of the entire OR group.
- An “exclusive or” condition requires one, and only one, of the members of the group to be present, and is indicated with “XOR” in the Usage column. A thin dashed line separates the members of the XOR, while a thicker dotted line indicates the boundaries of the entire XOR group.
- An “and” condition requires all or none of the members of the group to be present, and is indicated with a “AND” in the Usage column. A thin dashed line separates the members of the AND, while a thicker dotted line indicates the boundaries of the entire AND group.
- Indentation indicates that the indented element or aggregate is contained in the most recent aggregate that has one less indentation.

Tag	Type	Usage	Description

Tag	*Type*	*Usage*	*Description*
<RequiredElement>	C-12	Required	Required Element. Occurs exactly once in each Request, or exactly once in each successful Response.
<OptionalElement>	C-36	Optional	Optional Element. Occurs zero or one times.
<OptionalRepeating>	C-12	Optional Repeating	Optional Repeating Element. Occurs zero or more times.
<ProfileAggregate>	Aggregate	Optional Profiled	Profiled Aggregate. Occurs zero or one times. Usage is controlled by an entry in the Services Profile.
<EchoAggregate>	Aggregate	Optional Echoed	Echo Aggregate. Occurs zero or one times. Must be included in the response message if it was received in the request.
<Element1>	Long	Optional AND	Optional and.
<Element2>	Decimal	Optional AND	Two valid cases (1) both <Element1> and <Element2>, (2) neither.
<Element1>	Long	Required OR	Inclusive or.
<Element2>	Decimal	Required OR	Three valid cases (1) <Element1>, (2) <Element2>, (3) both <Element1> and <Element2>.
<Element1>	Long	Optional XOR	Optional exclusive or.
<Element2>	Decimal	Optional XOR	Three valid cases: (1) neither <Element1> nor <Element2>. (2) <Element1>, but not <Element2>. (3) <Element2>, but not <Element1>.
<Element1>	Long	Required XOR	Required Exclusive or.
<Element2>	Decimal	Required XOR	Two valid cases: (1) <Element1>, but not <Element2>. (2) <Element2>, but not <Element1>.

2.3 Data Types

The IFX Specification is designed around a small number of data types that are used to represent all data passed between clients and servers using the messages defined in this specification. All information elements are based on these data types. Supported data types are:

Character

Narrow Character

Binary

Boolean

YrMon, Date, Time, DateTime, and Timestamp

Decimal

Currency Amount

Closed Enum

Open Enum

Long

Identifier

Phone Number

Universally Unique Identifier (UUID)

URL

IFX defines the semantics and logical attributes of each data type. The physical representations are defined within the implementation specifications.

2.3.1 Character

Character indicates an element that allows character data up to a maximum number of characters. The number after the hyphen specifies the maximum number of characters. For example, C-12 specifies an element of characters with maximum length 12 characters. C- indicates an element with no maximum length. It is expected that *character* type elements may contain multibyte representations of characters in some implementations, depending on the allowable character sets.

2.3.2 Narrow Character

Elements of type *Narrow Character* are elements of *character* data type with one additional restriction. The number after the hyphen specifies the maximum number of characters. For example, NC-12 specifies an element of narrow characters with maximum length 12 characters. *Narrow Character* type elements must always contain only single-byte representations of characters, even in implementations that support a multibyte character representation. Limiting these elements to values that may always be expressed in a single-byte character encoding significantly reduces implementation complexity on both clients and servers.

2.3.3 Binary

The *Binary* data type is a compound type consisting of three logical elements:

Tag	*Type*	*Usage*	*Description*
<ContentType>	Open Enum	Optional	Specified in IETF RFC 2046.
<BinLength>	Long	Required	Identifies the size of the binary data in number of bytes.
<BinData>	Raw Binary Data	Required	Binary data.

2.3.4 Boolean

Boolean indicates a logical TRUE or FALSE condition. The physical representation of Boolean data is specified by each implementation specification corresponding to this message specification.

2.3.4.1 Boolean Conventions and Selection Criteria

The general usage of Booleans within the IFX specification is:

<TagName> (optional)

- If TRUE, then...
- If FALSE or omitted, then...

This usage of Booleans in IFX allows optional Booleans to be added in future revisions while maintaining upward compatibility.

There are instances where it is considered too error prone to allow a default of FALSE. In these cases the Boolean is required.

The convention for the use of selection criteria for read messages is:

For Booleans:

<TagName> (optional)

- IF TRUE, then the selected set must include at least those items where the condition is TRUE.
- IF FALSE, then the selected set must include at least those items where the condition is FALSE.
- If omitted, then this item is ignored as a selection criterion.

Usage notes specify when a Boolean is used as a selection criterion.

For others:

<TagName> (optional)

- If present, the selected set must include at least those items where the value in the request matches a value for the corresponding tag in the searched region.
- If absent, the selected set must include items with any value, including NULL, for the corresponding tag in the searched region.

2.3.5 YrMon, Date, Time, DateTime, and Timestamp

The IFX Specification includes five time-related compound data types: YrMon, Date, Time, DateTime, and Timestamp. In all types that describe Date information, IFX 1.0.1 uses the Gregorian calendar. Other calendars may be considered for future versions as requirements are identified. Data types including time information refer to a 24-hour clock.

All date and time types include (with the largest units given first): year, month, day, hour, minute, second, and fractions of a second. Any particular type may include a subset of these possible values. Types including time information (hour, minute, etc.) may also include an offset from Coordinated Universal Time (UTC).

As a general rule for *date* and *time* compound data types, values may be entered that omit the smallest logical elements. In every case, the value is taken to mean the same thing as if the minimum values (such as zeroes) were included. (The default is always the start of an otherwise ambiguous range for types other than *YrMon*.) For example, a *DateTime* value omitting the time portion means the start of the day (12:00 midnight). Note that time zone qualifiers (in *time* and *DateTime* values) are an exception to this rule, as they may be included even if times are not specified to the millisecond.

The logical elements appearing in each of these compound data types are summarized below. “Required” means that the element must occur in all instances of the data type. “Recommended” means that the element should be included in all instances of the data type. “Optional” elements may be omitted from an instance of the data type. Optional elements must be included if smaller elements are to be included. For example, month must not be omitted from a *date* value if day is included.

	Contains	***YrMon***	***Date***	***Time***	***DateTime***	***Timestamp***
Year	YYYY 0000-9999	Required	Required	N/A	Required	Required
Month	MM 1-12	Required	Optional	N/A	Required	Required
Day	DD 1-31	N/A	Optional	N/A	Required	Required
Hours	HH 0-23	N/A	N/A	Required	Optional	Required
Minutes	MM 0-59	N/A	N/A	Optional	Optional	Required
Seconds	SS 0-60	N/A	N/A	Optional	Optional	Required
Fractional Seconds	XXX (minimum) Precision is determined by the implementation	N/A	N/A	Optional	Optional	Optional
UTC offset (time zone indication)	Minutes -720 to +720	N/A	N/A	Recommended	Recommended	Recommended

2.3.5.1 YrMon

Elements of data type *YrMon* contain an indication of a particular month. This data type describes a unique period of time (not a repeating portion of every year). This may (for example) be used to describe an expiration date for a credit card. In that case, the period represented is actually a shorthand for the last day (or millisecond or whatever minimum interval) of the month described.

Logically, the *YrMon* data type must contain a month and 4-digit year. Unlike the other date and type types, values of this type must ***not*** be shortened by any omission of its portions: Both the year and month are required.

YrMon Data type	*Type*	*Usage*	*Description*
<Year>	Long	Required	4-digit year value.
<Month>	Long	Required	Number of the represented month. Value must be within the range 1 through 31.

2.3.5.2 Date

Elements of data type *Date* contain an indication of a particular day. This data type describes a unique period of time, normally 24 hours (not a repeating portion of every year).

Logically, this data type must contain a 4-digit year, and may contain a month number, and day number.

Date Data type	*Type*	*Usage*	*Description*
<Year>	Long	Required	4-digit year value.
<Month>	Long	Optional but see Description	Number of the represented month. Value must be within the range 1 through 12. Must be included if <Day> is included. If absent, the value is assumed to be 1 (January).
<Day>	Long	Optional	Number of the represented day. Value must be within the range 1 through 31. If absent, the value is assumed to be 1.

2.3.5.3 Time

Elements of data type *Time* contain an indication of a particular time during a date. This data type describes a repeating portion of a day. That is, each time described (ignoring leap seconds) occurs once per calendar date. In the IFX specification, it is required that a *time* data type be able to represent a specific period with indefinite precision. Milliseconds are the minimum required precision of the *time* data type.

A time represented using this data type must not be ambiguous with respect to morning and afternoon. That is, the time must occur once and only once each 24-hour period.

In addition, the *Time* data type must not be ambiguous with respect to location at which the time occurs. If unspecified, the time zone defaults to Coordinated Universal Time (UTC). Generally, use of a specific time zone in the representation is preferred. The time zone should always be specified to avoid ambiguous communication between clients and servers.

Time Data type	***Type***	***Usage***	***Description***
<Hour>	Long	Required	Number of the represented hour. Value must be within the range 0 through 23.
<Minute>	Long	Optional but see Description	Number of the represented minute. Value must be within the range 0 through 59. Must be included if <Second> is included. If absent, the value defaults to 0.
<Second>	Long	Optional but see Description	Number of the represented second. Value must be within the range 0 through 60. The value “60” is used only to represent leap seconds. Must be included if <Fraction> is included. If absent, the value defaults to 0.
<Fraction>	Long	Optional	Number of represented microseconds. Value must be within the range 0 through 999,999. Particular implementations may choose to allow representations of smaller fractions. If absent, the value defaults to 0.
<UTCOffset>	Long	Optional	Offset from UTC in minutes. Value must be within the range -720 through +720. Value is typically a multiple of 60 (an exact number of hours), but the offset may also include half and quarter hours. Generally should be included. If absent, the value defaults to 0, i.e., UTC.

2.3.5.4 DateTime

Elements of data type *DateTime* contain all of the information expressed by the *date* and *time* data types. All ambiguities mentioned with *Date* and *Time* (see Sections 2.3.5.2 2.3.5.3) should be resolved in a similar fashion here.

DateTime Data type	***Type***	***Usage***	***Description***
<Year>	Long	Required	4-digit year value.
<Month>	Long	Required	Number of the represented month. Value must be within the range 1 through 12.
<Day>	Long	Required	Number of the represented day. Value must be within the range 1 through 31.

DateTime Data type	*Type*	*Usage*	*Description*
<Hour>	Long	Optional but see Description	Number of the represented hour. Value must be within the range 0 through 23. Must be included if <Minute> is included. If absent, the value defaults to 0.
<Minute>	Long	Optional but see Description	Number of the represented minute. Value must be within the range 0 through 59. Must be included if <Second> is included. If absent, the value defaults to 0.
<Second>	Long	Optional but see Description	Number of the represented second. Value must be within the range 0 through 60. The value "60" is used only to represent leap seconds. Must be included if <Fraction> is included. If absent, the value defaults to 0.
<Fraction>	Long	Optional	Number of represented microseconds. Value must be within the range 0 through 999,999. Particular implementations may choose to allow representations of smaller fractions. If absent, the value defaults to 0.
<UTCOffset>	Long	Optional	Offset from UTC in minutes. Value must be within the -720 through +720. Value is typically a multiple of 60 (an exact number of hours), but the offset may also include half and quarter hours. Generally should be included. If absent, the value defaults to 0, i.e., UTC.

2.3.5.5 Timestamp

Elements of data type *Timestamp* contain the same information as *DateTime* values. Unlike that data type, *Timestamp* information is not intended to have meaning at the other end of the communication. In addition, microseconds are the minimum required precision of the time portion of this data type.

The intent here is to describe a type identical to *DateTime* but without semantic meaning between two machines. The general *DateTime* data type has meaning on both ends of the protocol (even though time synchronization is not required by this specification). *Timestamp* indicates an exact point in time with respect to the generating application.

For example, a *Timestamp* value may be generated at a server when creating an audit response. The client application may return that value to the server in later requests, but the client software should not interpret the information.

Timestamp Data type	*Type*	*Usage*	*Description*
<Year>	Long	Required	4-digit year value.
<Month>	Long	Required	Number of the represented month. Value must be within the range 1 through 12.
<Day>	Long	Required	Number of the represented day. Value must be within the range 1 through 31.
<Hour>	Long	Required	Number of the represented hour. Value must be within the range 0 through 23.
<Minute>	Long	Required	Number of the represented minute. Value must be within the range 0 through 59.
<Second>	Long	Required	Number of the represented second. Value must be within the range 0 through 60. The value "60" is used only to represent leap seconds.

Timestamp Data type	*Type*	*Usage*	*Description*
<Fraction>	Long	Required	Number of represented microseconds. Value must be within the range 0 through 999,999. Particular implementations may choose to allow representations of smaller fractions.
<UTCOffset>	Long	Optional	Offset from UTC in minutes. Value must be within the range -720 through +720. Value is typically a multiple of 60 (an exact number of hours), but the offset may also include half and quarter hours. Generally should be included. If absent, the value is assumed to be 0, i.e. time is assumed to be UTC.

2.3.6 Phone Number

Phone Number indicates a string of up to 32 narrow characters in length (NC-32). It must begin with a plus sign "+" followed by country code, a hyphen, city/area code, another hyphen, then the local phone number. If a PBX extension is to be included, it must appear at the end of the field, separated from the rest of the telephone number by a plus sign.

For example, "+1-800-5551212+739" indicates PBX extension 739 at phone number 5551212 within area code 800 of North America (country code 1).

2.3.7 Decimal

Decimal indicates a numeric value that meets the following rules:

- The value is up to fifteen (15) digits in length, excluding any punctuation (e.g., sign, decimal, currency symbol, etc.).
- The value is not restricted to integer values and has a decimal point that may be placed anywhere from the left of the leftmost digit to the right of the rightmost digit (e.g., +.12345678901234 is acceptable while 12345678901234567 is not).
- The sign is always optional. If it is absent, the value is assumed to be positive.

2.3.8 Currency Amount

A *Currency Amount* is a compound data type consisting of four logical elements:

Tag	*Type*	*Usage*	*Description*
<Amt>	Decimal	Required	Amount.
<CurCode>	NC-3	Optional but see Description	Currency code. Required if <CurRate> is specified.
<CurRate>	Decimal	Optional AND	Exchange rate. The ratio of the foreign currency and the currency of the account. The <CurConvertRule> specifies which amount is the numerator and which is the denominator.
<CurConvertRule>	Closed Enum	Optional AND	Currency Conversion Indicator. Valid values are Direct and Indirect. See below for more information.

All monetary amounts in the IFX Specification are handled with the *Currency Amount* data type. When included, this data type contains a decimal value for the amount, an optional three-letter currency code defined in ISO-4217, an optional currency rate, and an optional exchange rate indicator. When Currency Code is not included, Amount is assumed to represent the currency of the account (in particular the "from" account in the case of a transfer or payment).

The valid values for the exchange rate indicator are Direct and Indirect. The indirect rate is the amount of foreign currency per one unit of base currency. The direct rate is the amount of base currency per one unit of foreign currency. If the exchange rate is present, the currency indicator must be specified. The currency indicator identifies the way in which the rate is calculated.

Derive USD equivalent from a specified foreign amount:

- If the rate is Indirect, divide the foreign amount by the rate,
- If the rate is Direct, multiply the foreign amount by the rate.

Derive foreign amount from a specified USD amount:

- If the rate is Indirect, multiply the USD amount by the rate,
- If the rate is Direct, divide the USD amount by the rate.

Examples (USD is the base):

USD GBP	Indirect = .6037	Direct = 1.6590
USD DEM	Indirect = 1.6866	Direct = .5929

The following table is intended to illustrate when the elements within the *Currency Amount* data type are used.

Use Cases	*<Amt>*	*<CurCode>*	*<CurRate> and <CurConvertRule>*
Initiating a transaction…			
In the currency of the FROM account	Required		
In a currency different than the FROM account	Required	Required	Only used when a pre-committed exchange rate is used, see the Foreign Exchange Rate message
Reporting a transaction e.g. within a statement…			
In the currency of the account.	Required		
Originating in a currency different from the currency of the account.	Required	Required	Required

2.3.9 Enum

Enum is a Narrow Character type that has a limited number of specified valid values, each of which is represented by a tag of up to 80 characters each. The *Enum* data type is either a *Closed Enum* or an *Open Enum*. Adding a value to a Closed Enum requires a spec update, while adding a value to an Open Enum only requires out-of-band agreement by the end points.

2.3.9.1 Closed Enum

A *Closed Enum* is an element where a number of valid values are defined within this specification. All other values should be rejected as invalid.

2.3.9.2 Open Enum

An *Open Enum* is an element where a number of valid values are defined within this specification, but other values should not be rejected as invalid by any system other than the final message destination. Open Enums provide a mechanism for a client and final destination server to communicate with values that may be known to both endpoints but not to all intermediate servers that route the message. If the client sends a value that was not listed in the Service Profile as being a supported value in a Profiled Values element, the server must respond with the most specific response code possible (for example, “Frequency not supported”). Otherwise, if the client or server receives a value that it does not recognize, it must be treated as the type “other.”

Open Enums are typically used for elements related to system message processing and have been defined as open to support extensibility and customization of the specification.

2.3.10 Long

Long is an integer value, which may be positive, negative, or zero, with values ranging from –2147483648 to 2147483647.

2.3.11 Identifier

Object identifiers in the IFX Specification are of the data type “Identifier.” This is a Narrow Character data type with a maximum length of 36.

2.3.12 Universally Unique Identifier (UUID)

UUID elements are Narrow Character with a maximum length of 36.

A UUID is an identifier that is unique across both space and time, with respect to the space of all UUIDs. To be precise, the UUID consists of a finite bit space. Thus, the time value used for constructing a UUID is limited and will roll over in the future (approximately at A.D. 3400, based on the specified algorithm). A UUID may be used for multiple purposes, from tagging objects with an extremely short lifetime to reliably identifying very persistent objects across a network. The following information on UUID is based on Internet-Draft <leach-uuids-uuids-01.txt>.

> The generation of UUIDs does not require that a registration authority be contacted for each identifier. Instead, it requires a unique value over space for each UUID generator. This spatially unique value is specified as an IEEE 802 address, which is usually already available to network-connected systems. This 48-bit address may be assigned based on an address block obtained through the IEEE registration authority. This section of the UUID specification assumes the availability of an IEEE 802 address to a system desiring to generate a UUID, but if one is not available, section 4 specifies a way to generate a probabilistically unique one that cannot conflict with any properly assigned IEEE 802 address.3.1 Format. In its most general form, all that may be said of the UUID format is that a UUID is 16 octets, and that some bits of octet 8 of the UUID called the variant field (specified in the next section) determine finer structure.
>
> For use in human-readable text, a UUID string representation is specified as a sequence of fields, some of which are separated by single dashes. Each field is treated as an integer and has its value printed as a zero-filled hexadecimal digit string with the most significant digit first. The hexadecimal values a to f inclusive are output as lower case characters, and are case insensitive on input. The sequence is the same as the UUID constructed type. The formal definition of the UUID string representation is provided by the following extended BNF:

UUID	<time_low> “-” <time_mid> “-” <time_high_and_version> “-” <clock_seq_and_reserved> <clock_seq_low> “-” <node>
time_low	4*<hexOctet>
time_mid	2*<hexOctet>
time_high_and_version	2*<hexOctet>
clock_seq_and_reserved	<hexOctet>
clock_seq_low	<hexOctet>
node	6*<hexOctet
hexOctet	<hexDigit> <hexDigit>

hexDigit	zero | “1” | “2” | “3” | “4” | “5” | “6” | “7” | “8” | “9”| “a” | “b” | “c” | “d” | “e” | “f” | “A” | “B” | “C” | “D” | “E” | “F”

The following is an example of the string representation of a UUID:

f81d4fae-7dec-11d0-a765-00a0c91e6bf6

2.3.13 URL

An element of the Uniform Resource Locator *URL* data type specifies the URL where a customer may access information. A URL is of the Narrow Character data type with a length of 1024 Characters (NC-1024). The format of a URL begins with a string that identifies which protocol is to be used to access the information, such as “http://”.

2.4 Building Blocks

The IFX Specification is constructed using the following building blocks:

Element The most basic unit of data in the IFX Specification to define a single piece of information (of a specific data type) that is passed between the client and the server.

Aggregate A group of related elements to provide a mechanism for coding logic rules and a convenient method to refer to related information using a single name.

Field Used to define either an atomic element or aggregate.

Message A collection of elements and/or aggregates to be passed from the client to the server (Request Message) or from the server to the client (Response Message).

Service A single function or a collection of similar functions that are branded and marketed to individuals and small businesses by a financial institution.

2.4.1 Element

An *Element* is the most basic unit of data in the IFX Specification. An element is defined based on one of the supported data types to define a single piece of information that is passed between the client and server. An element is named according to specific rules and has a definition associated with it to provide additional information on what it contains. An element may also have some usage rules associated with it, which describe how the client and server interact with the element.

For example, the <ChkNum> element is based on the Character data type with a maximum of 12 characters and specifies a Check Number. Depending on where it is used in the specification, it may be provided by the client or the server.

2.4.2 Aggregate

A number of related elements may be grouped together into an *Aggregate*. An aggregate provides a mechanism for coding logic rules (“element 1 or element 2 must be provided”) and also provides a convenient way for programmers to specify all of the related information by using a single name.

Note that multiple aggregates may use the same structure. This may occur to allow use of a more descriptive name, e.g., <BillingAddr> in the <PresAcctId> aggregate has the same structure as the <PostAddr> aggregate. This may also occur when a unique name must be used due to the same aggregate being used more than once within a request/response message, e.g. TO/FROM account within a transfer message.

2.4.3 Field

For the purpose of this documentation, the term *Field* is used to define either an atomic element or an aggregate.

2.4.4 Message

A *Message* is a collection of elements and/or aggregates, which is passed from the client to the server (Request Message) or from the server to the client (Response Message).

A response message is typically a superset of the request that echoes back the information included in the request and adds new information as appropriate to the message being conducted. All requests are sent from client to server and all responses are sent from server to client.

IFX uses several common message types to perform specific functions. Within IFX, the following naming conventions are used, where the messages associated with objects of type *xxx* include:

- Add request <*xxx*AddRq> and response <*xxx*AddRs>
- Modify request <*xxx*ModRq> and response <*xxx*ModRs>
- Delete request <*xxx*DelRq> and response <*xxx*DelRs>
- Cancel request <*xxx*CanRq> and response <*xxx*CanRs>
- Inquiry request <*xxx*InqRq> and response <*xxx*InqRs>
- Audit request <*xxx*AudRq> and response <*xxx*AudRs>

2.4.4.1 Add Message

The Add IFX message has a name structure of <*xxx*AddRq>/<*xxx*AddRs>. It is used to create a new instance of object *xxx* (such as creating a new payment using <PmtAddRq>).

2.4.4.2 Modify Message

The modify IFX message has a name structure of <*xxx*ModRq>/<*xxx*ModRs>. It is used to modify an existing instance of object *xxx* (such as modifying an existing payment using <PmtModRq>.

A client modifies a record on the server using the appropriate Modify request message and replaces the *entire* existing object (all properties) with the newly-submitted object. Therefore, within a modify request message, all properties of the object that are intended to be stored must be included, even if the values are not modified.

2.4.4.3 Delete and Cancel Messages

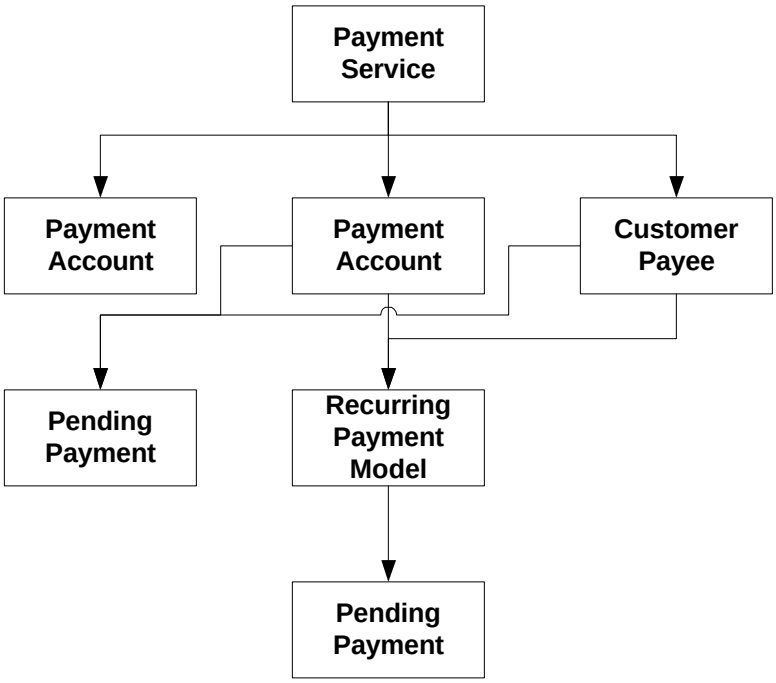
The delete and cancel IFX messages have a name structure of <*xxx*DelRq>/<*xxx*DelRs> and <*xxx*CanRq>/<*xxx*CanRs>, respectively. They are used to delete an existing instance of object *xxx* (such as deleting a payee from the customer payee list using <CustPayeeDelRq>), or to cancel an existing scheduled object (such as canceling a pending payment using <PmtCanRq>.

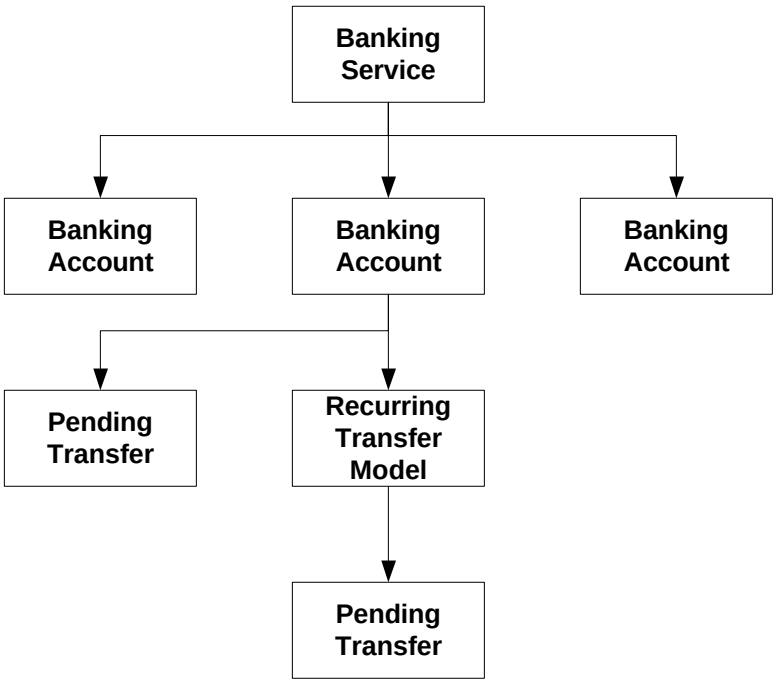
2.4.4.3.1 Cascading Deletes

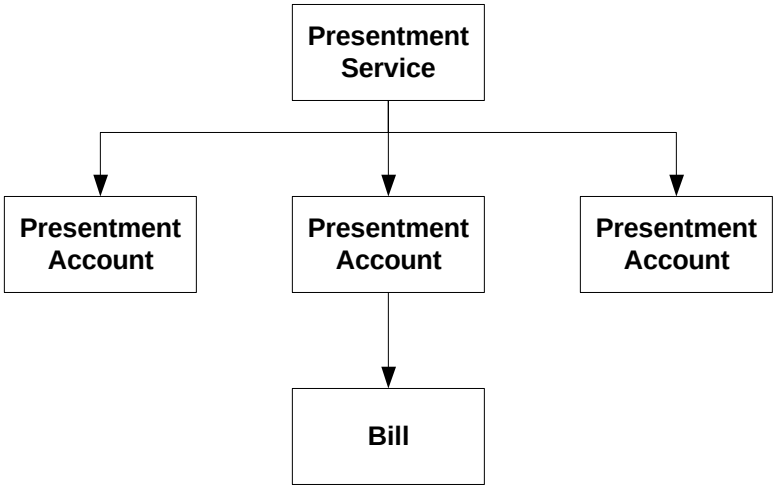
The following objects are considered related (the leftmost object cascades into the object on the right of the colon):

- Presentment service: Bill Presentment Accounts
- Payment service: Payment Service Accounts, Pending payments (that are not yet in process or processed), Recurring payment models, Customer payees
- Payment Service Accounts: Pending Payments (that are not yet in process or processed), Recurring Payment models. While Customer payees are not truly related objects to accounts, a customer payee object may contain default payment information <DfltPmtInfo> with a <DepAcctIdFrom> or <CardAcctIdFrom>. If the Account being deactivated is listed as a default payment funding account for any customer payee, the customer payee object should be updated to remove the reference to the deactivated account. Since default payment information accounts are optional, this will not present any data integrity problems.

- Banking Service: Banking Service Accounts, Pending Transfers (that are not yet in process or processed), Recurring transfer models
- Banking Service Accounts: Pending Transfers (that are not yet in process or processed), Recurring transfer models
- Bill Presentment Accounts: Bills (Conditional)
  - New, Delivered or Viewed bills must be deleted.
  - Retired, Withdrawn or Undeliverable bill summaries should remain at the CSP/BSP. The CSP/BSP is responsible for allowing continued customer access to those bill summaries.
- Customer Payees: Pending payments (that are not yet in process or processed), Recurring payment models
- Pending Payments: none
- Recurring Payment Models: Pending payments (that are not yet in process or processed)
- Pending Transfers: none
- Recurring Transfer Models: Pending transfers (that are not yet in process or processed)







Each object defined above is represented in the hierarchy. Whenever an object is disabled/deactivated/deleted, all the objects below that are also deleted if <CascadeDel> = TRUE.

If related objects are deleted as part of a cascade delete, <*xxx*DelRs> and <*xxx*CanRs> messages must be placed in appropriate Audits and Syncs to properly communicate what was removed as part of the cascade delete. The sole exception to this is when an entire service is removed. If the service removal is successful, only the service Rs is required in the Audit or Sync.

Cascade Delete should *not* apply to completed/processed payments or retired/withdrawn/undelivered bills. It is expected that the service will denormalize the payment-payee-account data for a processed payment such that the record of the payment (and all its related data) is self-contained. It is also expected that a service will denormalize the biller-billing account-bill data for a retired/withdraw/undelivered bill such that a record of the bill is self-contained.

A <BillStatusModRq> should be issued when a payment related to a bill is deleted as a result of a cascade from a higher-level object delete. The <BillStatusModRq> must include a <BillPmtStatus> aggregate with the <BillPmtStatusCode> supplied with a value of Cancelled. The specification provides for supplying a bill identifier when adding/modifying a payment to establish this relationship.

If a server is unable to remove the entire dependent object tree during a cascade delete, then all of them must remain and the request must respond with a code of ***.

2.4.4.4 Inquiry Message

The inquiry IFX message has a name structure of <*xxx*InqRq>/<*xxx*InqRs>. It is used to search for and/or gain information about the current state of existing objects *xxx* (such as finding one or more existing payments using <PmtInqRq>. Inquiry messages limit the response set to records matching the selection criteria used in the request. Selection criterion elements in the request are sometimes repeating elements; where more than one value is given for a particular element (i.e., a repeating element), the query ORs those values. Where selection criteria across multiple different elements are provided, the query ANDs those values. Where an element is absent, the query is not filtering on that element.

Several <Status> codes are available to help the client understand the results of an Inquiry Message:

- 1120 indicates that all of the search criteria that the server was aware of were applied, and no records that matched the search criteria were found.
- 1140 indicates that some, but not all, of the search criteria that the server was aware of were applied, and records were found that matched the search criteria used.
- 1160 indicates that none of the search criteria that the server was aware of were applied, and an unfiltered set of records were found for the implicit search criteria, e.g. <CustId>.

The phrase “that the server was aware of” is used to recognize the situation that a client is implemented using a more recent level of the IFX spec than the server has implemented. If the later version added optional search criteria, the server won’t be aware of them. The client may test the <Version> value within the <SvcCore> aggregate to determine if the server is implemented to an older version of IFX. The client may use this knowledge to better interpret the <Status> code values.

> ***Note:*** *A server is not obligated to support filtering on all selection criterion elements. If a server chooses not to support a particular element as a selection criterion, it must treat that element as if it were not present in the request. That is, the server must return the appropriate record set for the elements on which it does support filtering. As a result, clients should be prepared to receive records outside the scope of the selection criteria submitted in the request.*

2.4.4.5 Audit and Synchronization Messages

This section introduces the concepts of message auditing and data synchronization as used in the IFX Specification. As its name implies, message auditing is a server function that records all message activity (i.e., creates and stores records for audit trail purposes) affecting the state of IFX defined objects. Examples of such objects are transfers, payees, and payments. The specific audit messages provide a user (typically a customer service representative [CSR]) the capability to retrieve a time-sequenced listing of message responses that added, modified or deleted a user’s objects. Informational messages like inquiries do not affect the state of the business objects, and therefore are not recorded in the audit logs.

Data Synchronization is a server function providing clients that store their object data locally (e.g., personal finance managers [PFMs]) a means to update the state of their local data by requesting and comparing data from the server. This allows the client to learn about data resulting from actions that occurred at the server but are unknown to the client. For example, a user may have multiple PFM clients or employ other channel devices, such as telephones or web browsers, to submit requests to the server that changes the state of their business object data.

The IFX audit and synchronization messages are similar in that their responses contain a list of add, modify or delete records. As described above, the difference between audit and synchronize is in their intended use, which is manifested in their message requests as differences in the selection criteria.

The sync messages are specifically tailored for PFM-like clients (i.e., clients that keep local "state"). The only selection criterion for the synchronization requests is the <Token>, which, by standard convention, is the value that determines the starting point for the synchronization. The server assigns the <Token> value, except when the client sets the token to zero to request the first synchronization or to request a refresh. In the synchronization response, the server must return an updated <NewToken> that the client should save for use in its next synchronization request. A more detailed description of the token is provided in Section 3.1.7.

Audit messages are used (typically by CSR's) to trace the sequence of messages that users may have issued during a certain timeframe that affected the state of their objects. The audit requests contain optional selection criteria specifically defined for flexible and effective tracing of user message activity. For example, a CSR may want to resolve a payment dispute, by verifying that a change to a scheduled payment had been made prior to the payment being processed. Via the payment audit message, the CSR can request from the server all modifications made to a scheduled payment that occurred before it was processed.

As in the case of user or CSR initiated actions (i.e., adds, mods or deletes), server-initiated actions for spawning instances from recurring models should also create and log audit records (i.e., "adds") for these instances. In addition, whenever the server spawns a skipped instance, it should create and log an "add" audit record for the skipped instance, and must create and log a "mod'' audit record for the model to account for the skip count adjustment made to the model. The "mod" audit record for the model is required because the user randomly may initiate the skip function any time after the model has been created. However, normal model status changes (e.g., number of remaining instances) do not require any model "mod" audit records, since these changes are predictable from the definition of the model.

2.4.4.5.1 Audit Message

The IFX audit message has a name structure of <*xxx*AudRq>/<*xxx*AudRs>. Audit in the IFX Specification refers to those messages that allow clients to receive specific message responses that change the state of an object, or class of objects, stored on the server since some past point in time. This "past point in time" is either identified by customer-specified audit selection criteria, including a DateTime range.

Clients initiate audits to find out what messages were executed that caused the server to get to the current state. Clients initiate inquiry messages to view the current state of an object, or class of objects.

"Thin" (stateless) clients also use audit messages to allow a customer or CSR to receive a change history of an object, such as a payment, typically to assist in problem resolution.

The server should return as many state changes as are practical in response to an audit request. The specific type of object determines state changes. Most objects have a clear life cycle, from creation to deletion. Each status change must be reported using the appropriate add, modify, delete/cancel message response.

The life cycle of some objects, e.g. a payment, is more complex. Information about a payment may change after it moves from a pending state to a processed state. The state change from pending to processed, and any subsequent changes must be included in the Audit response as modification response messages.

Audit must contain all messages where the <Severity> within <Status> is Info or Warn. Audit may contain messages where the <Severity> is Error, at server discretion.

2.4.4.5.2 Synchronization Message

The IFX synchronization message has a name structure of <*xxx*SyncRq>/<*xxx*SyncRs>. Synchronization in the IFX Specification refers to those messages that allow clients to receive specific message responses that change the state of an object, or class of objects, stored on the server since some past point in time. This "past point in time" is identified by a server-assigned token. The server-assigned token is only meaningful to the server that assigned it.

"Thick" (stateful) clients send synchronization requests to find out what messages were executed that caused the server to get to the current state, so as to synchronize their local data with the server (system of record). Clients send inquiry requests to view the current state of an object, or class of objects.

The server should return state changes using the same rules as Audit. However, at server discretion, multiple messages in a Sync may be "collapsed" into a single message. The resultant collapsed message must provide

information sufficient to bring the client up to date. For example, if an object has been modified several times, then deleted, it is sufficient for the server to return only the <*xxx*DelRs> response in a Sync.

The Sync response includes a playback of messages for a single customer. In it, the messages that affect a specific object must be returned in the order that that server processed then, i.e. ascending by <EffDt>. Examples of objects that the Sync refers to are a payment, in the case of <PmtSyncRq>. A client that maintains local data, sometimes referred to as "thick" or "stateful" client, should apply the changes in the order received to synchronize its local data with that maintained by the server (system of record).

The server may, at server discretion, not maintain sufficient history to reliably update the client to be consistent with the current data at the server. The server must detect this condition by checking the <Token> supplied by the client in the Sync request versus the oldest <Token> maintained by the server. When the <Token> supplied by the client is older than the history maintained by the server, the server must return <Status> with <Severity>Error. The server must not return any message records within the response. The typical client error recovery should be to issue an inquiry message to receive the current status of the object class, e.g. payments, intrabank transfers, etc.; and use it to synchronize its local database.

2.4.5 Service

A *Service* is a collection of related messages.

Service wrappers are aggregates containing one or more IFX messages of the same service. Except for Signon and Signoff, all IFX messages have a service associated with them, and must be contained in a service wrapper. <*xxx*SvcRq> is used in requests, where *xxx* is the name of the service (e.g., Pay, Bank, Base). <*xxx*SvcRs> is used in corresponding responses.

Within a service wrapper is an optional <SPName> element. The client must provide this value in cases where the destination service provider is ambiguous. One example of such a scenario is where there are multiple service providers for a given service, supporting the same messages.

The service response wrapper also contains an optional <Status>. If the associated service provider returns an error to the CSP, or the CSP is unable to forward the messages to the associated service provider, the <Status> in the service response wrapper is used to communicate this failure to the client.

A service wrapper for a particular service may repeat. One possible use for this is a case where one SP supports different messages than another. In such a situation, the client may use one SP for some messages, and another SP for others. This also makes it possible for a single SP to route internally using the service wrapper, simply by giving each destination a different <SPName> in the Service Profile.

2.4.5.1 Request <*xxx*SvcRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<SPName>	Identifier	Optional	Service provider name. May be provided by the client to clearly identify which service provider is the intended destination for this collection of messages. It is required in instances where the destination would otherwise be ambiguous.
<*xxx*Rq>	Aggregate	Optional Repeating	A collection of various requests associated with the service named in the service wrapper.

2.4.5.2 Response <*xxx*SvcRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<SPName>	Identifier	Optional Echoed	Service provider name. May be provided by the client to clearly identify which service provider is the intended destination for this collection of messages. It is required in instances where the destination would otherwise be ambiguous.
<*xxx*Rs>	Aggregate	Optional Repeating	A collection of various responses associated with the service named in the service wrapper.

2.5 Naming Convention

Naming in the IFX Specification follows a convention based on a small set of rules and a set of standard abbreviations. These rules and abbreviations apply to names of elements, aggregates, messages, messages, and services.

2.5.1 Rules for Naming in the IFX Specification

- Names should be designed for clarity of semantic meaning.
- Names are case-sensitive.
- Within an aggregate, the sequence of fields matters.
- Names must follow standard abbreviations when possible.
- The letters “Rq” and “Rs” must only appear as the final two letters of a name if it is a Request message, Response message, Request Service Wrapper, or Response Service Wrapper.
- Message names must follow Class, Object, Property, Method, Direction order; e.g., <PmtAddRq>.
- All custom values for Open Enum elements must be based on the established naming conventions and must use a name to designate the organization that developed the extension.
- All custom tags must have a prefix and a suffix. The delimiter between the prefix and the suffix may be specific to the implementation. For example all extensions developed by ABC Corporation must use the syntax <com.abccorp:Something> when “:” is specified as the delimiter. The governance process will support the registration of prefixes.

2.6 International Support

2.6.1 Country Codes

All representations of countries in the IFX Specification are the three-character codes for countries that are defined in ISO-3166. These country codes are used in this specification for the following purposes:

- In postal addresses,
- With language codes to specify a dialect of a language used in multiple countries, and
- To identify country-specific tags that may be used to pass and store data relevant only in a single country.

2.6.2 Character Sets

The IFX Specification defines two data types to represent character data.

The most general data type used for the representation of most character data is called *Character*. The IFX Specification places no restrictions on the character sets used to represent elements based on the *Character* data type. The *Character* data type is intended to allow data in either single or multi-byte character sets to be passed between client and server and stored on the server to facilitate implementation in the broadest possible number

of countries. It is expected, but not required, that multi-byte implementations be based on the UTF representation of Unicode.

The other data type used for representation of character data is called *Narrow Character*. This data type is used in the IFX Specification to restrict some elements to single-byte characters so that implementation may be simplified. These elements tend to be related to system functions and should not interfere with the ability of an implementation to support multiple-byte character sets for most character data.

2.6.3 Reporting Foreign Exchange Transactions

Messages that allow a customer to execute or schedule a financial transaction allow the customer to optionally specify the currency for the message. The SP may refuse to honor the message in the requested currency.

The client may use the Deposit Account Transaction Inquiry or similar messages for other services to retrieve financial transaction detail. The detail available for a message that was executed with an amount in a foreign currency must include the original financial transaction amount in the requested currency with the actual exchange rate applied against the customer's account for the financial transaction and the rate indicator. The converted amount in the account currency is also explicitly provided with the financial transaction detail.

The <EU.Cur> element in the Signon message supports the new laws governing the use of the euro in Europe. This allows the client to specify whether they want to see financial amounts in euros or the currency of the local country.

2.6.4 Languages and Dialects

Whenever languages are specified in the IFX Specification, they use the ISO-639 international standard for three-letter language codes. Languages also include an optional ISO-3166 three-letter country code to account for differences in dialect when the same language is used in multiple countries.

When a customer signs on, using <SignonRq>, the customer may specify a preferred language and dialect from the list provided by the server in <BaseSvcProfInfo>.

2.6.5 Times and Time Zones

Whenever times are specified in the IFX Specification, they are specified with an offset from Coordinated Universal Time (UTC). This specificity allows clients and servers in different time zones to communicate without ambiguity.

2.6.6 Country-Specific Tags

To accommodate differences in financial regulations and practices in different countries, the IFX Specification allows extensions to be created that allow additional information to be passed between clients and servers that implement messages under an individual country's regulations. Elements, Aggregates, or Messages that are added to the IFX Specification to support the needs of a single country should use custom tag names with a prefix that is the ISO-3166 three-letter country code. All three-letter tag name prefixes in the IFX Specification are reserved for country-specific tag names.

In addition, the two-character prefix "EU" has been reserved for use by the European Union.

2.7 Versioning and Specification Evolution

The IFX Specification has been designed to support two separate but related mechanisms for evolution. The specification evolves formally through an open governance process that provides a mechanism for features that are widely useful to be incorporated into the core specification, which is defined in this document, and the associated IFX implementation specifications.

In addition, individual Financial Institutions or Service Providers may design their own custom extensions to the current version of the specification to quickly add needed functionality. Through this process, new custom elements or custom aggregates may be added to existing messages, custom messages and custom profile options may be added to existing services, and entirely new custom services may be developed.

2.7.1 Extension and Customization

Since all tag names in the IFX Specification follow the same naming conventions, it is relatively easy for a Financial Institution or Service Provider to design customized extensions.

All custom tag names and custom values for Open Enum elements must be based on the established naming conventions and must use a name to designate the organization that developed the extension. All custom names and values have a prefix and a suffix. The delimiter between the prefix and the suffix may be specific to the implementation. For example, all extensions developed by ABC Corporation must use the syntax com.abccorp:Something, where ":" is specified as the delimiter.

To avoid clashes, all organizations using custom tags must either register their prefix with IFX Forum or follow the convention described below to name their prefixes:

- Use the organization's fully qualified Internet domain name, reverse the order, with the top level domain first, e.g. org.ifxforum for IFX Forum (with a domain name of ifxforum.org)
- If the organization wants to have multiple prefixes for different services or different versions of customization, it may attach a sub-domain name, e.g. org.ifxforum.banking or org.ifxforum.v2
- The prefix must consist only of lower-case alphanumeric characters or dash ('-')

Note: *the maximum length of a custom value is 80 characters.*

The governance process will support the registration of prefixes. All ISO country-specific values are reserved and assigned country specific prefixes. Therefore, it is required to reserve three-letter prefixes for such use. Unreserved values for private tags (i.e. ones that may be registered with the IFX Forum) are four or more positions in length.

2.7.2 Mandatory and Optional Elements

The IFX Specification documentation conveys a number of rules for usage of elements and aggregates. This section provides a general taxonomy of rules for mandatory and optional elements and aggregates. For documentation conventions, see Section 2.2. These rules may be categorized as follows:

- ***Required messages within a service***—Each chapter that defines messages contains a table (normally section *x*.2, where *x* is the chapter number) that includes a column labeled "Req." "Yes" in this column indicates that a server supporting the service that the chapter is part of must support this message if it supports the service. The server indicates support for optional messages (i.e., those with blank in the "Req." column) using <MsgSupt> in the section of the Service Profile that corresponds to that service. (The Service Profile section for each service is typically documented at the end of each chapter.)
- ***Message Level Required Element or Aggregate***—Within each message, the elements and aggregates that appear at the highest level (as opposed to within nested aggregates) and that are required are marked. The meaning of required is slightly different for request and response messages.
  - In a request message, an element or aggregate that is marked as required must be included.
  - In a response message, an element or aggregate that is marked as required must be included in every successful response, but need not be included in failed responses. In addition, whether the message succeeds or fails, every response must contain a Request Unique ID <RqUID>, a Response Status aggregate <Status>, and a Customer Identification aggregate <CustId> if it was provided in the request.
- ***Elements and Aggregates Within a Higher Level Aggregate***—Within an aggregate, elements and aggregates are marked in the same way that they are at the message level. When an element or aggregate is marked required within the context of a higher-level aggregate that is optional, there is an implied "required if the aggregate is included."
- ***Or and Exclusive Or Conditions***—There are places in the specification where one or more elements or aggregates from a list must be included. Where exactly one item must be included it is marked as an "Exclusive Or" or XOR condition. Where one or more items must be included it is marked as an OR condition.

- ***Required by Service Profile***—Some but not all Financial Institutions or Service Providers may require many elements and aggregates. Each FI or SP must indicate to the client through the Service Profile for the service whether or not these elements and aggregates are required.
- ***Required by Biller Directory***—Some but not all Billers may require some elements and aggregates. Each Biller must indicate to the client through their entry in the Biller Directory whether or not these elements and aggregates are required. Note that the Biller Service Provider (BSP) may actually create/maintain the directory entry for each Biller that it supports.
- ***Complex Usage Rules***—Some elements and aggregates have usage rules that are more complex than the conditions described above. Where this is the case, these conditions are described in text in the Description column.

3 Common Elements and Aggregates

3.1 Common Elements

A number of elements are used throughout the IFX Specification.

3.1.1 Edit Masks <AcctMask>, <SecretMask>

Edit masks are elements that provide a way for a user interface to determine and check what type or actual value must be in each character position of a particular related element. For example, edit mask is used in <AcctMask> in the <BillerAcctIdInfo> Aggregate to provide a way to determine whether the number entered by the user for <BillingAcct> in <PresAcctId> is a valid billing account number for that biller. Edit mask is used by <SecretMask> in <SecretPrompt> to check the characters input by the user for <Secret>.

Multiple edit masks may exist for the customer accounts at each biller, as identified by a unique <BillerNum>. In this case, the client may apply one rule at a time until one succeeds. If all edit masks fail for the account number given, the client should consider the account number as invalid. Clients are not required to support this function; instead, they may rely on the presentment service provider to do it for them. Clients may wish to support this checking for performance reasons.

Note that <AcctMask> is only 32 characters long, while the <SecretMask> is 80 characters long. The length of the edit mask element matches the length of the element being masked.

Tag	*Type*	*Usage*	*Description*
<*xxx*Mask>	NC-		Edit Mask. String describing the valid values to be checked against values input by the user.

The table below identifies the characters that may be in the mask and what they mean.

Mask Position Contains	*Position in Account Number Must Contain*	*Explanation*
a	Alpha characters A through Z	A lowercase “a” means an alpha character A through Z must appear in that position.
b	Space (blank)	A space must appear in that position.
c	Alphanumeric characters, no space	An alpha character, A through Z or a through z, or a number, 0 through 9, must appear in that position.
i	Ignore	Ignore this position. Do not check for any character’s presence or absence.
n	Digits 0 through 9	A numeric character from 0 through 9 must appear in that position
x	Alphanumeric characters, space allowed	An alpha character, A through Z or a through z, a number, 0 through 9, or a space must appear in that position.
A through Z	That exact character	An upper case letter means that that alpha character must appear in that position.
0 through 9	That exact number	A number means that that specific number must appear in that position.
Special Characters—/ * $ # , @	The specified character	A special character means that that exact special character must appear in that position in the account number.

3.1.2 Customer Login ID <CustLoginId> and Customer Permanent ID <CustPermId>

The IFX Specification uses two different identifiers for customers. The first, Customer Login ID <CustLoginId>, is a user-friendly name or number (such as tax id) that is used with a password for customer

authentication. The second identifier, Customer Permanent ID <CustPermId>, is a unique permanent id used by the SP as a database key and is not typically known by the customer. The <CustPermId> is used by the CSP to identify the Customer in messages to other service providers. When the <CustPermId> is used outside of the entity that assigned it, further qualification may be required. See Section 3.2.1.1 for more information.

Clients do not typically need to use <CustPermId>. When <CustPermId> is used, it is returned as part of the Signon Response and added to each subsequent message by an intermediate server prior to the message being delivered to the SP. Both IDs may appear together in the <CustId> aggregate (See Section 3.2.1.1).

An optional request, <CustModRq>, is defined to allow a customer to modify the <CustLoginId>. Since all back end systems at the SP should use <CustPermId> as the database key for this customer, changing the Customer Login ID should be just a matter of deleting the old <CustLoginId> and mapping the new <CustLoginId> to the existing <CustPermId>.

Tag	*Type*	*Usage*	*Description*
<CustLoginId>	NC-32		Customer Login ID. Used as a user-friendly name for the customer for authentication purposes. Some implementations may allow a user to change his or her Login ID.

Tag	*Type*	*Usage*	*Description*
<CustPermId>	NC-32		Customer Permanent ID. Used as a database key to uniquely identify an FI or CSP customer. Cannot be changed by the customer.

3.1.3 Request Identifier <RqUID>

A client uses <RqUID> to uniquely identify a request message. It is a universally unique ID (UUID) that is generated according to an algorithm specified by the Open Software Foundation Distributed Computing Environment (OSF/DCE) standards to produce a 36-character hexadecimal encoding of a 128-bit number.

A Request Identifier <RqUID> is included in request messages. A server must echo the received <RqUID> in the response message it generates.

The server must store the <RqUID> for as long as the server stores the audit/sync log. The client may then correlate the asynchronous response with the request when the response is not immediate.

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID		Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.

3.1.4 Customer Service Provider Reference Identifier <CSPRefId> and Service Provider Reference Identifier <SPRefId>

Customer Service Provider Reference Identifier <CSPRefId> and Service Provider Reference Identifier <SPRefId> are typically optional elements that allow both the CSP and SP to return message reference numbers for a message. These identifiers provide permanent reference for the message and may be used by the customer in reporting problems to Customer Service.

The reference identifiers may be different for different methods applied to a given object. For example, the reference identifier in the <PmtAddRs> may be different from the reference identifier in a subsequent <PmtModRs> applied to the same payment. Therefore, it is not safe to assume that the reference identifiers may be used to uniquely identify a particular instance of an object (such as a payment).

Tag	*Type*	*Usage*	*Description*
<CSPRefId>	Identifier		Customer Service Provider Reference Identifier.

Tag	Type	Usage	Description
<SPRefId>	Identifier		Service Provider Reference Identifier.

3.1.5 Service Provider Name <SPName>

The Service Provider Name element is a globally unique identifier for a service provider, e.g., the domain name of the SP. <SPName> is used for 2 distinct purposes.

- To specify the destination of a message, when multiple service providers provide the same service, e.g. a CSP interacts with multiple BSPs to implement Bill Presentment.
- To qualify the id of an object that it owns, e.g. <BillerId>, <PmtId>, <XferId>. There are cases where different <SPName>s may be used in the same message, e.g. the BSPs <SPName> is used to qualify the <BillerId> in a message that is sent to the CPP.

To ensure uniqueness, <SPName> must be an Internet domain name registered to the SP. For example, the Banker's Roundtable would use "org.bankersround" as the value for <SPName>.

Tag	Type	Usage	Description
<SPName>	Identifier		Service Provider Name. To ensure uniqueness, a <SPName> should be an Internet domain name.

3.1.6 Organization <Org>

The Organization <Org> element is used where information must be qualified by the organization that assigned it. For example, a tax authority assigns the tax type in the <TaxInqRq>, so <Org> is used to identify the state or country tax authority. In general, <Org> is used to make identifiers globally unique in cases where there is no existing real-world globally unique identifier. An organization may register a globally unique value of <Org> through the IFX Governance Process to avoid conflicts with other organizations with similar names. The value of <Org>, like <SPName>, may also be an Internet domain name registered to the Organization defining that name space.

Tag	Type	Usage	Description
<Org>	Identifier		Organization. Organization defining this name space.

3.1.7 Token <Token>

The IFX Specification allows clients to perform synchronization of data with servers using a server-assigned <Token>. This synchronization is performed using synchronization messages, where the client inserts a <Token> it has previously received from the server as an indication of the point in history from which the synchronization should occur. The server must return all messages that are relevant to the type of object being synchronized and have occurred since the <Token> was originally sent to the client. A client may also include <Token>0 in a Sync request, which results in the server returning all known messages of the relevant type.

The "assignor" determines the scope of uniqueness. The scope may either be globally across all customers known to the assignor or specific to a customer.

The server must echo the <Token> in the response and also generate and return a new token <NewToken>, which may be used by the client in the future to specify the current point in history. See Section 2.4.4.5.2 for information regarding synchronization message definition.

Tag	Type	Usage	Description

Tag	Type	Usage	Description
<Token>	Identifier		Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first-time requests.

3.2 Common Aggregates

3.2.1 Customer Data

3.2.1.1 Customer Identification Aggregate <CustId>

The <CustId> aggregate is used to uniquely identify the customer who submits a request. Its use is optional when the customer ID may be inferred from the session (i.e., when it matches the <CustId> in the <SignonRq>).

When the <CustId> is used in messages between a Customer and CSP, the <CustPermId> element is an optional element and the <CustLoginId> is a required element. This is because the CSP server assigns the <CustPermId>, and the client may not know this ID the first time it logs in. When <CustId> is used in messages between CSP and BSP or other service provider, <CustPermId> may be required with <CustLoginId> optional. The <SPName> indicates the CSP that issued the customer permanent id.

Because the <CustPermId> is used as a key field with many SPs, the <CustPermId> must not be changed once assigned. Future versions of IFX may permit this identifier to be changed by providing messages to communicate these changes to other SPs.

Tag	Type	Usage	Description
<SPName>	Identifier	Required	Service Provider Name. The SP that assigned the <CustPermId>.
<CustPermId>	Identifier	Required OR	Customer Permanent ID. Used as a database key to uniquely identify an FI or CSP customer. Cannot be changed by the customer. When <CustId> is used in messages between CSP and BSP or other service provider, <CustPermId> may be required with <CustLoginId> optional.
<CustLoginId>	Identifier	Required OR	Customer Login ID. Used as a user-friendly name for the customer for authentication purposes. Maps directly to Customer Permanent ID. Some implementations may allow a user to change his or her Login ID. When the <CustId> is used in messages between a Customer and CSP, the <CustPermId> element is an optional element and the <CustLoginId> is a required element.

3.2.1.2 Customer Name Aggregate <CustName>

The <CustName> aggregate is used to specify a customer's name and, optionally, nickname.

Tag	Type	Usage	Description
<LastName>	C-40	Required Repeating	Customer Last Name.
<FirstName>	C-40	Required	Customer First Name.

Tag	Type	Usage	Description
<MiddleName>	C-40	Optional Repeating	Customer Middle Name.
<TitlePrefix>	C-8	Optional	Customer Title Prefix. For example, “Ms.”, or “Dr.”
<NameSuffix>	C-40	Optional	Customer Name Suffix. For example, “MD” or “Jr.”
<Nickname>	C-40	Optional	Customer Nickname. Assigned by customer if desired.
<LegalName>	C-96	Optional	Used by entities like trusts, or businesses in the case where an individual is “doing business as” a company. Organizations and companies will be supported more fully in future versions of IFX.

3.2.1.3 Customer Contact Aggregate <CustContact>

The <CustContact> aggregate appears wherever the customer’s contact information is needed. The customer provides this information to the SP regarding how and when to contact him or her. This aggregate is most commonly used to allow the customer to override the default customer contact information stored as part of the Customer Profile.

Tag	Type	Usage	Description
<CustContactPref>	Open Enum	Optional	Customer Contact Preference. This is a customer-provided preference for contact by FI and SP staff. Valid values: DayPhone, EvePhone, DayFax, EveFax, Email, Post.
<PrefTimeStart>	Time	Optional	Preferred Customer Contact Start Time. This is a customer-provided start time preference for contact by SP staff.
<PrefTimeEnd>	Time	Optional	Preferred Customer Contact End Time. This is a customer-provided end time preference for contact by SP staff.
<DayPhone>	Phone Number	Optional	Customer Daytime Telephone Number.
<EvePhone>	Phone Number	Optional	Customer Evening Telephone Number.
<DayFax>	Phone Number	Optional	Customer Daytime Fax Number.
<EveFax>	Phone Number	Optional	Customer Evening Fax Number.
<EmailAddr>	NC-128	Optional	Customer Email Address.

3.2.1.4 Customer Name/Address Aggregate <CustNameAddr>

Tag	Type	Usage	Description
<NameAddrType>	Open Enum	Required	Name/Address type. Defined values: Billing, Delivery, Customer
<FullName>	C-96	Required XOR	Concatenated customer name associated with the address.
<CustName>	Aggregate	Required XOR	Customer name. The customer name associated with the address.
<PostAddr>	Aggregate	Required	Postal address.
<CustContact>	Aggregate	Optional	Customer contact aggregate.

3.2.2 Organizational Contact Aggregate <OrgContact>

The <OrgContact> aggregate appears wherever contact information for an organization is needed. Contact type indicates the function of the contact.

Tag	*Type*	*Usage*	*Description*
<ContactType>	Open Enum	Required	Contact Type. Defined values: CustSvc, Enroll, Tech, Personal
<Desc>	C-80	Optional	Description. Text describing the purpose of the group providing contact information is provided.
<Phone>	Phone Number	Optional	Telephone Number.
<Fax>	Phone Number	Optional	Fax Number.
<EmailAddr>	NC-128	Optional	Email Address.
<URL>	URL	Optional	Web Site Address.

3.2.3 Biller Data

3.2.3.1 Biller Contact Aggregate <BillerContact>

The <BillerContact> aggregate is used to provide complete information about a Biller.

Tag	*Type*	*Usage*	*Description*
<LegalName>	C-96	Optional	Biller Legal Name. Assigned by the service provider. Cannot be changed by the client.
<Name>	C-40	Optional	Biller Business Name. Assigned by the service provider. Cannot be changed by the client.
<PostAddr>	Aggregate	Optional	Biller Address Aggregate. Assigned by the service provider. Cannot be changed by the customer.
<BillRetAddr>	Aggregate	Optional	Bill Return Address Aggregate. Assigned by the service provider. Cannot be changed by the customer.
<RemitName>	C-40	Optional	Remittance Name. Assigned by the service provider. Cannot be changed by the customer.
<RemitAddr>	Aggregate	Optional	Remittance Address Aggregate. Assigned by the service provider. Cannot be changed by the customer.
<OrgContact>	Aggregate	Optional Repeating	Support Contact Detail Aggregate.

3.2.3.2 Biller Pay Information Aggregate <BillerPayInfo>

The <BillerPayInfo> identifies payment information associated with a Biller.

Tag	*Type*	*Usage*	*Description*

Tag	Type	Usage	Description
<PmtInst>	Aggregate	Optional Repeating Profiled values	Payment Instrument Aggregate. Types of payment that the biller accepts via the Pay provider. This is used to restrict the methods by which the customer may pay for the bill. The Service Provider indicates which Payment Instruments are supported via profile. The Biller may choose to override this list by including <PmtInst> in the Bill Summary <BillRec>.
<DaysToEPost>	Long	Optional	Days to post an electronic payment. The maximum days between receipt of an electronic payment by the biller or BPP and posting to the customer's account.
<PrenoteReqd>	Boolean	Optional	Pre-Note Required. If set to TRUE, the Biller requires pre-noting by the Service Provider. NOTE: Pre-noting is the transmission of a zero dollar message to verify consumer information, usually the payee account information.

3.2.4 Postal Addresses

3.2.4.1 Postal Address Aggregate <PostAddr>

The <PostAddr> aggregate is used wherever a postal address is needed.

Tag	Type	Usage	Description
<Addr1>	C-64	Required	Address Line 1.
<Addr2>	C-64	Optional *but see Description*	Address Line 2. Required if <Addr3> is present.
<Addr3>	C-64	Optional *but see Description*	Address Line 3. Required if <Addr4> is present.
<Addr4>	C-64	Optional	Address Line 4.
<City>	C-32	Optional	City.
<StateProv>	C-32	Optional	State or Province.
<PostalCode>	C-11	Optional	Postal Code.
<Country>	NC-3	Optional	Country. Values are defined by ISO-3166 3-letter codes.

3.2.5 Account Data

3.2.5.1 Bank Account Record Aggregate <BankAcctRec>

The <BankAcctRec> aggregate provides information about a customer banking account.

Tag	Type	Usage	Description
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<BankAcctInfo>	Aggregate	Required	Bank Accont Information Aggregate
<BankAcctStatus>	Aggregate	Optional	Bank Account Status Aggregate

Tag	Type	Usage	Description
<AcctTaxType>	Open Enum	Optional	Account Tax Type. Defined values: TaxDeferred, Standard.
<XferFromSupt>	Aggregate	Optional	Transfer From Supported Aggregate. If present, indicates that this account is a valid source account for a transfer.
<MinCurAmt>	Currency Amount	Optional	Minimum amount that may be transferred from the account, if applicable.
<MaxCurAmt>	Currency Amount	Optional	Maximum amount that may be transferred from the account, if applicable.
</XferFromSupt>			
<XferToSupt>	Aggregate	Optional	Transfer To Supported Aggregate. If present, indicates that this account is a valid target account for a transfer.
<MinCurAmt>	Currency Amount	Optional	Minimum amount that may be transferred to the account, if applicable.
<MaxCurAmt>	Currency Amount	Optional	Maximum amount that may be transferred to the account, if applicable.
</XferToSupt>			
<PaySupt>	Aggregate	Optional	Payment Supported Aggregate. If present, indicates that this account is a valid source account for payment.
<MinCurAmt>	Currency Amount	Optional	Minimum amount that may be paid from the account, if applicable.
<MaxCurAmt>	Currency Amount	Optional	Maximum amount that may be paid from the account, if applicable.
</PaySupt>			
<AcctBal>	Aggregate	Optional Repeating	Account Balance Aggregate. Repeated for each balance to be included for this account.

3.2.5.1.1 Deposit Account Identifier Aggregates <DepAcctId>, <DepAcctIdFrom>, <DepAcctIdTo>

The <DepAcctId> aggregate is used to uniquely identify a deposit-type account. When a single account is specified in a message or aggregate, it is referred to as <DepAcctId>. When multiple accounts appear in a message or aggregate, the <DepAcctIdFrom> and <DepAcctIdTo> aggregates are used for clarity. The <DepAcctIdFrom> and <DepAcctIdTo> aggregates have the same structure as the <DepAcctId> aggregate.

Tag	Type	Usage	Description
<BankId>	NC-34	Optional	Bank Identifier. Qualifies account number if known by the customer/client. Usage is expected to be routing and transit number in the US or the equivalent in an international implementation.
<BranchId>	NC-22	Optional	Branch Identifier. Qualifies account number if known by the customer/client. Used to indicate which branch the account is with. Usage is expected to be primarily in countries where regulations require an account to be assigned to a branch office, but a US-based FI may also require this field for operational reasons.
<AcctId>	NC- 32	Required	Account Identifier. Sometimes known as account number, but not restricted to numeric characters.

Tag	Type	Usage	Description
<AcctType>	Open Enum	Required	Account Type. See Data Dictionary for details. Defined values: DDA, SDA, CDA, MMA, CMA. NOTE: An Open Enum data type permits the addition of account types specific to an FI, country, etc.
<AcctKey>	NC-22	Optional	Account Key. Checksum for international banks.
<BankInfo>	Aggregate	Optional *but see Description*	Bank Information aggregate. This aggregate contains additional information used to identify a bank, typically when used for payment transfers, payee requests, and wire transfers. It is required in CHE, DEU, and ITA.
<Name>	C-40	Optional *but see Description*	Bank name, required in CHE, DEU, and ITA.
<BranchName>	C-40	Optional *but see Description*	Bank branch name, required in ITA.
<PostAddr>	Aggregate	Optional XOR	Bank Postal Address. Either <PostAddr> or broken-out <City>, <StateProv>, and <PostalCode> may be used, but not both.
<City>	C-32	Optional *but see Description* XOR	Bank Branch City. Required in CHE. Either <PostAddr> or broken-out <City>, <StateProv>, and <PostalCode> may be used, but not both.
<StateProv>	C-32	Optional *but see Description*	Bank Branch State or Province. Either <PostAddr> or broken-out <City>, <StateProv>, and <PostalCode> may be used, but not both.
<PostalCode>	C-11	Optional *but see Description*	Bank Branch Postal Code. Required in CHE. Either <PostAddr> or broken-out <City>, <StateProv>, and <PostalCode> may be used, but not both.
</BankInfo>			

3.2.5.1.2 Card Account Identifier Aggregates <CardAcctId>, <CardAcctIdFrom>, <CardAcctIdTo>

The <CardAcctId> aggregate is used to uniquely identify a card-type account, such as credit card or debit card. When a single account is specified in a message or aggregate, it is referred to as <CardAcctId>. When multiple accounts appear in a message or aggregate, the <CardAcctIdFrom> and <CardAcctIdTo> aggregates are used for clarity. The <CardAcctIdFrom> and <CardAcctIdTo> aggregates have the same structure as the <CardAcctId> aggregate.

Tag	Type	Usage	Description
<AcctId>	NC-32	Required XOR	Account Identifier. Sometimes known as account number, but not restricted to numeric characters.
<CardMagData>	Aggregate	Required XOR	Card Magnetic Stripe Data
<AcctType>	Open Enum	Required	Account Type. See Data Dictionary for details. Defined values: CCA, DDA, SDA, Default, Unknown
<CCMotoAcct>	Aggregate	Optional	Credit Mail Order/Telephone Order Account Aggregate. Used for defining additional Credit Card data when required for "card not present" transactions, such as when activating a Credit Card as a funding account for payment.

Tag	Type	Usage	Description
<ExpDt>	DateTime	Required	Expiration date for card. If the card has only month and year expiration, the *last* day of the month must be specified here.
<Name>	C-40	Optional	Customer Name. Name on the front of this card (exactly as name appears on the card).
<PostAddr>	Aggregate	Optional XOR	Customer Address Aggregate. Customer address associated with this card.
<PostalCode>	C-11	Optional XOR	Customer Postal Code. The postal code associated with this card.
<Phone>	Phone Number	Optional	Customer phone number associated with this card.
<Brand>	Open Enum	Optional	Type of credit card (VISA, AMEX, etc.). This is just a check against the information provided by the account number.
</CCMotoAcct>			

3.2.5.1.2.1 Card Magnetic Stripe Data <CardMagData>

Tag	Type	Usage	Description
<MagData1>	NC-180	Required OR	Track 1 data
<MagData2>	NC-180	Required OR	Track 2 data
<MagData3>	NC-180	Required OR	Track 3 data

3.2.5.1.3 Loan Account Identifier Aggregates <LoanAcctId>, <LoanAcctIdFrom>, <LoanAcctIdTo>

The <LoanAcctId> aggregate is used to uniquely identify a loan-type account. When a single account is specified in a message or aggregate, it is referred to as <LoanAcctId>. When multiple accounts appear in a message or aggregate, the <LoanAcctIdFrom> and <LoanAcctIdTo> aggregates are used for clarity. The <LoanAcctIdFrom> and <LoanAcctIdTo> aggregates have the same structure as the <LoanAcctId> aggregate.

Tag	Type	Usage	Description
<AcctId>	NC-32	Required	Account Identifier. Sometimes known as account number, but not restricted to numeric characters.
<AcctType>	Open Enum	Required	Account Type. See Data Dictionary for details. Defined values: MLA, ILA, LOC, EQU, CLA. NOTE: An Open Enum data type permits the addition of account types specific to an FI, country, etc.

3.2.5.1.4 Bank Account Information Aggregate <BankAcctInfo>

Tag	Type	Usage	Description
<CurCode>	NC-3	Required	Currency Code. As defined by ISO-4217.

Tag	Type	Usage	Description
<Desc>	C-80	Optional	Account Description. Assigned by the Financial Institution. Typically product name. Does not include account number.
<CustName>	Aggregate	Optional	Customer Name in which this account is held.
<OrgContact>	Aggregate	Optional	FI Customer Service Contact Information.
<CustAcctUse>	Open Enum	Optional	Customer Account Use. Defined values: Retail, Business.
<PrimaryAcct>	Boolean	Optional	Primary Account Indicator. If TRUE, used as default account for a class of accounts (i.e., checking). Typically used in ATM networks. May be used for other applications by some FIs.
<Term>	Aggregate	Optional	Term Aggregate.

3.2.5.1.5 Bank Account Status <BankAcctStatus>

Tag	Type	Usage	Description
<BankAcctStatusCode>	Closed Enum	Required	Account Status. Valid values: Open—Open. Closed—Closed. Inactive—Inactive. NotAvail—Not Available.
<StatusDesc>	C-255	Optional	Status Description. Explanatory text associated with this bank account status.
<EffDt>	DateTime	Optional	Effective Date/Time. The date/time the <BankAcctStatusCode> took effect.
<StatusModBy>	Open Enum	Optional	

3.2.5.2 Presentment Account Record Aggregate <PresAcctRec>

Tag	Type	Usage	Description
<PresAcctId>	Aggregate	Required	Presentment Account Identification Aggregate
<PresAcctInfo>	Aggregate	Required	Presentment Account Information Aggregate
<PresAcctStatus>	Aggregate	Optional	Presentment Account Status Aggregate

3.2.5.2.1 Presentment Account Identification <PresAcctId>

The <PresAcctId> aggregate uniquely identifies a customer's account with a biller by the combination of BSP <Org>, biller identifier <BillerNum>, and account number <BillingAcct>. <BillerNum>s must be unique within a BSP.

Upon account activation, the <BillingName> has to be correlated to the <CustPermId> from the CSP's records by either or both the CSP and the BSP to ensure that the Customer's identification as known to the CSP is properly matched to the Customer's identification as known to the Biller. For account activation, this aggregate also provides the CSP with the ability to send the BSP the <StdPayeeId> or the <CustPayeeId> that is

associated with this biller at the appropriate CPP. If the BSP stores these data, they must be returned in any inquiry response that includes <PresAcctId> (e.g. <BillInqRs>).

Tag	Type	Usage	Description
<BillingAcct>	C-32	Required	Customer account with the Biller.
<BillerId>	Aggregate	Required	Biller Identifier Aggregate. This is an identifier created by the Biller or BSP.

3.2.5.2.2 Presentment Account Information Aggregate <PresAcctInfo>

Tag	Type	Usage	Description
<StdPayeeId>	Aggregate	Optional	Standard Payee Identifier. The identification for this biller at the CPP. When sent in account activation, it is intended for storage on the BSP database, such that it may be returned in subsequent inquiries utilizing this aggregate. This may be used by the CSP to help match the BSP's Biller ID to the Standard Payee ID as known to the CPP.

3.2.5.2.3 Presentment Account Status <PresAcctStatus>

The <PresAcctStatus> aggregate describes the status of the customer's account with a biller.

Tag	Type	Usage	Description
<PresAcctStatusCode>	Closed Enum	Required	Account Status. Valid values: Open—Open. Closed—Closed. Inactive—Inactive. NotAvail—Not Available.
<StatusDesc>	C-255	Optional	Status Description. Explanatory text associated with this bank account status.
<EffDt>	DateTime	Optional	Effective Date/Time. The date/time the <BankAcctStatusCode> took effect.
<StatusModBy>	Open Enum	Optional	

3.2.6 Account Balances

The following tables identify the account balances applicable to a specific account type. The first table identifies the association between the account type and the specific account classification <DepAcctId>, <CardAcctId> and <LoanAcctId>. The second table identifies the relationship between account classification and balance types. All account types use the balances associated with their account classification.

The <EffDt> associated with each balance indicates the time and date as of which the balance is current. The frequency of changing this date depends on how often the FI/SP refreshes the balances for a specific account type. This time and date might be updated by every update message in an on-line bank or be refreshed once a day by a strip file implementation. Even within a single FI/SP, the Demand Deposit ("Checking") Account (DDA) might be online, while Savings (SDA) or Certificates of Deposit (CDA) might be updated overnight.

3.2.6.1 Account Balance Aggregate <AcctBal>

The <AcctBal> aggregate is used to express an account balance. The balance being specified is identified in <BalType>, which is an Open Enum. When <AcctBal> is used, it may repeat, to allow multiple balances to be specified for the given account.

Tag	*Type*	*Usage*	*Description*
<BalType>	Open Enum	Required	Balance Type. Defined Values: Ledger, OpeningLedger, ClosingLedger, MinLedger, AvgLedger, Avail, Current, Outstanding, OpeningOutstanding, ClosingOutstanding, AvailCredit, CreditLimit, PayoffAmt, Principal, Escrow.
<CurAmt>	Currency Amount	Required	Balance Amount.
<EffDt>	DateTime	Optional	Effective DateTime.
<ExpDt>	DateTime	Optional	Expiration Date of the Balance. For example, a payoff amount expiration date.
<Desc>	C-80	Optional	Description.

3.2.6.1.1 Balance Type and Classification Association

	Account Classification		
Balance Type	*Deposit (Asset)*	*Loan (Liability)*	*Card (Asset or Liability)*
Ledger	×		×
OpeningLedger	×		×
ClosingLedger	×		×
MinLedger	×		×
AvgLedger	×		×
Avail	×		×
Current	×		×
Outstanding		×	×
OpeningOutstanding		×	×
ClosingOutstanding		×	×
AvailCredit		×	×
CreditLimit		×	×
PayoffAmt		×	×
Principal		×	
Escrow		×	

3.2.6.2 Extended Account Balance Aggregate <ExtAcctBal>

The <ExtAcctBal> aggregate is used to express an extended account balance. The balance being specified is identified in <ExtBalType>, which is an Open Enum. When <ExtAcctBal> is used, it may repeat, to allow multiple balances to be specified for the given account.

Tag	*Type*	*Usage*	*Description*

Tag	Type	Usage	Description
<ExtBalType>	Open Enum	Required	Balance Type. Defined Values: PrepayPenalty, Orig, UnpaidAccruedInt, Redemption, YTDWithhold, LastYrWithhold, Overdraft, Overlimit, PastDue, CashLine, OutstandingCashAdv, CashAvail, PendAuthAmt, LastStmtBal, YTDInterest, LastYrInt, YTDFees, PeriodFees, LastYrFees.
<CurAmt>	Currency Amount	Required	Balance Amount.
<EffDt>	DateTime	Required	Balance DateTime.
<ExpDt>	DateTime	Optional	Expiration Date of the Balance. For example, a payoff amount expiration date.
<Desc>	C-80	Optional	Description.

3.2.6.2.1 Extended Balance Type and Classification Association

	Account Classification		
Extended Balance Type	*Deposit (Asset)*	*Loan (Liability)*	*Card (Asset or Liability)*
PrepayPenalty		×	
Orig	×	×	×
UnpaidAccruedInt		×	
Redemption			
YTDWithhold			
LastYrWithhold			
Overdraft	×		
Overlimit			×
PastDue		×	×
CashLine		×	×
OutstandingCashAdv		×	×
CashAvail	×		×
PendAuthAmt		×	×
LastStmtBal	×	×	×
YTDInterest	×	×	×
LastYrInt	×	×	×
YTDFees	×	×	×
PeriodFees	×	×	×
LastYrFees	×	×	×

3.2.6.3 <AcctType> and Classification Association

	Account Classification		
<AcctType>	*Deposit*	*Loan*	*Card*

	(Asset)	*(Liability)*	*(Asset or Liability)*
CCA—Credit Card Account			×
CDA—Certificate of Deposit	×		
CLA—Commercial Loan Account		×	
CMA—Cash Management Account	×		
DDA—Demand Deposit Account	×		×
EQU—Home Equity Loan		×	
ILA—Installment Loan Account		×	
LOC—Consumer Line of Credit		×	
MLA—Mortgage Loan Account		×	
MMA—Money Market Account	×		
SDA—Savings Account	×		×

The sign of account balances are stated as positive values when greater than zero (outstanding balance due on a customer's liability account and positive balance in a customer's asset account) and negative numbers when less than zero (credit balance in a customer's liability account and overdrawn balance in a customer's asset account). For a Card Account, the <AcctType> is used to indicate whether the account is an asset or liability account.

3.2.7 Term Data

3.2.7.1 Deposit Term Aggregate <Term>

The <Term> aggregate contains details of the term of a deposit or a loan.

Tag	*Type*	*Usage*	*Description*
<Count>	Long	Required	Count. In this context, <Count> is the number of <TermUnits> for which a term deposit is effective. At the end of this period, the Term Deposit ceases to exist. Note: if <TermUnits>=DAYS, this is the number of calendar days, not business days.
<TermUnits>	Closed Enum	Required	Term Units. Units in which the Term Deposit is measured.
<Desc>	C-80	Optional	Short Description.
<DaysCall>	Long	Optional *but see Description*	Days Call. The number of business days' notice required for withdrawal. Required if <TermUnits>=Indefinite.

3.2.8 Selection Criteria

3.2.8.1 Selection Range Date Aggregates <SelRangeDt>, <SelRangeDueDt>, <SelRangePrcDt>

The <SelRangeDt> aggregate identifies the start date and/or the end date as a selection criterion. The Selection Range Due Date <SelRangeDueDt> and Selection Range Processing Date <SelRangePrcDt> have the same structure as the <SelRangeDt>.

The <SelRangeDt> aggregate contains two optional fields: <StartDt> and <EndDt>.

The following rules are established for the <StartDt> and <EndDt> fields:

- If <StartDt> but not <EndDt> is supplied, then the search must match dates that are later than or equal to <StartDt>.

- If <EndDt> is supplied but not <StartDt>, then the search must match dates that are earlier than or equal to the <EndDt>.
- If both dates are supplied, then the search must match dates that are between or equal to the two dates supplied.
- If a client wishes to search for a specific date, then both <StartDt> and <EndDt> should be supplied and set to the same date, with <EndDt> set for 23:59:59. If <StartDt> and <EndDt> both have no times included, the semantic intent of the request is to search for items that occurred precisely at midnight on the specified date.

Note: *<StartDt> and <EndDt> are date fields typically entered by a consumer and are not timestamp fields. <StartTS>, on the other hand, is a timestamp (date and time) field found on many <xxxAudRq> messages. This timestamp is used by a client to ask for any audit records created after the value of <StartTS>. A service provider determines this by comparing <StartTS> to the <EffDt> (also a timestamp field) that is held for each audit record. The server must return an <EffDt> back on the <xxxAudRs> message for each audit record. The client may use the latest <EffDt> as the value of <StartTS> on the <xxxAudRq> on a subsequent call.*

Tag	*Type*	*Usage*	*Description*
<StartDt>	DateTime	Required OR	Selection Start Date.
<EndDt>	DateTime	Required OR	Selection End Date.

3.2.8.2 Selection Range Currency Amount Aggregate <SelRangeCurAmt>

The <SelRangeCurAmt> aggregate contains two optional fields: <LowCurAmt> and <HighCurAmt>. The following rules are established for these fields:

- If <LowCurAmt> but not <HighCurAmt> is supplied, then the search must match amounts that are greater than or equal to <LowCurAmt>.
- If <HighCurAmt> is supplied but not <LowCurAmt>, then the search must match amounts that are smaller than or equal to the <HighCurAmt>.
- If both amounts are supplied, then the search must match amounts that are between or equal to the two amounts supplied.
- If a client wishes to search for a specific amount, then both <LowCurAmt> and <HighCurAmt> should be supplied and set to the same value.
- Both amounts must be of the same currency.

Tag	*Type*	*Usage*	*Description*
<LowCurAmt>	Currency Amount	Required OR	Selection Low Amount.
<HighCurAmt>	Currency Amount	Required OR	Selection End Amount.

3.2.9 Recurring Model Data

3.2.9.1 Recurring Model Information Aggregate <RecModelInfo>

Tag	*Type*	*Usage*	*Description*
<Freq>	Open Enum	Required	Recurring Model Frequency. See Data Dictionary for details.
<NumInsts>	Long	Optional XOR Profiled support	Recurring Model Total Instances. If none of <NumInsts>, <FinalPrcDt>, or <FinalDueDt> is provided, the model is open-ended.
<FinalPrcDt>	Date	Optional XOR Profiled support	Final Processing Date. Subject to server support for Processing Date Model in profile. Only payments allow the option of processing date. All transfers must use due date. If none of <NumInsts>, <FinalPrcDt>, or <FinalDueDt> is provided, the model is open-ended.
<FinalDueDt>	Date	Optional XOR Profiled support	Final Due Date. Subject to server support for Due Date Model in profile. If none of <NumInsts>, <FinalPrcDt>, or <FinalDueDt> is provided, the model is open-ended.
<InitialCurAmt>	Currency Amount	Optional Profiled support	Recurring Model Initial Amount. May be different from nominal instance amount. If omitted, the default instance amount is used for the initial transfer or payment.
<FinalCurAmt>	Currency Amount	Optional Profiled support	Recurring Model Final Amount. May be different from nominal instance amount. If omitted, the default instance amount is used for the final transfer or payment.
<Nickname>	C-40	Optional Profiled support	Nickname. Optionally assigned by the customer. Pay provider may indicate support for longer-term storage of nicknames in the Service Profile. May be modified by the client.
<SkipNextN>	Long	Optional Profiled support	Skip Next *N* Instances. A non-zero value indicates that the next *N* transaction instances are to be skipped. Subject to server support in Service Profile. ***Note:*** *The server must decrement the Skip Next N count and generate the instance with a processing status code of Skipped, based on the timing of the recurring model. The server must handle the skipped instance like any other instance, except that the server does not execute the instance for fulfillment purposes, including creating audit and sync records for the instance addition, and for the recurring model status changes. Therefore, each decrement must be reflected as a separate <RecXferModRs> within audits and syncs, and each status change to the specific instances of the transfers must be reflected as a separate <XferModRs>.*

3.2.10 System Message Data

3.2.10.1 Response Status Aggregate <Status>

The <Status> aggregate is used in each response to indicate the status of the message. See Appendix A for information on Response Status Codes.

When <Severity> = Info or Warn, the server generates the complete response message. When <Severity> = Error, the server is only required to include the <Status> aggregate and echo the <RqUID> and <CustId> fields. The server should include these fields unless a system error prevents the server from reading the request message contents. In the latter situation, it may only be possible for the server to generate a response message containing the <Status> aggregate by itself. Other actions to be taken in the case of severe errors are specified for each implementation.

Note: *The explanatory text provided in the <StatusDesc> is freeform text that is not strictly dictated by the status code. Appendix A provides default <StatusDesc> text, but this text may be replaced or enhanced by particular implementations.*

Tag	*Type*	*Usage*	*Description*
<StatusCode>	Long	Required	Response Status Code. Valid values depend on context. See Appendix A.
<SrvrStatusCode>	C-20	Optional	Server Status Code. The value placed here is used to allow the client to display the status code to the user. This allows the user to read the code to a customer service representative for debugging purposes.
<Severity>	Closed Enum	Required	Severity. Valid values: Error—There was an error in the request severe enough that the requested message was not processed. Warn—There was a problem with the request, but a valid response is still present. The requested message was processed. Info—Information Only. The requested message was processed.
<StatusDesc>	C-255	Optional	Status Description. Explanatory text associated with the status code. The Statue Description may be default text or Service Provider specific.

3.2.10.2 Records Control

Some response messages, typically associated with an inquiry/audit/retrieve email, may be quite large. The size of the response message may create problems for both the client and server. The problems are specific to the implementation. Some examples of these problems are:

- The client may not be able to allocate enough storage for the entire response.
- A network timeout may occur due to the time to return the complete HTTP response.
- Mixing very large with small messages within the message manager (OLTP) middleware may create serious performance problems for the small messages.

In order to address these potential problems, IFX allows either the client or server to constrain the size of a response message. The client initiates all IFX messages, so the server cannot “push” the additional information to the client. A mechanism is provided for the server to indicate that additional information is available and for the client to request the additional information

The RecCtrl value within <OptSupt> in the Service Profile indicates whether a server supports this capability.

3.2.10.2.1 Records Control Input <RecCtrlIn>

The <RecCtrlIn> aggregate is used in inquiry/audit/retrieve email request messages to allow the client to specify a maximum number of records that it is capable of processing within a single response. The initial request omits the <Cursor> element.

Subsequent requests set the <Cursor> to the value returned by the server in the previous response (Rs) message. The <Cursor> value is meaningful to the server and opaque to the client. The <Cursor> is intended to be a pointer to a position within the answer set for the inquiry/audit/select email request.

The client should send the same selection criteria in the initial and subsequent requests. The server may verify that the cursor returned is valid by comparing the search criteria on subsequent requests with the search criteria on the initial request. If they differ, the cursor may be considered invalid and an error returned.

Similarly, if the time between the initial and subsequent requests is too long, the cursor may be rejected on the subsequent request. In this situation, the client may reinitiate the initial request.

Tag	Type	Usage	Description
<MaxRec>	Long	Required	Maximum Records Requested.
<Cursor>	Binary	Optional *but see Description*	Cursor Handle. Must be omitted in initial request. May be included in subsequent requests if server provided as a pointer to more records in previous <RecCtrlOut> in response.

3.2.10.2.2 Records Control Output <RecCtrlOut>

The <RecCtrlOut> aggregate is used in inquiry/audit/retrieve email responses to provide a client that used the <RecCtrlIn> aggregate in the request with information it needs to retrieve the rest of the records that matched the specified selection criteria.

The <RecCtrlOut> aggregate must be sent as part of the Rs if the server indicated support for Records Control in the Service Profile by setting <OptSupt> = RecCtrl and one or both of the following are true:

- The Rq contains an <RecCtrlIn> aggregate or
- The server is returning a subset of the available output records in an Rs.

Tag	Type	Usage	Description
<MatchedRec>	Long	Required	Matched Records. Total number of records matching the selection criteria.
<SentRec>	Long	Required	Sent Records. Number of records matching the selection criteria that are included in this message.
<Cursor>	Binary	Optional *but see Description*	Cursor. Included in the response only if additional records are available. When <Cursor> is present in the response, the Status code *** must be sent. Assume server retains cursor of matching records for some time in case client requests more. Used to allow the client to issue another request to fetch more matching records.

3.2.10.2.3 Example: Using Records Control

Assume:

- The server sets <OptSupt>=RecCtrl in Banking section of Service Profile.
- The client issues an Account History message.
- The client is capable of processing only process 10 records (history messages) within a single Rs.
- There are 25 messages in the answer set.

Initial request from client:

```
<MaxRec>=10
```

Note that the <Cursor> element is omitted.

Initial response:

```
<MatchedRec>=25

<SentRec>=10
```

```
<Cursor>=binary_value_1
```
Second request:

```
<MaxRec>=10
```

```
<Cursor>=binary_value_1
```
Second response:

```
<MatchedRec>=25
```

```
<SentRec>=10
```

```
<Cursor>=binary_value_2
```
Third request:

```
<MaxRec>=10
```

```
<Cursor>=binary_value_2
```
Third response:

```
<MatchedRec>=25
```

```
<SentRec>=5
```
<Cursor> element is omitted

3.2.10.3 Client Application <ClientApp>

The primary use of the <ClientApp> aggregate is to inform a customer service representative about what software was used to create or change an object. Every inquiry response message should contain the <ClientApp> of the client (customer's or customer service representative's) that last modified the object, unless the object was created or changed by the server. Each message returned in an Audit response should include the <ClientApp> of the client that initiated the message. <ClientApp> conveys the application that is acting as the client endpoint of the IFX message. This may be an application on the customer's PC if it is aware of Interactive Financial Exchange, or it may be an intermediate server that is composing IFX requests on behalf of a client that has connected to it using some other protocol (e.g., HTTP).

Tag	*Type*	*Usage*	*Description*
<Org>	Identifier	Required	Organization. This identifies the client application manufacturer.
<Name>	C-40	Required	Client Application Name.
<Version>	NC-12	Required	Client Application Version.

3.2.11 Industry Identifier Aggregate <IndustId>

The <IndustId> aggregate provides standard codes by which industries are classified. <IndustId> is used in payments and other messages as information for reporting and accounting by industry code.

Tag	*Type*	*Usage*	*Description*
<Org>	Identifier	Required	Organization. Identifies the organization assigning numbers to different industries. "SIC" identifies Standard Industrial Codes. "NAICS" identifies North American Industrial Code System.
<IndustNum>	NC-6	Required	Number identifying the industry.

3.2.12 Secrets Aggregates

3.2.12.1 Secret Prompt Aggregate <SecretPrompt>

The <SecretPrompt> aggregate is used to provide enough information to prompt a user for secrets used for initial authentication. For example, the server may request the user's tax ID, mother's maiden name or some other secret already shared between the user and the service provider operating the server.

The <SecretPrompt> may be used by CSPs during enrollment of customers for a specific service such as Bill Presentment. For example, a customer new to a specific biller may be authenticated against existing customer records at the biller. If omitted, the biller does not require the customer to enter any secrets for client enrollment.

Another use is to prompt for a security token or password that has been provided to the customer by some other means, possibly during an interactive identity verification session with the customer.

Tag	*Type*	*Usage*	*Description*
<SecretId>	Identifier	Required	The ID of this secret. The client must return the ID along with the secret itself in <SecretList>.
<Prompt>	C-20	Required	Prompt. The short prompt to be displayed to the user, explaining the value to be returned in <SecretList> for this secret.
<Memo>	C-255	Optional	Memo. This is a longer description of the meaning of <Prompt> above. This may be displayed to the user if the user requests more information regarding the intended meaning of <Prompt>.
<SecretOptional>	Boolean	Optional	If TRUE, the secret *does not* need to be provided in the <SecretList> sent to the server in order for the server to process the requested message. If FALSE or omitted, the secret *does* need to be provided in the <SecretList> sent to the server in order for the server to process the requested message.
<SecretFormat>	NC-1024	Optional	Secret Format. Regular expression describing the secret format. The definition and behavior of "Regular Expression" is per IEEE Std 1003.2-1992 (POSIX.2). General definition may be found at http://www.ciser.cornell.edu/info/regex.html
<SecretMask>	C-80	Optional	Secret Mask. Edit Mask for the user input for <Secret> in <SecretList>. The format for <SecretMask> is specified by EditMask in Section 3.1.1.

3.2.12.2 Secret List Aggregate <SecretList>

The <SecretList> aggregate provides a way for users to input shared secrets in response to a <SecretPrompt>.

Tag	*Type*	*Usage*	*Description*
<SecretId>	Identifier	Required	ID of Secret. The client must use the value of <SecretId> sent by the server in <SecretPrompt>.
<CryptType>	Open Enum	Required Profiled values	Encryption Type. Defined values: NONE, PKCS#1. Must be supported in list of encryption types in the SP's Service Profile.
<Secret>	C-80	Required XOR	Secret. The secret itself as entered by the user in response to the <Prompt> and/or <Memo> displayed in <SecretPrompt>. <Secret> is used when <CryptType> = NONE

Tag	Type	Usage	Description
<CryptSecret>	Binary, 128	Required XOR	Encrypted Secret. Positional list of customer-entered data corresponding to each secret prompt provided through the <BillerRec> aggregate. Typical uses are mother's maiden name, tax id, etc. <CryptSecret> is used when <CryptType> NONE

3.2.13 Payment Instrument Aggregate <PmtInst>

The <PmtInst> aggregate is used to communicate the types of payment that the biller is capable of accepting via the Pay provider. This is used to restrict the methods by which the customer may pay for the bill. The Service Provider indicates which Payment Instruments are supported via the Biller profile in <BillerPayInfo>. The Biller may use the <PmtInst> in <BillRec> to provide a subset of the types of payments the Biller accepts for a particular bill. Note that errors may occur if this aggregate contains types of payments different from those communicated through the Biller Profile.

Tag	Type	Usage	Description
<PmtInstType>	Closed Enum	Required	Payment type. Specifies the type of payment instruments that the biller is capable of accepting for electronic payment.
<Brand>	Open Enum	Optional	Accepted brand for a given payment type. If not specified, the client assumes that all brands of the given <PmtInstType> are acceptable.
<SettlementInfo>	Aggregate	Optional Repeating	Settlement Information. May contain complete or partial information for use by the CPP in transferring funds to the Biller or BPP. One or more options may be supported for each Payment instrument and brand.
<SettlementMethod>	Open Enum	Required	Settlement Method. The method accepted by the Biller or BPP for settling payments. (Defined values for Direct Deposit: RPS, EPAY, ACH, Concentrator.)
<DepAcctId>	Aggregate	Required OR	Deposit Account Aggregate.
<SettlementId>	C-20	Required OR	Settlement Identification. May contain the RPS or Epay ID as applicable, depending on the Settlement Type.
<OrgContact>	Aggregate	Required OR	Organization Contact. The Biller's Concentrator's contact name and telephone number for the CPP to find out additional information needed to complete payments or for clarification.
<Memo>	C-255	Required OR	Additional human-readable information that may be needed to complete payments or for clarification.
</SettlementInfo>			

4 Security

4.1 Channel Level Security vs. Application Level Security

The IFX Specification is designed to provide messages and data elements necessary to provide Application Level Security on top of an existing secure communications channel between the client and server. This Channel Level Security is not specified in this document, which is intentionally transport-independent. The mechanism used to provide this secure communications channel will be specified in each of the IFX Implementation Specifications associated with this document.

4.2 Application Level Security in the IFX Specification

The IFX Specification supports a Customer Login ID and Password for each Signon Realm that the customer must access for their services. Each Signon Realm may have different rules regarding Customer Password length and composition; the rules may be found in the <SignonInfo> aggregate returned to the client in the <*xxx*Svc>. Each CSP may also independently set its requirements for Customer Password Encryption.

> ***Note:*** *At present, the IFX Specification supports two values for encryption of passwords and other authentication information. The first value is NONE, which relies entirely on channel level security and sends passwords as clear text. The second value is PKCS#1, which enables a password or similar authentication information (up to 117 bytes) to be encrypted by the client or server and passed as a 128-byte binary element that may be decrypted by the receiver. It is assumed that the CSP generates keys with a 1024-bit modulus and distributes them using an "out of band" process.*

Once a client authenticates with a CSP, the client may perform messages for any of the Services enabled by the CSP. If the client wishes to perform a message from a Service that has not been enabled for the current CSP, the client must authenticate with the CSP for which the desired Service has been enabled.

4.3 Security Common Aggregates

4.3.1 Customer Password Aggregate <CustPswd>

Tag	*Type*	*Usage*	*Description*
<CryptType>	Open Enum	Required Profiled values	Encryption Type. Defined values: NONE, PKCS#1. Must be supported in list of encryption types in <SignonInfo> in the CSP's Service Profile.
<Pswd>	C-32	Optional XOR	Clear text Password. Used only when <CryptType> = NONE
<CryptPswd>	Binary, 128	Optional XOR	Encrypted Password. Used when <CryptType> NONE.

4.4 Authentication

4.4.1 Description

Customer Authentication Messages are used to authenticate customers and address security at the application level.

4.4.1.1 Session Key <SessKey>

As an option, a client may request that the server generate and return a Session Key <SessKey> during a <SignonRq>. A Session Key may have an associated Expiration DateTime, after which it may not be used. If a client has a Session Key that has not expired, it may use the <SessKey> in subsequent <SignonRq>s.

This mechanism addresses several requirements:

- It allows a server to efficiently process multiple <SignonRq>s from the same client over a relatively short time period. A batch-oriented client may group messages into a number of batches so that the results of the preceding batch are known when the next is submitted. The <SignonRq> in the first batch must contain a <CustId> and/or any other required authentication information. The <SignonRq> within subsequent batches may then contain the <SessKey> returned by the first <SignonRs>, in lieu of other authentication mechanisms.
- It allows a server to efficiently process multiple <SignonRq>s for the same user, but not necessarily the same client, over a short period of time. For example, a PFM client may invoke a web browser to support a CSP function that is not directly supported by the PFM client. The PFM client may use the <SessKey> obtained from the <SignonRs> to issue a <SignonRq> for the web browser session.
- A batch or interactive client may process multiple Services with different Service Providers; e.g., VRU sends Pay messages to a different SP than Banking. The CSP and *x*SP have a shared authentication mechanism. The client may use the <SessKey> obtained from the <SignonRs> from the CSP as part of the <SignonRq> to the *x*SP, rather than having to either store the user password or prompt the user to enter his/her password a second time.

4.4.2 Signon

The Signon message is used to authenticate an IFX client. Unlike other IFX messages, the <SignonRs> does not echo all elements of the request, for security reasons. Also, because there must be only one <SignonRq> per IFX block, there is no <RqUID> in either the request or the response.

If none of the optional signon methods are provided, the signon is anonymous.

Customer authentication is between the IFX client and the IFX server. Other providers behind the IFX server are expected to rely on the authentication performed by the initial IFX server (note this does not apply if the client is redirected, or handed off, to another CSP). Intermediate elements, such as HTTP proxies, do not participate in the customer authentication exchange. The IFX server may rely on a different server to actually verify the customer password, but this is a local matter.

> ***Note:*** *Signon and Signoff are the only two messages to occur at the top level of IFX, rather than being contained within a service wrapper.*

4.4.2.1 Request <SignonRq>

Tag	*Type*	*Usage*	*Description*
<SignonPswd>	Aggregate	Optional XOR	Signon with <CustId>/<CustPswd> authentication method.

Tag	*Type*	*Usage*	*Description*
<SignonRole>	Open Enum	Optional	Signon role. Defined values: Customer, CSR, Agent Default value is Customer
<CustId>	Aggregate	Required	Customer Identification Aggregate.
<CustPswd>	Aggregate	Required	Customer Password Aggregate.
<GenSessKey>	Boolean	Optional	Session Key Requested Indicator. TRUE indicates that the client is requesting a Session Key in the response. FALSE or absent indicates that the client is not requesting a Session Key in the response.
</SignonPswd>			
<SignonCert>	Aggregate	Optional XOR	Signon with embedded certificate.
<SignonRole>	Open Enum	Optional	Signon role. Defined values: Customer, CSR, Agent Default value is Customer
<CustId>	Aggregate	Required	Customer Identification Aggregate.
<Certificate>	Binary	Required	
<GenSessKey>	Boolean	Optional	Session Key Requested Indicator. TRUE indicates that the client is requesting a Session Key in the response. FALSE or absent indicates that the client is not requesting a Session Key in the response.
</SignonCert>			
<SignonTransport>	Aggregate	Optional XOR	Signon with transport certificate (i.e., SSL) authentication method.
<SignonRole>	Open Enum	Optional	Signon role. Defined values: Customer, CSR, Agent Default value is Customer
<CustId>	Aggregate	Required	Customer Identification Aggregate.
<GenSessKey>	Boolean		Session Key Requested Indicator. TRUE indicates that the client is requesting a Session Key in the response. FALSE or absent indicates that the client is not requesting a Session Key in the response.
</SignonTransport>			
<SignonMagPIN>	Aggregate	Optional XOR	Signon with a magnetic stripe card and a PIN pad
<CardMagData>	Aggregate	Required	Card Magnetic Stripe Data
<PINBlock>	NC-32	Required	PIN pad data
<GenSessKey>	Boolean		Session Key Requested Indicator. TRUE indicates that the client is requesting a Session Key in the response. FALSE or absent indicates that the client is not requesting a Session Key in the response.
</SignonMagPIN>			
<SessKey>	NC-64	Optional XOR	Authentication Key. Issued previously based on Password. Valid only for a fixed interval, as set by the server.
<ClientDt>	DateTime	Required	Client DateTime. Time according to the client. ***Note:*** *This is typically used by customer service to resolve problems regarding cut-off time or timeliness of customer messages. It may be compared with the server time to determine whether there is a discrepancy.*
<CustLangPref>	NC-6	Required Profiled values	The Client Language Preference has the same valid values as all other language elements. Must be one of those supported in the Service Profile.

Tag	Type	Usage	Description
<ClientApp>	Aggregate	Required	Client Application Aggregate. Manufacture, name, and version of the client endpoint of the IFX message.
<ProxyClient>	Aggregate	Optional	Proxy Client Aggregate. In the case where the IFX client is composing requests for some other application that the user is running (e.g., a Web browser), the details of the customer's interface application should be included here. It has the same structure as the <ClientApp> aggregate.
<EU.Cur>	Closed Enum	Optional	Euro currency selection. Used only in countries supporting the euro currency. Local indicates that: 1. Unless otherwise specified by use of <CurCode>, all messages initiated by the client are in the currency of the account 2. All statement downloads are in the currency of the account. Euro indicates that all messages initiated by the client are in euros (€), and that all statement downloads that support euros are to be provided in euros.

4.4.2.2 Response <SignonRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<SignonRole>	Open Enum	Optional Echoed	Signon role. Defined values: Customer, CSR, Agent Default value is Customer
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate.
<GenSessKey>	Boolean	Optional Echoed	User Key Requested Indicator.
<ClientDt>	DateTime	Required Echoed	Client DateTime.
<CustLangPref>	Aggregate	Required Echoed	Customer Language Preference Aggregate.
<ClientApp>	Aggregate	Required Echoed	Client Application Aggregate.
<ProxyClient>	Aggregate	Optional Echoed	Proxy Client Aggregate.
<EU.Cur>	Closed Enum	Optional Echoed	Euro Currency Selection.
<ServerDt>	DateTime	Required	Server DateTime. System time according to the server.
<SessKey>	NC-64	Optional	User Key. Included only at client request in <SignonRq>.
<ExpDt>	DateTime	Optional	Session Key Expiration DateTime. Included only if <SessKey> is returned.

Tag	Type	Usage	Description
<Language>	NC-11	Required	Language. The value of <Language> follows the definition of language identifiers in IETF RFC 1766. It is in the form of the two letter language code defined by ISO 639, followed by a hyphen, followed by the two letter country code defined by ISO 3166 or a local dialect name of up to 8 characters. Note that RFC 1766 states that the langauge tag is case insensitive. May or may not be the same as the Customer Preferred Language in <SignonRq>.

4.4.3 Signoff

A client that has no further messages to perform should send a <SignoffRq> to indicate to the CSP that no more messages will be sent in this session and server resources may be freed and reallocated to other customers. The client may begin another session whenever it has new messages to perform by using the <SignonRq>. Because there may be only one <SignoffRq> per IFX block, there is no <RqUID> in either the request or the response.

> ***Note:*** *Signon and Signoff are the only two messages to occur at the top level of IFX, rather than being contained within a service wrapper.*

4.4.3.1 Request <SignoffRq>

Tag	Type	Usage	Description
<CustId>	Aggregate	Optional	Customer ID. Provided only in the case where a CSR is forcing a Signoff of a customer.

4.4.3.2 Response <SignoffRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<CustId>	Aggregate	Optional Echoed	Customer ID.

5 The Base Service <BaseSvc>

The Base Service in IFX includes messages that perform basic communication functions that must take place between parties (client and server or between different service providers). These messages are grouped into functions including Service Profile, Customer Authentication, Customer Profile, General Email, and Customer Service.

IFX uses Service Profile messages to communicate information about a Service Provider's services and configuration to a client.

Customer Profile Messages provide for the registration of Customers with Service Providers, and the maintenance of that information.

Customer Service messages provide administrative functions to assist Customer Service Representatives in serving customers.

5.1 Base Service Message Summary

Function / Message Name	*Required*	*Comments*
Service Profile Inquiry **<SvcProfInqRq>** **<SvcProfInqRs>**	Yes	Allows the client to retrieve a current view of the services supported by the SP, the Authentication Realms that must be used for authentication, and basic information about the SP.
Holiday Inquiry **<HolInqRq>** **<HolInqRs>**		Allows a client to retrieve a list of Bank Holidays observed by the Service Provider.
Customer Add **<CustAddRq>** **<CustAddRs>**		Allows client to provide customer personal details and indicate desire to use IFX Specification to communicate with a Financial Institution. SP may respond with Customer Login ID and Password or may provide these to the customer through another channel.
Customer Modify **<CustModRq>** **<CustModRs>**		Allows client to modify previously provided customer name, address, and contact information.
Customer Password Modify **<CustPswdModRq>** **<CustPswdModRs>**		Allows a client to modify a Customer Password.
Customer Status Modify **<CustStatusModRq>** **<CustStatusModRs>**		The Customer Status Modify Message is used primarily by CSRs to reset the status and counters that were set when a user exceeded the invalid password threshold.
Customer Delete **<CustDelRq>** **<CustDelRs>**		Allows client to unenroll, deleting previously provided customer name, address, and contact information.
Customer Inquiry **<CustInqRq>** **<CustInqRs>**	Yes	Allows client to retrieve current customer profile, including customer name, address, and contact information.
Customer Identifier Inquiry **<CustIdInqRq>** **<CustIdInqRs>**		Allows client to retrieve either the login id corresponding to a permanent id, or the permanent id corresponding to a login id.

Function / Message Name	*Required*	*Comments*
Customer Audit **<CustAudRq>** **<CustAudRs>**		Allows client to audit Customer messages associated with the current customer.
Customer Synchronize **<SvcAcctSyncRq>** **<SvcAcctSyncRs>**		Allows client to synchronize Customer messages associated with the current customer.
Custmer/Service Link Add **<CustSvcAddRq>** **<CustSvcAddRs>**		Allows client to request the enabling of a service for a customer. Optionally allows an SP to return terms and conditions to the customer for acceptance.
Customer/Service Link Modify **<CustSvcModRq>** **<CustSvcModRs>**		Allows the client to modify the fee account for a service.
Customer/Service Link Delete **<CustSvcDelRq>** **<CustSvcDelRs>**		Allows client to request the disabling of a service for the customer.
Service/Account Link Add **<SvcAcctAddRq>** **<SvcAcctAddRs>**		Allows client to request activation of an account for one of the customer's services. Optionally allows an SP to return terms and conditions to the customer for acceptance.
ServiceAccount Link Modify **<SvcAcctModRq>** **<SvcAcctModRs>**		Allows client to request modification of an account for one of the customer's services. Optionally allows an SP to return terms and conditions to the customer for acceptance.
Service/Account Link Delete **<SvcAcctDelRq>** **<SvcAcctDelRs>**		Allows client to request deactivation of an account for one of the customer's services.
Service/Account Link Inquiry **<SvcAcctInqRq>** **<SvcAcctInqRs>**		Allows client to retrieve account and service current state, and account balances.
Service/Account Link Audit **<SvcAcctAudRq>** **<SvcAcctAudRs>**		Allows client to audit Service Enable and Disable and Account Activation and Deactivation messages associated with the current customer.
Service/Account Link Synchronize **<SvcAcctSyncRq>** **<SvcAcctSyncRs>**		Allows client to synchronize Service Enable and Disable and Account Activation and Deactivation messages associated with the current customer.
Customer/Disclosure Link Status Modify **<CustDiscStatusModRq>** **<CustDiscStatusModRs>**		Allows client to accept a disclosure received as part of the service enable/disable/modification and account activation/deactivation/modification process.
Customer/Disclosure Link Inquiry **<CustDiscInqRq>** **<CustDiscInqRs>**		Allows a customer to inquire about his/her associated disclosures for accounts or services.

5.2 Base Service Common Elements and Aggregates

5.2.1 Service Profile Common Elements

5.2.1.1 Messages Supported Element <MsgSupt>

The <MsgSupt> element is included in the Service Profile for each service. It provides the client with a list of supported messages for that service.

Tag	*Type*	*Usage*	*Description*
<MsgSupt>	Open Enum		Supported Messages. This is a list of messages that are supported for the current service. The convention is to use the name of the message without the Rq or Rs so that each message is only listed once.

5.2.1.2 Options Supported Element <OptSupt>

The <OptSupt> element is included in each Service Profile where there are options that the client should know are supported or not supported by the server. Unlike the <MsgSupt> element, which appears exactly once in the Service Profile for each service, the <OptSupt> element may appear more than once in a Service Profile, since different functions supported in the same service may have different option settings. Where <OptSupt> appears more than once in a Service Profile, the options it specifies are scoped to the aggregate that contains the <OptSupt> element.

Tag	*Type*	*Usage*	*Description*
<OptSupt>	Open Enum		Options Supported. Defined values: **AcctNickname**—Account Nickname. If included, the service provider stores user-assigned account nicknames. The nicknames are defined out of band or via the Account Activate message, and may be modified via the Account Modify message. If not included, account nicknames are not stored. **BillerPayee**—Biller Payee. If included, the Pay provider supports electronic payments to payees that are billers. The remittance and settlement information may be copied from the biller directory entry <BillerRec> returned by the Biller Inquiry message. If not included, payments to biller payees are not supported. **BillStatus**—Bill Status. If included, the BSP maintains the Bill Status information that flows from the Biller and from the CSP via <BillStatusModRq>. **CustPayeeNickname**—Customer Payee Nickname. If included, the Pay provider stores user-assigned payee nicknames. If not included, payee nicknames are not stored. **ForEx**—Foreign Exchange. If included, client may specify a transfer or payment amount in a currency other than that of the funding account. If not included, transfer or payment amount must be in currency of funding account. **ForExCommit**—Foreign Exchange Commitment. If included, client may request an exchange rate commitment via the ForExInq messages and specify a previously committed exchange rate in a foreign exchange transfer by including the <ForExCommit> aggregate. If not included, Foreign Exchange Commitments are not supported.

Tag	Type	Usage	Description
			FSPayee—Fully Specified Payee. If included, the Pay provider supports payments to payees based on name, address, and, optionally, phone number provided by the customer. If not included, payments to payees with customer-provided address information are not supported. **ImmediateXfer**—Immediate Transfers. If included, the provider must execute transfers that set the <ImmediateXfer> element to TRUE in real time. If not included, all transfers are assumed to be executed through end of day processing. **LineItem**—Line Item. If included, the Pay provider supports use of the <LineItem> aggregate within the <InvoiceInfo> aggregate. If not included, invoice line items are not supported. **RecCtrl**—Records Control. If included, the server reserves the right to return an incomplete result set for an Inquiry, Audit or Sync message and indicate that more matching records are available using the the <RecCtrlOut> aggregate. In this case, the client must perform one or more additional requests to retrieve the rest of the result set. If not included, the server must always send the complete set of records that match the criteria specified in the request message. **RecFinalCurAmt**—Recurring Model Final Currency Amount. If included the provider supports use of the <FinalCurAmt> element in the <RecModelInfo> aggregate to allow customers to specify a final transfer or payment generated from a recurring model for an amount different from the nominal amount of the model. If not included, the final transfer or payment generated from a recurring transfer or payment model must be for the nominal amount. **RecInitialCurAmt**—Recurring Model Initial Currency Amount. If included, the provider supports use of the <InitialCurAmt> element in the <RecModelInfo> aggregate to allow customers to specify an initial transfer or payment generated from a recurring model for an amount different from the nominal amount of the model. If not included, the first transfer or payment generated from a recurring transfer or payment model must be for the nominal amount. **RecPend**—Recurring Pending. If included, the server generates pending payments/transfers prior to the date required to process them. If not included, no pending payments/transfers are generated. **SchedXfer**—Scheduled Transfers. If included, the provider supports scheduled future transfers. If not included, but transfers are supported, they must be pay today (end of day processing) or immediate. **SkipInst**—Skip Instances. If included, the service provider supports use of the <SkipNextN> element in the Modify Recurring Payment/Transfer/Interbank Transfer Model Messages. If not included, the service provider does not support skipping payments/transfers generated from a recurring payment/transfer model. **StopChkRangeSC**—Stop Check Range Selection Criteria. If included, the server supports using the <ChkRange> aggregate in Stop Check Add and Cancel messages. If not included, <ChkRange> may not be used in request messages for Stop Check functions. **XferPayee**—Transfer Payee. If included, the Pay provider supports payments to payees based on customer-provided bank and account information (i.e., the customer provides the bank account number of the payee). If not included, payments to payees with customer-provided bank and account information are not supported.

The following table identifies the services that use each option support value.

<OptSupt> value	*Base*	*Bank*	*Pay*	*Pres*
AcctNickname		×	×	×
BillerPayee			×	
BillStatus				×
CustPayee			×	
CustPayeeNickname			×	
ForEx		×	×	
ForExCommit		×		
FSPayee			×	
ImmediateXfer		×		
LineItem			×	
RecCtrl	×	×	×	×
RecFinalCurAmt		×	×	
RecInitialCurAmt		×	×	
SchedXfer		×		
SkipInst		×	×	
StopChkRangeSC		×		
XferPayee			×	

5.2.2 Service Profile Common Aggregates

5.2.2.1 Service Core Aggregate <SvcCore>

The <SvcCore> aggregate provides information about the service that is critical for the correct processing of that service.

Tag	*Type*	*Usage*	*Description*
<Version>	NC-12	Required	The version of IFX that the server supports. May be used by the server to infer that any tags added beyond this level are not supported.
<SPName>	Identifier	Required	Service Provider Name.
<LegalName>	C-96	Required	SP Legal Name.
<Nickname>	C-40	Optional	SP Nickname.
<HoldCoId>	Identifier	Optional	SP Holding Company Identifier. A unique identifier assigned by a service provider to identify the holding company that owns the SP branding company.
<USA.RTN>	NC-9	Optional	Routing and Transit Number. This is the default RTN for a customer account associated with the SPName above.
<OrgContact>	Aggregate	Required Repeating	Service Provider Contact Aggregate. Usage is contact information for reaching relevant organizations at the SP (for example, the customer services area).

Tag	Type	Usage	Description
<Language>	NC-11	Required Repeating	Language. The value of <Language> follows the definition of language identifiers in IETF RFC 1766. It is in the form of the two letter language code defined by ISO 639, followed by a hyphen, followed by the two letter country code defined by ISO 3166 or a local dialect name of up to 8 characters. Note that RFC 1766 states that the langauge tag is case insensitive. If the server supports more than one language for this service, multiple Language aggregates may be listed.
<URL>	URL	Optional	URL. <URL> identifies where the client should send the request for that service.

5.2.2.2 Service Profile Information Aggregate (*xxx*SvcProfInfo)

The Service Profile Information aggregate is used to express profile options for particular services. A list of these aggregates comprises part of the Service Profile Inquiry Response message <SvcProfInqRs> that is used to communicate server capabilities to a client.

Tag	Type	Usage	Description
<SvcCore>	Aggregate	Required	Service Core Aggregate. Information specified for every service.
Service specific tags			Other tags as appropriate for profiling options for the specific service.

The definition of this aggregate is in the last section of each service chapter.

5.2.2.3 Processing Schedule Aggregate <PrcSched>

The <PrcSched> aggregate appears wherever information about a Service Provider's processing schedule is needed. Note that this aggregate does not take holidays into account.

Tag	Type	Usage	Description
<PrcDaysOff>	Closed Enum	Optional Repeating	Processing Days Off. Days of the week on which no processing occurs. Valid values: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, Holiday
<CutoffTm>	Time	Required	Cutoff Time for Daily Processing. This is the latest time a customer may submit a request for processing today.
<PrcDtAdj>	Open Enum	Required	Processing Date Adjustment. Algorithm used for adjustment of processing for requests that fall on non-processing days or holidays. Defined values: Earlier, Later.

5.2.3 Customer Profile Common Aggregates

5.2.3.1 Customer Record Aggregate <CustRec>

Tag	Type	Usage	Description

Tag	Type	Usage	Description
<CustId>	Aggregate	Required	Customer Identification Aggregate.
<CustInfo>	Aggregate	Optional	
<CustStatus>	Aggregate	Required	
<UpDt>	DateTime	Optional	Customer Record Update DateTime. The value is for the last update to <CustInfo> on the server.

5.2.3.1.1 Customer Information Aggregate <CustInfo>

The Customer Information aggregate <CustInfo> contains information about the customer, such as name, address, and contact information.

Tag	Type	Usage	Description
<CustName>	Aggregate	Required	Customer Name Aggregate.
<PostAddr>	Aggregate	Optional Profiled requirement	Customer Postal Address Aggregate.
<CustContact>	Aggregate	Optional Profiled requirement	Customer Contact Aggregate.
<TaxId>	NC-12	Optional	Customer Tax ID.
<CustType>	Open Enum	Optional	Customer Type. Defined values are Retail and Business. The type of relationship established between the CSP and the customer. Retail indicates an individual relationship, and Business indicates a small business relationship. Both personal and business accounts may exist within a Business relationship. <CustAcctUse> within <BankAcctRec> indicates the type of account.

5.2.3.1.2 Customer Status Aggregate <CustStatus>

Tag	Type	Usage	Description
<CustStatusCode>	Open enum	Required	Customer Status Code. Defined values: Enrolled, EnrollPend, Suspended, UnenrollPend, Unenrolled
<StatusDesc>	C-255	Optional	Status Description. Explanatory text associated with this service status.
<EffDt>	DateTime	Optional	Effective Date/Time. The date/time the <CustStatusCode> took effect.
<StatusModBy>	Open Enum	Optional	
<ResolvePendingDt>	Date	Optional	Resolve Pending Date. The date that the *xxx*Pend status in <CustStatusCode> is expected to change to a final status; e.g., Enrolled, Unenrolled. This date should be returned when the <CustStatusCode> is set to *xxx*Pend.

5.2.3.2 Customer/Disclosure Link Record Aggregate <CustDiscRec>

The <CustDiscRec> aggregate allows the CSP/SP to send Terms and Conditions to the customer. The CSP/SP may optionally require acceptance of these terms and conditions prior to enabling a service or activating/modifying an account for a service.

Tag	Type	Usage	Description
<CustId>	Aggregate	Required	Customer Identifier
<DiscId>	UUID	Required	Disclosure Identifier
<DiscInfo>	Aggregate	Required	Disclosure Information
<CustDiscStatus>	Aggregate	Required	

5.2.3.2.1 Disclosure Information Aggregate <DiscInfo>

Tag	Type	Usage	Description
<LongText>	C-	Required XOR	Disclosure Text.
<DiscURL>	URL	Required XOR	Disclosure URL. The address for obtaining human-readable disclosure information.
<AcceptReqd>	Boolean	Optional	Disclosure Accept Required—The server indicates whether the acceptance of this disclosure is required to proceed. If set to TRUE, acceptance is required.

5.2.3.2.2 Customer/Disclosure Link Status Aggregate <CustDiscStatus>

Tag	Type	Usage	Description
<CustDiscStatusCode>	Closed Enum	Required	Customer/Disclosure Link Status Code Defined Values: • None—The customer has not accepted, rejected, or viewed the disclosure • Accepted—The customer accepted the terms of the disclosure • Rejected—The customer rejected the terms of the disclosure • Viewed—The customer viewed the disclosure, but neither explicitly accepted nor explicitly rejected it.
<StatusDesc>	C-255	Optional	Status Description. Explanatory text associated with this customer/disclosure link status.
<EffDt>	DateTime	Optional	Effective Date/Time. The date/time the <CustDiscStatusCode> took effect.
<StatusModBy>	Open Enum	Optional	

5.2.3.3 Customer/Service Link Record Aggregate <CustSvcRec>

Tag	Type	Usage	Description
<CustSvcId>	Aggregate	Required Echoed	Customer/Service Link Identification Aggregate.
<CustSvcInfo>	Aggregate	Required Echoed	Customer/Service Link Information Aggregate.
<CustSvcStatus>	Aggregate	Required	Customer/Service Status Aggregate.

5.2.3.3.1 Customer/Service Link Identification Aggregate <CustSvcId>

Tag	*Type*	*Usage*	*Description*
<SvcName>	Open Enum	Required	Service Name.
<SPName>	Identifier	Required	Service Provider Name.

5.2.3.3.2 Customer/Service Link Information Aggregate <CustSvcInfo>

Tag	*Type*	*Usage*	*Description*
<DepAcctId>	Aggregate	Optional XOR	Deposit Account Aggregate. Used to specify a deposit account to be used for charging fees for the service.
<CardAcctId>	Aggregate	Optional XOR	Credit Account Aggregate. Used to specify a credit card account to be used for charging fees for the service.

5.2.3.3.3 Customer/Service Link Status Aggregate <CustSvcStatus>

Tag	*Type*	*Usage*	*Description*
<CustSvcStatusCode>	Closed Enum	Required	Customer/Service Status Code. Defined values: Enabled, EnablePend, Disabled, DisablePend, Rejected
<StatusDesc>	C-255	Optional *but see Description*	Status Description. Explanatory text associated with this customer/service link status. Required if service change is rejected.
<EffDt>	DateTime	Optional	Effective Date/Time. The date/time the <CustSvcStatusCode> took effect.
<StatusModBy>	Open Enum	Optional	
<ResolvePendingDt>	Date	Optional	Resolve Pending Date. The date that the *xxx*Pend status in <CustSvcStatusCode> is expected to change to a final status; e.g., Enabled, Rejected. This date should be returned when the <CustSvcStatusCode> is set to *xxx*Pend.

5.2.3.4 Service/Account Link Record Aggregate <SvcAcctRec>

Tag	*Type*	*Usage*	*Description*
<SvcAcctId>	Aggregate	Required	Service/Account Link Identifier Aggregate.
<SvcAcctInfo>	Aggregate	Required	Service/Account Link Information Aggregate.
<SvcAcctStatus>	Aggregate	Required	Service/Account Status Aggregate.

Tag	Type	Usage	Description
<EffDt>	Date	Optional	Effective Date. The date that the first account information (i.e. statement, bill, etc.) is expected to be available. This date should be returned when the <SvcAcctStatusCode> for an account is set to Activated. For billing accounts, this is the date of the first available bill. If absent, the semantic meaning is that information is available immediately.

5.2.3.4.1 Service/Account Link Identification Aggregate <SvcAcctId>

Tag	Type	Usage	Description
<SvcName>	Open Enum	Required	Service Name.
<SPName>	Identifier	Optional *but see Description*	Service Provider Name. May be provided by the client to clearly identify which SP's service is being requested, in cases where the Service Profile indicates that more than one SP provides the requested service. If <SPName> is absent and there are multiple providers for the service, the semantic meaning is to enable the account for the service at *all* providers of that service.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identifier Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identifier Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identifier Aggregate.
<PresAcctId>	Aggregate	Required XOR	Presentment Account Identifier Aggregate.
<PresAcctInfo>	Aggregate	Required	Presentment Account Information Aggregate.

5.2.3.4.2 Service/Account Link Information Aggregate <SvcAcctInfo>

Tag	Type	Usage	Description
<Nickname>	C-40	Optional	Account Nickname, as held by the Service Provider.

5.2.3.4.3 Service/Account Link Status Aggregate <SvcAcctStatus>

Tag	Type	Usage	Description
<SvcAcctStatusCode>	Closed Enum	Required	Customer/Service Status Code. Defined values: Activated, ActivatePend, Deactivated, DeactivatePend, Rejected
<StatusDesc>	C-255	Optional *but see Description*	Status Description. Explanatory text associated with this customer/service link status. Required if service change is rejected.
<EffDt>	DateTime	Optional	Effective Date/Time. The date/time the <SvcAcctStatusCode> took effect.
<StatusModBy>	Open Enum	Optional	

Tag	Type	Usage	Description
<ResolvePendingDt>	Date	Optional	Resolve Pending Date. The date that the *xxx*Pend value in <SvcAcctStatusCode> is expected to change to a final status; e.g., Activated, Rejected. This date should be returned when the <SvcAcctStatusCode> is set to *xxx*Pend.

5.3 Service Profile

5.3.1 Description

The Service Profile has two messages, a Service Profile Inquiry message, and a Holiday Inquiry message. The Holiday Inquiry Message allows a client to retrieve a list of Bank Holidays observed by a Financial Institution or Service Provider.

The Service Profile Inquiry Message allows a client to retrieve the Service Profile from a Financial Institution, CSP, or other Service Provider.

The Service Profile contains the following information about the Customer's Service Provider:

- Services Supported—Services are collections of messages that are functionally related. Each Service is designed with a corresponding section in the Service Profile to allow each Service Provider to inform clients as to which messages, functions, and options of the service supported.
- Signon Realms—Signon Realms provide a mechanism for a Service Provider to manage customer authentication across services. Service Providers may choose not to support IFX authentication when communicating to other service providers. Server-to-server authentication may take place in some other way.
- Service Provider Information—Provides the legal name of the SP, the SP address, the SP's Customer Service telephone number, and other useful information for that service provider.

Each time a client authenticates with a Signon Realm, the time and date of the current Service Profile is returned by the server. If the client does not have a copy of the current Service Profile, it should perform a Service Profile Inquiry message to retrieve the current Service Profile. This mechanism is used to inform clients of changes in server capabilities.

***Note**: All IFX servers must support the Service Profile Inquiry message.*

5.3.2 Service Profile Inquiry Message

5.3.2.1 Request <SvcProfInqRq>

The client submits a <SvcProfInqRq> that optionally includes a DateTime value <UpDt> for the last Service Profile received from the server. <UpDt> indicates when the Service Profile was last updated on the server. If <CustId> is not available (because of an anonymous login, for example), a generic profile must be returned. If <CustId> is available, then a custom profile for that customer may be returned. This capability allows the SP to support capabilities such as: (1) providing different contact information for preferred customers or (2) providing information specific to service providers for this particular customer when the SP has relationships with multiple service providers for the same service.

Tag	Type	Usage	Description
<UpDt>	Timestamp	Optional	Timestamp when the Service Profile was Last Updated on the Server. This is the timestamp of the last Service Profile received from the server.

5.3.2.2 Response <SvcProfInqRs>

The Service Profile Inquiry Response Message returns the following information in the following order:

- Sections for each service that the SP supports, including the parameters for that service, and the associated signon realm; Information about the SP, e.g. name, address, contact information, is included in <SvcCore> for the Base service, and
- Information about the signon realms.

If there have been no updates to the Service profile, the server returns a Response Code of 1 in the <Status> aggregate to indicate that the client has the most current Service Profile. In this case, the server does not return the <*xxx*SvcProfInfo> aggregates.

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<UpDt>	Timestamp	Optional Echoed	Timestamp when the Service Profile was Last Updated on the Server. The last timestamp received by the client.
<*xxx*SvcProfInfo>	Aggregate	Optional Repeating	Service Profile Information Aggregate. At least one per service supported.

5.3.3 Holiday Inquiry Message

The Holiday Inquiry message allows a client to retrieve a list of Bank Holidays observed by the Service Provider. The list of holidays is typically used to adjust the processing date for a scheduled payment or transfer.

5.3.3.1 Request <HolInqRq>

The client submits a <HolInqRq> in order to request a list of the Service Provider's observed holidays.

Tag	*Type*	*Usage*	*Description*
<SPName>	Identifier	Optional *but see Description*	Service Provider Name. If omitted, the receiver of the message must determine (1) that the message is unambiguous and provide a successful response or (2) that it is ambiguous and provide an error response.
<SvcName>	Open Enum	Optional *but see Description*	Service Name. If omitted, the receiver of the message must determine (1) that the message is unambiguous and provide a successful response or (2) that it is ambiguous and provide an error response.

5.3.3.2 Response <HolInqRs>

The server at the Financial Institution or Service Provider responds with a list of observed Bank Holidays if available.

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<SPName>	Identifier	Optional Echoed	Service Provider Name.
<SvcName>	Open Enum	Optional Echoed	Service Name.
<HolInfo>	Aggregate	Optional Repeating	Holiday Information Aggregate.
<Name>	C-40	Required	Holiday Name.
<HolDt>	Date	Required	Holiday Date.

Tag	Type	Usage	Description
</HolInfo>			

5.4 Customer Profile

5.4.1 Description

Customer Profile Messages provide the following functions:

- ***Enrollment and Acquisition of Login ID and Password.*** An Enrollment message that allows an individual to submit enough personal information to be identified to a Service Provider is provided. If the SP supports customer self-enrollment using the IFX Specification, it may elect to return a login ID and password to the customer. Alternately, the SP may collect the customer's information and use a separate process (possibly a mailing or a customer service telephone call) to provide the customer with a login and password.
- ***Updating of Customer Name, Address, and Contact Information.*** After the customer initially provides his or her personal information through the enrollment process, he or she may need to update this information periodically. The Modify Customer Information message is provided for this purpose.
- ***Customer Profile***. The Customer Profile message provides a current view of the customer's information, accounts balances, and services. It is useful both in initial setup and on an ongoing basis.
- ***Service Enabling and Disabling***. Services may be enabled for a customer, independently of any account. The customer and CSP may use the Customer/Service Link Add and Customer/Service Link Delete messages to manage which services are currently activated for the customer. A fee account may optionally be specified. Any fees attributable to the use of the service are charged to the fee account.
- ***Service Modification.*** The Customer/Service Link Modify message is used to change the fee account.
- ***Account Activation and Deactivation.*** Accounts may be activated for use by a specific service. The customer and CSP may use the Service/Account Add and Service/Account Delete messages to manage which accounts are currently activated for each service.
- ***Account Modification.*** The Service/Account Modify message is used to modify a funding account for the Pay service, or to change an account nickname.

5.4.1.1 Account Activation and Deactivation

Accounts are established with an SP and may be used electronically if activated for a service; e.g., a checking account may be activated for electronic banking (Bank service). Another CSP/ SP may reference the account; e.g., the Pay service provider may use it as a funding account. Thus, the meaning of activating an account for a service is specific to the service.

The use of an account may be constrained based on a combination of the type of account, its tax status, and CSP/SP policy. For example:

1. Neither an IRA <TaxStatus>=TaxDeferred) nor a certificate of deposit (CD) account may be a funding or fee account for Pay.

2. A CSP might not allow Interbank transfers from a savings (SDA) account.

The allowable usage for a bank account is specified by a series of aggregates <XferFromSupt>, <XferToSupt>, and <PaySupt> within the Bank Account Information Aggregate <BankAcctRec>. These tags are sometimes referred to as the account's message authority.
The following table defines the meaning of activating an account for a service.

Service	Activation Means	Comments

Base	Not applicable	
Bank	Banking Accounts	Allows all banking services subject to message authority.
Pay	Funding Accounts	
Pres	Biller Accounts	Customer's account at the biller.

Services that normally require an account to be associated with them may remain enabled even when no account is activated. These situations are typically temporary conditions. For example, a customer's checkbook for the sole Pay funding account may be stolen. The checking account is closed immediately to reduce the risk of loss, and there is a processing delay while a replacement account number is assigned. The Pay service must remain enabled during the processing delay, even though there was no funding account specified, in order to maintain the customer payee list, pending payments, and payment models that are defined. The Modify Account message is used to replace an existing account (e.g., change a funding account for Pay), change information for a Bill Presentment account, and modify the account nickname

> ***NOTE****: The Enable/Modify/Disable Service and Activate/Modify/Deactivate Account messages are intentionally limited to a single service/account per message to simplify error reporting when the request is rejected by the CSP/SP.*

5.4.1.2 Disclosures

Services may have terms and conditions associated with them. These terms and conditions may be a combination of fees, terms mandated by statute, and terms required by the CSP/SP. Banking regulations and contract law may restrict the acceptable means of notification; e.g., US Mail to the address of record in the US.

Some services, notably Bill Presentment, may require disclosures at the account level. In Bill Presentment, individual Billers may need to disclose their specific terms and conditions. Thus, disclosures may be associated with account-level messages as well as with service-level messages.

Disclosures are presented when a service is enabled or an account is activated. Changes to the terms and conditions that were presented to the customer may also need to be presented. These may be considered to be "server-initiated changes" and are conveyed to the client through standard IFX mechanisms described below.

IFX provides both synchronous and asynchronous capabilities to send disclosures to the customer and to optionally transmit a confirmation from the customer that he or she accepts or rejects the terms and conditions.

- In synchronous mode, disclosures are sent to the client as part of the Enable/Modify Service or Activate/Modify Account Rs, and the client sends confirmations using the Accept Disclosure message.
- In asynchronous mode, disclosures are sent to the client using either (1) the Service/Account Audit Message, with confirmations sent using the Accept Disclosure Message, or (2) using the IFX email facility, with confirmations sent by the client using the Manage Email message.

The SP must indicate in the appropriate Rs message whether a disclosure should be presented to the customer for information only or whether it requires a confirmation. Some SPs require the user to reenter their password as part of accepting a disclosure; since multiple people may have access to the PC keyboard, and the user may have stepped away after displaying the disclosures. This is a step towards "non-repudiation." The Accept Disclosures message contains an optional aggregate to contain the password.

Some SPs also require disclosures to be sent and confirmed when an additional account is activated for an existing service. In this case, the disclosure is sent as part of the Activate/Modify Account Rs.

In the event that a Biller needs to republish disclosure information after the initial Bill Presentment account activation, they may choose to send a Bill Summary with the <BillType>=Notice. The Notice provides the ability to send any type of notification information to a customer using the <Memo> field. For disclosure-type notices, the <Memo> field may contain the disclosure text or may give instructions for accessing a URL. Note that this process of utilizing the Bill Summary for the delivery of disclosure information does not provide the Biller with the ability to receive a customer acceptance response.

5.4.1.3 Customer Profile Inquiry

IFX requires that a Service Provider generate a customer profile for each customer who identifies himself or herself through an enrollment process. Once a customer is identified, the profile may be automatically generated to include information about the customer and his or her accounts.

The Customer Profile Inquiry message provides an efficient mechanism for client software to determine the current state of the customer's accounts and what services are activated. This capability is especially important for customers who use multiple channels or multiple software clients to interact with an SP.

The format of the response to a Customer Profile Inquiry is constant regardless of which SP generates it; however, the content may differ. For example, a CSP may return information about all of the checking accounts for a customer, while the Pay Service Provider may choose to return only the identifiers of the checking accounts that have been identified to it as funding accounts.

5.4.1.4 Using the Customer Profile

A list of available accounts and services is often displayed to the end user so that he or she may select from them. This technique is preferred over asking the user to enter the information since it eliminates the possibility of errors when the user keys in account identifiers; e.g., account number, account type, routing and transit numbers.

The list of services available to the customer, or being used by the customer, is contained within <CustSvcRec>. A service is available (Available) if the SP offers the service and the specific customer may enable it.

The list of available accounts is contained within <CustSvcRec>. The message authority aggregates noted above should be used for displaying appropriate candidates for use within a specific service.

5.4.2 Customer

5.4.2.1 Customer Add Message

The Customer Add message is used to add a Customer record at a Service Provider. It also allows an individual to submit enough personal information to be identified to a Customer Service Provider. If the CSP supports customer self-enrollment using the IFX Specification, it may elect to return a login ID and password to the customer. Alternatively, the CSP may collect the customer's information and use a separate process (possibly a mailing or a customer service telephone call) to provide the customer with a login and password.

A CSP may choose not to support the Customer Add functionality defined in IFX for enrollment, and may provide a customized World Wide Web (WWW) page or some other means to support its own enrollment process. In this case, the CSP may communicate the URL for this page in the Service Profile. A short note about an alternative enrollment process may also be provided to the client through the Service Profile.

5.4.2.1.1 Request <CustAddRq>

If the CSP supports customer self-enrollment using this message, the CSP may specify what information it must receive to authenticate the customer using the <SecretPrompt> aggregate within the <EnrollProf> aggregate in the Service Profile. Information requested through this mechanism typically includes things like Tax ID, Date of Birth, or Mother's Maiden Name. The customer must provide this information in the <CustAddRq> to prove that the enrollment is valid against existing customer information stored in the SP customer database.

The secret list is provided so that a customer new to this channel may be authenticated against existing customer records at the SP. Note that this is a variable-length list, depending on the requirements of the SP as expressed in <SecretPrompt> in the Service Profile. The client should display each <SecretPrompt> and collect customer input. This information is returned in the <SecretList> either in clear text (relying on channel security) or encrypted

> ***Note****: The Customer Add message requires special handling because it does not assume that a client is authenticated. By definition, a client that is performing an enrollment using the Customer Add message*

does not have a Customer Login ID or Password. Therefore, the Signon is likely anonymous under these circumstances.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustInfo>	Aggregate	Optional Profiled requirement	Customer Information Aggregate.
<SecretList>	Aggregate	Optional Repeating	Secret List. This aggregate is for a customer to input the secret required for client enrollment.
<CryptType>	Open Enum	Required Profiled values	Desired encryption type for the password to be sent by the server in <CustAddRs>.
<CustId>	Aggregate	Optional	Customer ID. Usage is actual <CustId> if preassigned, or preference for SPs that allow customer to pick.
<AuthInfo>	Aggregate	Optional	Authentication Information. The client may provide information within this aggregate to send to the server the authentication tokens (such as password; other mechanisms will be defined in future releases of IFX) that may be used in future communications with the server.
<CustPswd>	Aggregate	Optional	Customer Password Aggregate.
</AuthInfo>			

5.4.2.1.2 Response <CustAddRs>

The <CustAddRs> message acknowledges the client's Enroll request. If the SP chooses to deliver a Customer Login ID and a password in the response, they are also included. Otherwise, the SP may provide the Customer Login ID and password through the postal mail, electronic mail, or over the telephone.

Note: *when the <CustAddRs> message is returned in an Audit Response <CustAudRs> or Sync Response <CustSyncRs>, <CryptType> and <PswdDelivery> may not be included, for security reasons.*

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustInfo>	Aggregate	Optional Echoed	Customer Information Aggregate.
<CustId>	Aggregate	Optional Echoed	Customer Login ID.
<CustRec>	Aggregate	Required	Customer Record Aggregate.
<PswdDelivery>	Aggregate	Optional	Password Delivery Aggregate.
<CustPswd>	Aggregate	Required	Customer Password Aggregate.
<ExpDt>	DateTime	Optional	Password Expiration date and time. If absent, the password never expires.
</PswdDelivery>			
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.2.2 Customer Modify Message

A client may use the Customer Modify message to update his or her name, address, or contact details in the SP's records. For information on Conventions for Modification of Server-Based Data, see Section 2.4.4.2.

5.4.2.2.1 Request <CustModRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer ID.
<CustInfo>	Aggregate	Required	Customer Information Aggregate.

5.4.2.2.2 Response <CustModRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer ID.
<CustInfo>	Aggregate	Required Echoed	Customer Information Aggregate.
<CustRec>	Aggregate	Optional	Customer Record Aggregate. Provided if the server modified <CustInfo>
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.2.3 Customer Password Modify Message

A client may use the Modify Customer Password message to change the password. The new Customer Password must follow the password rules established in the <SignonInfo> aggregate, and must be encrypted if the Signon Realm requires it.

5.4.2.3.1 Request <CustPswdModRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer ID.
<CustPswd>	Aggregate	Required	Customer Password Aggregate.

5.4.2.3.2 Response <CustPswdModRs>

The <CustPswdModRs> message provides the client with an acknowledgement that the Customer Password has been changed. If an error causes a client not to receive a response for a Modify Customer Password message, the server is in an indeterminate state and the client should attempt to establish a session with the new Customer Password. If that attempt fails, the client should attempt to establish a session with the old Customer Password.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer ID.
<ExpDt>	DateTime	Optional	Password Expiration date and time. If absent, the password never expires.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.2.4 Customer Status Modify Message

The Customer Status Modify Message may be used to reset the status and counters that were set when a user exceeded the invalid password threshold.

5.4.2.4.1 Request <CustStatusModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the customer whose status is being modified.
<CustStatus>	Aggregate	Required	Customer Status Aggregate.

5.4.2.4.2 Response <CustStatusModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate.
<CustStatus>	Aggregate	Required Echoed	Customer Status Aggregate.
<CustRec>	Aggregate	Required	Customer Record Aggregate.

5.4.2.5 Customer Delete Message

A client may use the Customer Delete message to unenroll.

5.4.2.5.1 Request <CustDelRq>

Tag	Type	Usage	Description

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer ID.
<CascadeDel>	Boolean	Optional	Cascade Delete. If TRUE, server must delete all dependent objects when this object is deleted. If FALSE or omitted, the customer/service link must not be deleted if dependent objects exist. For a service, a dependent object is a service/account link, pending transactions (transfers or payments), recurring models, bills or payees, depending on the service.

5.4.2.5.2 Response <CustDelRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer ID.
<CascadeDel>	Boolean	Optional Echoed	Cascade Delete.
<CustRec>	Aggregate	Required XOR	Customer Record Aggregate
<DependentType>	Open Enum	Required XOR Repeating	Depentent object type that exists for the customer/service link. One element must be returned for each dependent.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.2.6 Customer Inquiry Message

If <UpDt> is equal the server's date of last update, a <Status> code of 1 is returned, and the profile is not returned in the response. This is the newest <CustRec> <UpDt> received by the client. If this information is absent, or does not match the server's date of last update, the server must send all data in the response.

5.4.2.6.1 Request <CustInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Repeating	Customer Identification Aggregate.
<UpDt>	DateTime	Optional	Customer Record Update DateTime. The value is for the last update to <CustInfo> on the server.
<IncToken>	Boolean	Optional	Include Token. If TRUE, a <Token> should be included in the response to set a base for future Service/Account Sync messages. If FALSE or omitted, no <Token> is returned.

5.4.2.6.2 Response <CustInqRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Repeating Echoed	Customer Identification Aggregate.
<UpDt>	DateTime	Optional Echoed	Customer Record Update DateTime.
<IncToken>	Boolean	Optional Echoed	Include Token. If TRUE, a <Token> should be included in the response to set a base for future Service/Account Sync messages. If FALSE or omitted, no <Token> is returned.
<CustRec>	Aggregate	Optional Repeating	Customer Record Aggregate. One record is returned for each customer matching the selection criteria in the request.
<Token>	Identifier	Optional	Token. Server Assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it.

5.4.2.7 Customer Identifier Inquiry Message

A client, typically a CSR, may send a request to the Server to retrieve either the login id corresponding to a permanent id, or the permanent id corresponding to a login id.

5.4.2.7.1 Request <CustIdInqRq>

The client must specify either the Customer Permanent identifier, or the Customer Login identifier.

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustPermId>	C-32	Required OR	Customer Permanent ID. Used as a database key to uniquely identify a Service Provider's customer. Cannot be changed by the customer.
<CustLoginId>	C-32	Required OR	Customer Login ID. Used as a user-friendly name for the customer for authentication purposes. Maps directly to Customer Permanent ID. Some implementations may allow a user to change his or her Login ID.

5.4.2.7.2 Response <CustIdInqRs>

The SP Server responds with a code indicating the processing status and the client identifiers.

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Required	Customer Identification Aggregate.

5.4.2.8 Customer Audit Message

Allows client to audit Customer Add/Modify/Delete messages. When the Cust object changes, the server must generate an Rs message to the Rq that created the pending state.

5.4.2.8.1 Request <CustAudRq>

Tag	*Type*	*Usage*	*Description*
<SPName>	Identifier	Optional	Service Provider Name.
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<SelRangeDt>	DateTime	Optional	Selection Range Date Time Aggregate.

5.4.2.8.2 Response <CustAudRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate		Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if client included <RecCtrlIn> in request and the server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<CustMsgRec>	Aggregate	Optional Repeating	
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the SVCMSGREC request.
<MsgRecDt>	DateTime	Optional	Audit Record Creation Date. Date/time at which the audit record was stored/created by the service provider.
<CustAddRs>	Aggregate	Required XOR	Customer Add Response Record Aggregate. One record is returned for each Activate Account message for this customer, subject to selection criteria or token.
<CustModRs>	Aggregate	Required XOR	Customer Modify Response Record Aggregate. One record is returned for each Modify Account message for this customer, subject to selection criteria or token.
<CustDelRs>	Aggregate	Required XOR	Customer Delete Response Record Aggregate. One record is returned for each Deactivate Account message for this customer, subject to selection criteria or token.
</CustMsgRec>			

5.4.2.9 Customer Sync Message

Allows client to synchronize on Customer Add/Modify/Delete messages. When the Cust object changes, the server must generate an Rs message to the Rq that created the pending state.

5.4.2.9.1 Request <CustSyncRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.

5.4.2.9.2 Response <CustSyncRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if client included <RecCtrlIn> in request and server supports Records Control.
<Token>	Identifier	Required Echoed	Token.
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others it may have received. The token is only significant to the server that originally assigned it.
<CustMsgRec>	Aggregate	Optional Repeating	
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the <CustMsgRec> request.
<MsgRecDt>	DateTime	Optional	Audit Record Creation Date. Date/time at which the audit record was stored/created by the service provider.
<CustAddRs>	Aggregate	Required XOR	Service Add Response Record Aggregate. One record is returned for each Activate Account message for this customer, subject to selection criteria or token.
<CustModRs>	Aggregate	Required XOR	Service Modify Response Record Aggregate. One record is returned for each Modify Account message for this customer, subject to selection criteria or token.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<CustDelRs>	Aggregate	Required XOR	Service Delete Response Record Aggregate. One record is returned for each Deactivate Account message for this customer, subject to selection criteria or token.
</CustMsgRec>			

5.4.3 Customer/Service Link

5.4.3.1 Customer/Service Link Add

5.4.3.1.1 Request <CustSvcAddRq>

A client uses the <CustSvcAddRq> message to request that a service be enabled. If the <SvcName> value is Pay, then this message indicates enabling the Pay service, and should be sent from the CSP to the CPP; likewise, if the <SvcName> is Pres, then it should be sent to a BSP. The fee account should be passed to Service Providers who are actually charging for the service.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustSvcId>	Aggregate	Required	Customer/Service Link Identification Aggregate
<CustSvcInfo>	Aggregate	Required	Customer/Service Link Information Aggregate

5.4.3.1.2 Response <CustSvcAddRs>

The Customer/Service Link Add Response Message acknowledges receipt of the enabling request. It also optionally allows the SP to return disclosures associated with the activation request.

> ***NOTE**: <CustSvcAddRs> may be returned before enabling a service if several steps must be performed in enabling a service. Therefore, a <SvcAcctStatusCode> of* xxx*Pending and the Resolve Pending Date may be returned to the CSP and/or Customer to provide an estimated date for completion of the request. This may be used, for example, when a CPP does a prenotification for a fee account, since it might take several days to obtain results.*

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustSvcId>	Aggregate	Required Echoed	Customer/Service Link Identification Aggregate.
<CustSvcInfo>	Aggregate	Required Echoed	Customer/Service Link Information Aggregate.
<CustSvcRec>	Aggregate	Required	Customer/Service Record Aggregate
<DiscRec>	Aggregate	Optional Repeating	Disclosure Record Aggregate.

Tag	Type	Usage	Description
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.3.2 Customer/Service Link Modify

The <CustSvcModRq> allows a customer to modify specific information about a customer/service link. This capability is currently limited to modifying the fee account.

5.4.3.2.1 Request <CustSvcModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustSvcId>	Aggregate	Required	Customer/Service Link Identification Aggregate.
<CustSvcInfo>	Aggregate	Required	Customer/Service Link Information Aggregate

5.4.3.2.2 Response <CustSvcModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustSvcId>	Aggregate	Required Echoed	Customer/Service Link Identification Aggregate.
<CustSvcInfo>	Aggregate	Required Echoed	Customer/Service Link Information Aggregate.
<CustSvcRec>	Aggregate	Required	Customer/Service Link Record Aggregate
<DiscRec>	Aggregate	Optional Repeating	Disclosure Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.3.3 Customer/Service Link Status Modify

The <CustSvcStatusModRq> allows a Service Provider to temporarily change the customer's service status. For example, this function is used in the Presentment Service to notify a BSP that the CSP is unable to deliver bills or other material to a customer and to notify the BSP that reverse the notification. This may occur when there is a potential security/fraud problem or when a technical problem precludes the customer's access to his/her account with the CSP. The BSP may need to notify the Biller to send paper bills, depending on regulatory or business practices, to ensure the customer's billing is not interrupted.

5.4.3.3.1 Request <CustSvcStatusModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustSvcId>	Aggregate	Required	Customer/Service Link Identification Aggregate
<CustSvcStatus>	Aggregate	Requried	Customer/Service Link Status Aggregate

5.4.3.3.2 Response <CustSvcStatusModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustSvcId>	Aggregate	Required Echoed	Customer/Service Link Identification Aggregate.
<CustSvcStatus>	Aggregate	Requried Echoed	Customer/Service Link Status Aggregate
<CustSvcRec>	Aggregate	Required	Customer/Service Link Record Aggregate
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.3.4 Customer/Service Link Delete

5.4.3.4.1 Request <CustSvcDelRq>

The <CustSvcDelRq> allows a customer to disable a service offered by a specified service provider. A customer may initiate the process by sending the <CustSvcDelRq> to the CSP. The CSP may then send a <CustSvcDelRq> message to a BSP to disable, for example, Bill Presentment Service for a particular customer at the BSP.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Optionally sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustSvcId>	Aggregate	Required	Customer/Service Indentification Aggregate
<CascadeDel>	Boolean	Optional	Cascade Delete. If TRUE, server must delete all dependent objects when this object is deleted. If FALSE or omitted, the customer/service link must not be deleted if dependent objects exist. For a service, a dependent object is a service/account link, pending transactions (transfers or payments), recurring models, bills or payees, depending on the service.

5.4.3.4.2 Response <CustSvcDelRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustSvcId>	Aggregate	Required Echoed	Customer/Service Indentification Aggregate
<CascadeDel>	Boolean	Optional Echoed	Cascade Delete.
<CustSvcRec>	Aggregate	Required XOR	Customer/Service Link Record Aggregate
<DependentType>	Open Enum	Required XOR Repeating	Depentent object type that exists for the customer/service link. One element must be returned for each dependent.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.3.5 Customer/Service Link Audit

Allows client to audit Customer/Service Link Add/Modify/Delete messages associated with the current customer. When the <SvcStatus> changes, the server must generate an Rs message to the Rq that created the pending state. The <Status> <Severity> must always be Info. The <StatusDesc> is used to supply the details of a rejection.

5.4.3.5.1 Request <CustSvcAudRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<SelRangeDt>	DateTime	Optional	Selection Range Date Time Aggregate.
<Method>	Closed Enum	Optional	Customer/Service Link Method. This field is used as a selection criterion.
<SvcName>	Open Enum	Optional	Service Name
<SPName>	Identifier	Optional	Service Provider Name.

5.4.3.5.2 Response <CustSvcAudRs>

Tag	Type	Usage	Description

<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if client included <RecCtrlIn> in request and server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional Echoed	Customer/Service Link Action.
<SvcName>	Open Enum	Optional Echoed	Service Name
<SPName>	Identifier	Optional Echoed	Service Provider Name.
<CustSvcMsgRec>	Aggregate	Optional Repeating	Customer/Service Message Record.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the <CustSvcMsgRec> request.
<MsgRecDt>	DateTime	Optional	Audit Record Creation Date. Date/time at which the audit record was stored/created by the service provider.
<CustSvcAddRs>	Aggregate	Required XOR	Customer/Service Link Add Response Record Aggregate. One record is returned for each Activate Account message for this customer, subject to selection criteria or token.
<CustSvcModRs>	Aggregate	Required XOR	Customer/Service Link Modify Response Record Aggregate. One record is returned for each Modify Account message for this customer, subject to selection criteria or token.
<CustSvcDelRs>	Aggregate	Required XOR	Customer/Service Link Delete Response Record Aggregate. One record is returned for each Deactivate Account message for this customer, subject to selection criteria or token.
</CustSvcMsgRec>			

5.4.3.6 Customer/Service Link Sync

Allows client to synchronize Customer/Service Link Add/Modify/Delete messages associated with the current customer. When the <SvcStatus> changes, the server must generate an Rs message to the Rq that created the pending state. The <Status> <Severity> must always be Info. The <StatusDesc> is used to supply the details of a rejection.

5.4.3.6.1 Request <CustSvcSyncRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.

Tag	Type	Usage	Description
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.

5.4.3.6.2 Response <CustSvcSyncRs>

Tag	Type	Usage	Description
<Status>	Aggregate		Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if client included <RecCtrlIn> in request and server supports Records Control.
<Token>	Identifier	Required Echoed	Token.
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others it may have received. The token is only significant to the server that originally assigned it.
<CustSvcMsgRec>	Aggregate	Optional Repeating	Customer/Service Message Record.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the <CustSvcMsgRec> request.
<MsgRecDt>	DateTime	Optional	Audit Record Creation Date. Date/time at which the audit record was stored/created by the service provider.
<CustSvcAddRs>	Aggregate	Required XOR	Customer/Service Link Add Response Record Aggregate. One record is returned for each Activate Account message for this customer, subject to selection criteria or token.
<CustSvcModRs>	Aggregate	Required XOR	Customer/Service Link Modify Response Record Aggregate. One record is returned for each Modify Account message for this customer, subject to selection criteria or token.
<CustSvcDelRs>	Aggregate	Required XOR	Customer/Service Link Delete Response Record Aggregate. One record is returned for each Deactivate Account message for this customer, subject to selection criteria or token.
</CustSvcMsgRec>			

5.4.4 Service/Account Link

5.4.4.1 Service/Account Link Add Message

5.4.4.1.1 Request <SvcAcctAddRq>

A client uses the <SvcAcctAddRq> message to request that a particular account be activated for an enabled service, e.g., a long distance telephone account for a bill presentment service. Multiple accounts may be activated by sending a separate <SvcAcctAddRq> for each account to be activated.

Bill Presentment:

When activating an account for Bill Presentment Service, the customer and/or CSP must send secrets as requested by the Biller, names and addresses as known to the Biller to be associated with the account, and the customer account with the Biller <BillingAcct>, in order for a BSP or a Biller to correctly identify which customer and account to activate. The Customer name and address at the CSP may be used to assist a BSP or Biller in matching the correct customer with the account to be activated.

Account specific disclosures:

Note that if the Service Provider (e.g., CSP, CPP, or BSP) has account-specific disclosures that were not displayed in the service activation process, the Service Provider must send disclosure text with the <SvcAcctAddRs> when an account is being activated.

If presentation of terms and conditions is required prior to account activation, Disclosure Record Aggregates must be returned in the response. If acceptance is required, and the customer does not accept, the account activation may not be sent to the BSP. If the customer does accept, the client may use <CustDiscStatusModRq>.

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<SvcAcctId>	Aggregate	Required	Service/Account Link Identification Aggregate
<SvcAcctInfo>	Aggregate	Required	Service/Account Link Information Aggregate
<SecretList>	Aggregate	Optional Repeating	Secret List. This is the list of responses to <SecretPrompt> provided for account activation in aggregates such as <BillerRec>.
<CryptType>	Open Enum	Optional Profiled values	Encryption Type. Defined values: None, PKCS#1. Indicates the encryption algorithm used for the secrets within the <SecretList>. Must be one of the types in the Biller directory entry <BillerRec>.
<CustNameAddr>	Aggregate	Optional Repeating	Customer Name/Address aggregate. This may be required for account activation, such as when adding a presentment account.

5.4.4.1.2 Response <SvcAcctAddRs>

The Activate Account Response Message acknowledges receipt of the activate request and contains as much status information as is available.

> ***Note****: When a CSP activates an account on behalf of a customer, the* x*SP may return a status description of PendAdd. Upon subsequent Service/Account Link Audit Requests <SvcAcctAudRq> and Service/Account Link Sync Requests <SvcAcctSyncRq>, the Effective Date may be filled in when the account is activated. The <EffDt> is advisory information coming from the* x*SP.*

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<SvcAcctId>	Aggregate	Required Echoed	Service/Account Link Identification Aggregate.
<SvcAcctInfo>	Aggregate	Required Echoed	Service/Account Link Information Aggregate.
<SvcAcctRec>	Aggregate	Required	Service/Account Record Aggregate.
<DiscRec>	Aggregate	Optional Repeating	Disclosure Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.4.2 Service/Account Link Modify Message

5.4.4.2.1 Request <SvcAcctModRq>

The <SvcAcctModRq> allows a customer to modify specific information about an account. This message is currently limited to modifying (1) a Pay funding account, and (2) the account Nickname.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<SvcName>	Open Enum	Required	Service Name.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<SvcAcctId>	Aggregate	Required	Service/Account Link Identifier.
<SvcAcctInfo>	Aggregate	Required	Service/Account Link Information Aggregate.
<SecretList>	Aggregate	Optional Repeating	Secret List. This is the list of responses to <SecretPrompt> provided for account activation in aggregates such as <BillerRec>.
<CryptType>	Open Enum	Optional Profiled values	Encryption Type. Defined values: None, PKCS#1. Indicates the encryption algorithm used for the secrets within the <SecretList>. Must be one of the types in the Biller directory entry <BillerRec>.
<CustNameAddr>	Aggregate	Optional Repeating	Customer Name/Address aggregate. This may be required for account modification, such as with a presentment account.

5.4.4.2.2 Response <SvcAcctModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status
<RqUID>	UUID	Required Echoed	Request Identifier.

Tag	Type	Usage	Description
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate.
<SvcAcctId>	Aggregate	Required Echoed	Service/Account Link Identification Aggregate.
<SvcAcctInfo>	Aggregate	Required Echoed	Service/Account Link Information Aggregate.
<SvcAcctRec>	Aggregate	Required	Service/Account Record Aggregate.
<DiscRec>	Aggregate	Optional Repeating	Disclosure Record Aggregate.
<CSPRefId>	Identifier	Optional	Financial Institution Reference Number.
<SPRefId>	Identifier	Optional	Service Provider Reference Number.

5.4.4.3 Service/Account Link Delete Message

5.4.4.3.1 Request <SvcAcctDelRq>

The <SvcAcctDelRq> allows a customer or CSP to deactivate a specific account for the specified service. Note that deactivating an account in IFX refers to a service provider removing the linkage between the account and the service, and does not impact the status of the actual account, e.g. suspending an electronic billing account should not be confused with halting the actual service, such as turning off the electricity.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<SvcAcctId>	Aggregate	Required	Service/Account Link Identification Aggregate
<CascadeDel>	Boolean	Optional	Cascade Delete. If TRUE, server must delete all dependent objects when this object is deleted. If FALSE or omitted, the service/account link must not be deleted if dependent objects exist. For a service/account link, a dependent object is pending transactions (transfers or payments), recurring models, bills or payees, depending on the service.

5.4.4.3.2 Response <SvcAcctDelRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<SvcAcctId>	Aggregate	Required Echoed	Service/Account Link Identification Aggregate.
<CascadeDel>	Boolean	Optional Echoed	Cascade Delete.
<SvcAcctRec>	Aggregate	Required XOR	Service/Account Link Record Aggregate

<DependentType>	Open Enum	Required XOR Repeating	Depentent object type that exists for the customer/service link. One element must be returned for each dependent.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.4.4 Service/Account Link Inquiry Message

If <UpDt> is equal the server's date of last update, a <Status> code of 1 is returned, and the profile is not returned in the response. This is the last <NewUpDt> received by the client. If this information is absent, or does not match the server's date of last update, the server must send all data in the response.

5.4.4.4.1 Request <SvcAcctInqRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<SvcName>	Open Enum	Optional Repeating	Service Name. This field identifies the specific set of services for which the server should provide customer profile information. This allows the client to scope the profile request to provide account information only for a specific set of services.
<IncBal>	Boolean	Optional	Include Balances Indicator. If TRUE, the <AcctBal> aggregate within <BankAcctRec> is being requested. If FALSE or omitted, the <AcctBal> aggregate within <BankAcctRec> is not being requested.
<IncToken>	Boolean	Optional	Include Token. If TRUE, a <Token> should be included in the response to set a base for future Service/Account Sync messages. If FALSE or omitted, no <Token> is returned.
<UpDt>	DateTime	Optional	Service Information Update DateTime. The value is for the last update to <CustSvcRec> on the server.

5.4.4.4.2 Response <SvcAcctInqRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<SvcName>	Open Enum	Optional Repeating Echoed	Service Name.
<IncBal>	Boolean	Optional Echoed	Include Balances Indicator. If TRUE, the <AcctBal> aggregate within <BankAcctRec> is being requested. If FALSE or omitted, the <AcctBal> aggregate within <BankAcctRec> is not being requested.

Tag	Type	Usage	Description
<IncToken>	Boolean	Optional Echoed	Include Token. If TRUE, a <Token> should be included in the response to set a base for future Service/Account Sync messages. If FALSE or omitted, no <Token> is returned.
<UpDt>	DateTime	Optional Echoed	Service Information Update DateTime.
<SvcUpDt>	DateTime	Required	New Service/Account Link Update DateTime. The value is for the last update to <CustSvcRec> on the server.
<*xxx*AcctRec>	Aggregate	Optional Repeating	Various Account Record Aggregates, as defined in Section 3.2.5.
<CustSvcRec>	Aggregate	Optional Repeating	Customer/Service Link Record Aggregate. This aggregate contains the status of each service and, if applicable, which accounts are associated with it and their status with respect to the specific service.
<Token>	Identifier	Optional	Token. Server Assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it.

5.4.4.5 Service/Account Link Audit Message

Allows client to audit Service/Account Link Add/Modify/Delete messages associated with the current customer and the current customer's accounts. When the <SvcAcctStatus> changes, the server must generate an Rs message to the Rq that created the pending state. The <Status> <Severity> must always be Info. The <StatusDesc> is used to supply the details of a rejection.

5.4.4.5.1 Request <SvcAcctAudRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<SelRangeDt>	DateTime	Optional	Selection Range Date Time Aggregate.
<Method>	Closed Enum	Optional	Service/Account Link Method. This field is used as a selection criterion.

5.4.4.5.2 Response <SvcAcctAudRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if client included <RecCtrlIn> in request and server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional Echoed	Service/Account Link Action. This field is used as a selection criterion.
<SvcAcctMsgRec>	Aggregate	Optional Repeating	
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the SVCMSGREC request.
<MsgRecDt>	DateTime	Optional	Audit Record Creation Date. Date/time at which the audit record was stored/created by the service provider.
<SvcAcctAddRs>	Aggregate	Required XOR	Service/Account Link Add Response Record Aggregate. One record is returned for each Activate Account message for this customer, subject to selection criteria or token.
<SvcAcctModRs>	Aggregate	Required XOR	Service/Account Link Modify Response Record Aggregate. One record is returned for each Modify Account message for this customer, subject to selection criteria or token.
<SvcAcctDelRs>	Aggregate	Required XOR	Service/Account Link Delete Response Record Aggregate. One record is returned for each Deactivate Account message for this customer, subject to selection criteria or token.
</SvcAcctMsgRec>			

5.4.4.6 Service/Account Link Sync Message

Allows client to synchronize Service/Account Link Add/Modify/Delete messages associated with the current customer and the current customer's accounts. When the <SvcAcctStatus> changes, the server must generate an Rs message to the Rq that created the pending state. The <Status> <Severity> must be Info. The <StatusDesc> is used to supply the details of a rejection.

5.4.4.6.1 Request <SvcAcctSyncRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.

5.4.4.6.2 Response <SvcAcctSyncRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if client included <RecCtrlIn> in request and server supports Records Control.
<Token>	Identifier	Required Echoed	Token.
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others it may have received. The token is only significant to the server that originally assigned it.
<SvcAcctMsgRec>	Aggregate	Optional Repeating	
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the <SvcAcctMsgRec> request.
<MsgRecDt>	DateTime	Optional	Audit Record Creation Date. Date/time at which the audit record was stored/created by the service provider.
<SvcAcctAddRs>	Aggregate	Required XOR	Service/Account Link Add Response Record Aggregate. One record is returned for each Activate Account message for this customer, subject to selection criteria or token.
<SvcAcctModRs>	Aggregate	Required XOR	Service/Account Link Modify Response Record Aggregate. One record is returned for each Modify Account message for this customer, subject to selection criteria or token.
<SvcAcctDelRs>	Aggregate	Required XOR	Service/Account Link Delete Response Record Aggregate. One record is returned for each Deactivate Account message for this customer, subject to selection criteria or token.
</SvcAcctMsgRec>			

5.4.5 Disclosure

5.4.5.1 Customer/Disclosure Link Status Modify Message

A customer uses the Customer/Disclosure Link Status Modify Message to accept or reject a disclosure received via various messages.

5.4.5.1.1 Request <CustDiscStatusModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.

Tag	Type	Usage	Description
<DiscId>	UUID	Required	Disclosure Identifier.
<CustDiscStatusCode>	Closed enum	Required	Customer/Disclosure Link Status Code Defined Values: Accept—The customer accepted the terms of the disclosure Reject—The customer rejected the terms of the disclosure Viewed—The customer viewed the disclosure, but neither explicitly accepted nor explicitly rejected it.

5.4.5.1.2 Response <CustDiscStatusModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DiscId>	UUID	Required Echoed	Disclosure Identifier.
<CustDiscStatusCode>	Closed enum	Required Echoed	Customer/Disclosure Link Status Code Defined Values: Accept—The customer accepted the terms of the disclosure Reject—The customer rejected the terms of the disclosure Viewed—The customer viewed the disclosure, but neither explicitly accepted nor explicitly rejected it.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

5.4.5.2 Customer/Disclosure Link Inquiry Message

A customer or CSP may make a query about disclosure terms and conditions for a given account, service, or biller.

5.4.5.2.1 Request <CustDiscInqRq>

Tag	Type	Usage	Description
<SvcName>	Open Enum	Required	Service Name.
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DiscId>	UUID	Optional	Disclosure ID. This element is used as a selection criterion.
<CustDiscStatusCode>	Closed enum	Optional	Customer/Disclosure Link Status Code. This element is used as a selection criterion

Tag	Type	Usage	Description
<IncDisc>	Boolean	Optional	Include Disclosure Boolean. If TRUE, the client is requesting that the disclosure be included with each record in the response. If FALSE or omitted, the client is requesting that the disclosure *not* be included with each record in the response.

5.4.5.2.2 Response <CustDiscInqRs>

The SP Server responds with disclosure details.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DiscId>	UUID	Optional Echoed	Disclosure ID. This element is used as a selection criterion.
<CustDiscStatusCode>	Closed enum	Optional Echoed	Customer/Disclosure Link Status Code. This element is used as a selection criterion
<IncDisc>	Boolean	Optional Echoed	Include Disclosure Boolean. If TRUE, the client is requesting that the disclosure be included with each record in the response. If FALSE or omitted, the client is requesting that the disclosure *not* be included with each record in the response.
<CustDiscRec>	Aggregate	Optional XOR Repeating	Customer/Disclosure Record Aggregate. One aggregate is returned for each record matching the selection criteria in the request, if <IncDesc> is TRUE.
<CustDiscStatus>	Aggregate	Optional XOR Repeating	Customer/Disclosure Status Aggregate. One aggregate is returned for each record matching the selection criteria in the request, if <IncDesc> is FALSE or omitted.

5.5 Base Service Profile <BaseSvcProfInfo>

The Base Service Profile functions the same way as all other services; therefore, it contains a Profile description for that service.

Profiles for other services may be found at the end of each chapter of this document. A SP must return a complete set of Profiles for the services that it supports in response to a customer-initiated Service Profile Inquiry <SvcProfInqRq>.

The Base Service Profile is defined below. This information is returned to the client in the Service Profile Inquiry message and provides information on how the client should use the Base Service Profile.

Tag	Type	Usage	Description
<SvcCore>	Aggregate	Required	Service Core Aggregate. Information specified for every service.

Tag	*Type*	*Usage*	*Description*
<MsgSupt>	Open Enum	Required Repeating	Supported Messages. This is a list of messages that are supported for this Service Profile. The convention is to use the name of the message without the Rq or Rs so that each message is only listed once. Valid values: SvcProf, HolInq, CustAdd, CustMod, CustPswdMod, CustStatusMod, CustDel, CustInq, CustIdInq, CustAud, CustSync, CustSvcAdd, CustSvcMod, CustSvcDel, CustSvcInq, CustSvcAud, CustSvcSync, SvcAcctAdd, SvcAcctMod, SvcAcctDel, SvcAcctInq, SvcAcctAud, SvcAcctSync, CustDiscStatusMod, CustDiscInq.
<OptSupt>	Open Enum	Optional Repeating	Options Supported. Defined values: RecCtrl
<SignonInfo>		Required	
<AuthSupt>	Open Enum	Optional Repeating	Authentication methods supported. Defined values: Password, EmbedCert, TranspCert, MagPIN, SessKey
<CryptType>	Open Enum	Required Repeating	Encryption Type. Provides a list of supported encryption types. Authentication messages must use one of the types supported here.
<ChgPswdFirst>	Boolean	Optional	Password Modification Required First. If set to TRUE, indicates server requires customer to perform Modify Password message prior to executing any messages other than signon in the first session. *The following elements provide options of allowable password characters. They are provided to allow the client to validate a customer-entered password during password creation.*
<Min>	Long	Optional	Minimum Number of Password Characters.
<Max>	Long	Optional	Maximum Number of Password Characters.
<CharType>	Open Enum	Optional	Character Type Code.
<CaseSen>	Boolean	Optional	Case Sensitive Flag. If set to TRUE, indicates that the password is case-sensitive.
<SpaceAlwd>	Boolean	Optional	Spaces Allowed. If set to TRUE, indicates that the password allows spaces.
<Memo>	C-255	Optional	Memo Text. The text to be displayed to the user to indicate password-formatting rules.
</SignonInfo>			
<EnrollProf>	Aggregate	Optional	Enroll Profile Aggregate.
<WebEnrollURL>	URL	Optional	Web URL to start Enrollment Process in lieu of using the <CustAddRq> message.
<EnrollDesc>	C-255	Optional	Enrollment Description. Instructions to customer for other enrollment process. For example, the customer services telephone number.
<CryptType>	Open Enum	Optional *but see Description* Repeating	Encryption Type. Indicates which encryption types are supported for the secret list in the Enroll message. Valid values: NONE, PKCS#1. Required if IFX enrollment is supported.
<SecretPrompt>	Aggregate	Optional Repeating	Secret Prompt. If omitted, the SP does not require the customer to enter any secrets for client enrollment.

Tag	*Type*	*Usage*	*Description*
<CustNameReqd>	Boolean	Optional	Customer Name Required Indicator. If set to TRUE, SP requires that <CustName> be included in <CustAddRq>.
<PostAddrReqd>	Boolean	Optional	Customer Postal Address Required Indicator. If set to TRUE, SP requires that <PostAddr> be included in <CustAddRq>.
</EnrollProf>			

6 The Banking Service <BankSvc>

The IFX Specification provides core-banking capabilities through the Banking Service. The Banking Service includes functions such as Statements and Account Inquiries, Transfers, Recurring Transfers, Customer Communications, and Bank Mail. These banking functions may be applied to either deposit accounts, loan accounts, or credit card accounts.

6.1 Description

The Banking Service allows clients to perform the following functions:

- Statements and Account Inquiries, including Account Balances, Account Detail Inquiry, Account Closing Statement, Account History, and Interest Rate, Account Taxation and Exchange Rate Inquiries.
- Add, Modify and Delete a Stop Check
- Add, Modify, and Delete and Interbank Transfers (may be immediate or scheduled), and request Line of Credit Advance and loan repayment
- Add, Modify and Delete Recurring and Interbank Transfer Models for scheduled recurring transfers
- Customer Communications such as order a copy of a statement, order check copies, reorder checkbooks, open a term deposit account, order a deposit book, report a lost card, request a copy of a credit card sales slip, report a credit card dispute, report a lost credit card, change the credit limit on a credit card, and send and retrieve banking Email.

6.1.1 Accounts

The Banking Service specifies accounts using the Account Identification aggregates <DepAcctId>, <LoanAcctId>, and <CardAcctId>

6.1.2 Client-Initiated Actions

6.1.2.1 Account Inquiries

The Banking Service provides messages that allow a client to retrieve detailed information about bank accounts, such as balances or previously initiated transactions.

6.1.2.2 Bank Statement View

The Banking Service provides messages that allow a client to retrieve Bank Account Statements for one or more given periods, comparable to traditional paper statements. The Statement includes various balances, and optionally the transaction detail for the statement period.

6.1.2.3 Stop Checks

The Banking Service provides messages that allow a client to stop payment on a check. Messages are also provided to inquire about stopped checks and to play back messages associated with stopped checks.

6.1.2.4 Reorder Checks

A customer may use the Reorder Checks message to request that an order of checks be sent. The customer may specify the total quantity of checks to be sent, starting number of the checks, and style of the checks.

6.1.2.5 Order Deposit Books

A customer may use the Deposit Book Order message to request that deposit book be sent.

6.1.2.6 Retrieve Interest Rates

A customer may request quotes on interest rates for a variety of financial products using the Interest Rate Inquiry message. The customer enters the Account Type, Tax Status, Amount, and Term, and the message returns the current interest rates.

6.1.2.7 Add, Modify, and Delete Transfers

A customer may add, modify, or delete a non-recurring funds transfer.

6.1.2.8 Add, Modify and Delete Recurring Transfer Models

Messages are also provided in the Banking Service to allow a client to manage models for fixed amount recurring funds transfers. The customer enters parameters, such as the transfer amount, the date of the first transfer, the frequency of the transfers, and the term of the model, and the Financial Institution automatically generates transfers based on this model at the requested frequency for the model term. Recurring transfer models may be closed-ended or open-ended (if no term is provided).

6.1.2.9 Credit Card Messages

The Banking Service provides customer service messages related to credit card accounts, including client request for a sales slip copy or report of a credit card dispute.

6.2 Banking Service Message Summary

Function / Message Name	*Required*	*Comments*
Balance Inquiry **<BalInqRq>** **<BalInqRs>**	Yes	Allows client to retrieve appropriate balances as of the time the message is executed based on the type of Banking Account.
Account Inquiry **<AcctInqRq>** **<AcctInqRs>**		Allows client to retrieve current information about the Account that varies by account type. This inquiry includes balances, but does not include message detail.
Deposit Account Statement Inquiry **<DepAcctStmtInqRq>** **<DepAcctStmtInqRs>**		Allows client to retrieve a Deposit Account Closing Statement for one or more given periods, comparable to a traditional paper statement. The Closing Statement includes Opening, Closing, and Minimum Ledger Balances and optionally the messages detail for the statement period
Credit Card Statement Closing **<CCAcctStmtInqRq>** **<CCAcctStmtInqRs>**		Allows client to retrieve a Credit Card Account Closing Statement for one or more given periods, comparable to a traditional paper statement. The Closing Statement includes Opening and Closing Credit Card Balances, as well as Date Due and a number of summary level amounts and optionally the messages detail for the statement period
Deposit Account Transaction Inquiry **<DepAcctTrnInqRq>** **<DepAcctTrnInqRs>**		Allows client to retrieve the financial messages posted against an account over a specified time period.
Card Account Transaction Inquiry **<CardAcctTrnInqRq>** **<CardAcctTrnInqRs>**		Allows a client to retrieve the financial messages posted against an account over a specified time period.
Interest Rate Inquiry **<IntRateInqRq>** **<IntRateInqRs>**		Allows client to retrieve a list of financial products and their associated interest rates.

Function / Message Name	*Required*	*Comments*
Bank Account Taxation Inquiry **<BankAcctTaxInqRq>** **<BankAcctTaxInqRs>**		Allows client to request details of taxation on a specific bank account.
Foreign Exchange Rate Inquiry **<ForExRateInqRq>** **<ForExRateInqRs>**		Allows client to inquire about an exchange rate from a Financial Institution, and to optionally request an exchange rate commitment
Stop Check Add **<StopChkAddRq>** **<StopChkAddRs>**		Allows a client to stop a check or a range of checks.
Stop Check Cancel **<StopChkCanRq>** **<StopChkCanRs>**		Allows client to cancel a previous Stop Check request with a Financial Institution.
Stop Check Inquiry **<StopChkInqRq>** **<StopChkInqRs>**		Allows client to view current Stopped Check records.
Stop Check Audit **<StopChkAudRq>** **<StopChkAudRs>**		Allows client to play back Stopped Check messages associated with the current customer since some past point in time.
Stop Check Synchronization **<StopChkSyncRq>** **<StopChkSyncRs>**		Allows client to play back Stopped Check messages associated with the current customer since some past point in time.
Transfer Add **<XferAddRq>** **<XferAddRs>**		Allows client to schedule an Transfer. Includes immediate transfer.
Transfer Modify **<XferModRq>** **<XferModRs>**		Allows client to modify a pending Transfer.
Transfer Cancel **<XferCanRq>** **<XferCanRs>**		Allows client to cancel a pending Transfer.
Transfer Inquiry **<XferInqRq>** **<XferInqRs>**		Allows client to view Transfer records.
Transfer Audit **<XferAudRq>** **<XferAudRs>**		Allows client to play back the Transfer messages associated with the current customer since some past point in time.
Transfer Synchronization **<XferSyncRq>** **<XferSyncRs>**		Allows client to play back the Transfer messages associated with the current customer since some past point in time.
Recurring Transfer Model Add **<RecXferAddRq>** **<RecXferAddRs>**		Allows client to create a Recurring Transfer Model.

Function / Message Name	Required	Comments
Recurring Transfer Model Modify **<RecXferModRq>** **<RecXferModRs>**		Allows client to modify an existing Recurring Transfer Model.
Recurring Transfer Model Cancel **<RecXferCanRq>** **<RecXferCanRs>**		Allows client to cancel an existing Recurring Transfer Model.
Recurring Transfer Model Inquiry **<RecXferInqRq>** **<RecXferInqRs>**		Allows client to retrieve current Recurring Transfer Model records.
Recurring Transfer Model Audit **<RecXferAudRq>** **<RecXferAudRs>**		Allows client to play back the messages associated with the model itself, as opposed to messages that are created by it, for the current customer since some past point in time.
Recurring Transfer Model Synchronization **<RecXferSyncRq>** **<RecXferSyncRs>**		Allows client to play back the messages associated with the model itself, as opposed to messages that are created by it, for the current customer since some past point in time.
Check Order Add **<ChkOrdAddRq>** **<ChkOrdAddRs>**		Allows client to reorder checks.
Deposit Book Order Add **<DepBookOrderAddRq>** **<DepBookOrderAddRs>**		Allows client to order a Deposit Book from a Financial Institution.

6.3 Banking Service Common Aggregates

6.3.1 Deposit Account Record Aggregate <DepAcctRec>

Tag	Type	Usage	Description
<StopCount>	Long	Optional	Number of Stopped Payments.
<HoldCount>	Long	Optional	Number of Holds outstanding.
<LastDepDt>	Date	Optional	Last Deposit Date.
<LastDepCurAmt>	Currency Amount	Optional	Last Deposit Amount.
<DepMatureDt>	Date	Optional	Deposit Maturity Date.

6.3.2 Credit Card Account Record Aggregate <CCAcctRec>

Tag	Type	Usage	Description
<DueDt>	Date	Optional	Payment Due Date.

Tag	Type	Usage	Description
<ExpDt>	DateTime	Optional	Expiration date for card. If the card has only month and year expiration, the *last* day of the month must be specified here.
<LastPmtDt>	Date	Optional	Last Payment Date.
<LastPmtCurAmt>	Currency Amount	Optional	Last Payment Amount.

6.3.3 Common Loan and LOC Account Aggregate <LoanInfoCommon>

This aggregate contains information that is common to both the <LoanInfo> and <LocInfo> aggregates.

Tag	Type	Usage	Description
<OrigDt>	Date	Optional	Original Date.
<Freq>	Open Enum.	Optional	Frequency. Repayment Frequency.
<PurposeDesc>	C-80	Optional	Purpose Description—Purpose for Loan or LOC
<CollateralDesc>	C-80	Optional	Collateral Description.
<CompletedPmtCount>	Long	Optional	Number of Payments Completed.
<RegPmtCurAmt>	Currency Amount	Optional	Regular Payment Amount
<PmtDueDt>	Date	Optional	Date Next Payment Due.
<NextPmtCurAmt>	Currency Amount	Optional	Next Payment Amount, including any past due amounts that are now due. A first or last payment may also differ from the regular payment amount, even if there is no past due amount.
<LastPmtDt>	Date	Optional	Last Payment Date.
<LastPmtCurAmt>	Currency Amount	Optional	Last Payment Amount.
<PastDuePmtCount>	Long	Optional	Number of Past Due Payments.

6.3.4 Loan Account Record Aggregate <LoanAcctRec>

Tag	Type	Usage	Description
<Term>	Aggregate	Optional	Term Aggregate.
<MatDt>	Date	Optional	Loan Maturity Date.
<LoanInfoCommon>	Aggregate	Optional	Loan and LOC common information aggregate

6.3.5 Line of Credit Account Record Aggregate <LOCAcctRec>

Tag	Type	Usage	Description
<LocLimit>	Currency Amount	Optional	LOC Limit—The maximum authorized amount for the line of credit.
<MinPmtCurAmt>	Currency Amount	Optional	Minimum Payment Amount.

Tag	Type	Usage	Description
<ExpDt>	DateTime	Optional	Expiration Date. If omitted, the LOC is open ended.
<LoanInfoCommon>	Aggregate	Optional	Loan and LOC common information aggregate

6.3.6 Mortgage Account Record Aggregate <MortAcctRec>

Tag	Type	Usage	Description
<OrigDt>	Date	Optional	Origination Date.
<LoanInfoCommon>	Aggregate	Optional	Loan Information.
<LastPmtDt>	Date	Optional	Last Payment Date.
<TaxPaidDt>	Date	Optional	Tax Paid Date.
<TaxPaidCurAmt>	Currency Amount	Optional	Tax Paid Amount.
<LastYrTaxPaidDt>	Date	Optional	Last Year Tax Paid Date.
<LastYrTaxPaidCurAmt>	Currency Amount	Optional	Last Year Tax Paid Amount.
<InsPaidDt>	Date	Optional	Mortgage Insurance Paid Date.
<InsPaidCurAmt>	Currency Amount	Optional	Mortgage Insurance Paid Amount.
<LastYrInsPaidDt>	Date	Optional	Mortgage Last Year Insurance Paid Date.
<LastYrInsPaidCurAmt>	Currency Amount	Optional	Mortgage Last Year Insurance Paid Amount.

6.3.7 Bank Account Transaction Record Aggregate <BankAcctTrnRec>

The <BankAcctTrnRec> aggregate describes the data common to all banking account transaction types.

Tag	Type	Usage	Description
<TrnType>	Open Enum	Optional	Transaction Type. See Data Dictionary for details.
<TrnSrc>	Open Enum	Optional	Transaction Source. See Data Dictionary for details.
<PostedDt>	Date	Required	Posted Date. For banking, the date on which the transaction was recorded against the account. For transfers, the date on which the entries were made on the books of the receiving Financial Institution.
<OrigDt>	Date	Optional	Customer Origination Date. The date on which the customer originated the transaction.
<CurAmt>	Currency Amount	Required	Currency Amount. Always in the currency of the account.
<OrigCurAmt>	Currency Amount	Optional	Original Currency Amount. Indicates the transaction was initiated in a currency other than the default for the account (funding account in the case of transfers or payments). The currency rate, if provided, must be included here.
<IndustId>	Aggregate	Optional	Standard Industrial ID Aggregate.
<Memo>	C-255	Optional	Memo. Additional information about the transaction.
<Name>	C-40	Optional XOR	Name of payee or merchant. Either <Name> or <CustPayeeInfo>, but not both.

Tag	Type	Usage	Description
<CustPayeeInfo>	Aggregate	Optional XOR	Payee Information Aggregate. Either <Name> or <CustPayeeInfo>, but not both.
<CSPRefId>	Identifier	Optional	CSP Reference Identifier. Used for reconciliation.
<SPRefId>	Identifier		Service Provider Message Identifier.
<SPRefIdCorrect>	Aggregate	Optional	SP Reference Identifier Correction. Either replaces or deletes a previous message depending on the value of <CorrectAction>.
<SPRefId>	Identifier	Required	Old Message Identifier. This is the <SPRefId> of the message that is being replaced or deleted.
<CorrectAction>	Closed Enum	Required	Correction Action.
</SPRefIdCorrect>			

6.3.8 Deposit Account Transaction Record Aggregate <DepAcctTrnRec>

The <DepAcctTrnRec> aggregate describes a single transaction posted against a Deposit Account. It is used in messages that provide transaction detail.

The sign convention for the statement message aggregate is as follows: a positive <CurAmt> is added to the account balance and a negative <CurAmt> is subtracted from the account balance

Sign of <CurAmt>	Meaning
Positive	Increase customer asset.
Negative	Decrease customer asset.

Tag	Type	Usage	Description
<BankAcctTrnRec>	Aggregate	Required	Banking Transaction Record.
<AvailDt>	Date	Optional	Available Date. The date on which funds are available.
<ChkNum>	NC-12	Optional	Check Number.
<XferId>	Identifier	Optional	Transfer Identifier. Assigned by the server at the time the Transfer is first added. Cannot be changed by the client.
<XferId>	Identifier	Optional	Funds Transfer Identifier. Assigned by the server at the time the Funds Transfer is first added. Cannot be changed by the client.
<PmtId>	Identifier	Optional	Payment Identifier. Assigned by the server at the time the Payment is first added. Cannot be modified by the client.
<DepAcctIdTo>	Aggregate	Optional XOR	Deposit Account Detail. Used for transfer messages.
<CardAcctIdTo>	Aggregate	Optional XOR	Credit Account Detail. Used for transfer messages.
<LoanAcctIdTo>	Aggregate	Optional XOR	Loan Account Detail. Used for transfer messages.
<ATMTrnInfo>	Aggregate	Optional	ATM (Automated Teller Machine) Message Information.
<ATMOwner>	C-14	Optional	The name of the individual or organization which owns the ATM.
<ATMLocation>	C-18	Optional	Alphanumeric string assigned by the ATM owner that specifies the ATM location.

Tag	Type	Usage	Description
<ATMCityState>	C-15	Optional	Alphanumeric string assigned by the ATM owner that specifies the ATM city and state or province.
</ATMTrnInfo>			
<USA.ACHTrnInfo>	Aggregate	Optional	ACH Message Information Aggregate.
<OriginatorName>	C-40	Required	Name of originator of the ACH Message.
<USA.RTN>	NC-9	Optional	Transit Routing Number. A number uniquely identifying an organization providing products and services of a monetary or financial nature within the United States.
</USA.ACHTrnInfo>			

6.3.9 Credit Card Account Transaction Record Aggregate <CCAcctTrnRec>

The sign convention for the statement transaction aggregate is as follows: a positive <CurAmt> is added to the account balance and a negative <CurAmt> is subtracted from the account balance.

Sign of <CurAmt>	Meaning
Positive	Increase customer liability
Negative	Decrease customer liability

Tag	Type	Usage	Description
<BankAcctTrnRec>	Aggregate	Required	Banking Transaction Record.
<SalesSlipRefNum>	NC-23	Required	Sales Slip Reference Number.
<Memo>	C-255	Optional	Memo. Additional information about the message.

6.3.10 Selection Range Check Number Aggregate <ChkRange>

Tag	Type	Usage	Description
<ChkNumStart>	NC-12	Required OR	Selection Low Check Number.
<ChkNumEnd>	NC-12	Required OR	Selection High Check Number.

6.3.11 Stop Check Record Aggregate <StopChkRec>

The <StopChkRec> aggregate is generally used in response messages related to Stopped Checks.

Tag	Type	Usage	Description
<StopChkInfo>	Aggregate	Required	Stop Check Information.
<PostedDt>	Date	Optional	Check Posting Date. The date on which the check is posted against the customer's account.
<StopChkStatusCode>	Closed Enum	Required	Stop Check Status Code. The status of a Stop Check request that is returned as part of a Stop Check inquiry.
<StatusDesc>	C-255	Optional	Check Error Description.

6.3.11.1 Stop Check Information Aggregate <StopChkInfo>

The <StopChkInfo> aggregate is used in messages related to Stop Checks. It is generally used in the request messages.

Tag	*Type*	*Usage*	*Description*
<ChkNum>	NC-12	Required	Check Number.
<Name>	C-40	Optional	Check Pay-To Name.
<CurAmt>	Currency Amount	Optional	Check Currency Amount.
<OrigDt>	Date	Optional	Check Origination Date. The date on which the customer originates the check.

6.3.12 Foreign Exchange Rate Information Aggregate <ForExRateInfo>

A customer may request a foreign exchange rate commitment from a Financial Institution using the Foreign Exchange Rate Message. This commitment may be stored by the client and may be referenced by an Intra- or Interbank transfer until the expiration date/time.

Tag	*Type*	*Usage*	*Description*
<CurAmt>	Currency Amount	Optional *but see Description*	Message Amount. This is the amount in the TO currency at the quoted exchange rate. Must be included when the Financial Institution quotes the commitment in the Foreign Exchange Response Message. Included by the client in a subsequent transfer request only if the transfer amount = the commitment amount.
<CurRate>	Decimal	Required	Exchange Rate.
<CurConvertRule>	Closed Enum	Required	Currency Conversion Indicator. Valid values are Direct and Indirect. See Section 2.3.8 for more information.
<FeeCurAmt>	Currency Amount	Optional	Fee. The fee required to execute the exchange may be provided to the customer for informational purposes.
<ExpDt>	DateTime	Optional *but see Description*	Expiration Date/Time. Required if it is a committed rate. Explicit commitment expiration date and time. After the date/time specified in this field, the Financial Institution is under no obligation to honor the rate returned.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.3.13 Transfer Record Aggregate <XferRec>

Tag	*Type*	*Usage*	*Description*
<XferId>	Identifier	Required	Transfer Identifier. Assigned by the server at the time the Transfer is first added.

Tag	Type	Usage	Description
<RecXferId>	Identifier	Optional	Recurring Transfer Model Identifier. Assigned by the server at the time the Recurring Transfer Model is first added. Cannot be changed by the client. Included only if the Banking provider generated the transfer instance from a recurring model.
<RecXferMod>	Boolean	Optional *but see Description*	Recurring Transfer Modified Indicator. Required if <XferModRq> has subsequently modified a transfer generated from a Recurring Transfer Model so it no longer matches the Recurring Transfer Model. This may be supplied only for recurring transfer instances.
<XferInfo>	Aggregate	Required	Transfer Information Aggregate.
<XferStatus>	Aggregate	Required	Transfer Status.

6.3.13.1 Recurring Transfer Record Aggregate <RecXferRec>

Tag	Type	Usage	Description
<RecXferId>	Identifier	Required	Recurring Transfer Model Identifier. Assigned by the server at the time the Recurring Transfer Model is first added. Cannot be changed by the client.
<XferInfo>	Aggregate	Required	Transfer Information Aggregate.
<RecModelInfo>	Aggregate	Required	Recurring Model Information Aggregate.
<RemainingInsts>	Long	Required	Remaining Instance Count. The server must calculate this number as the number of actual payments to be made plus the number of instances to skip based on the customer-entered <RecSeriesEnd>. Server must calculate on <RecXferAddRq> and return in response. Server must recalculate in case of an <RecXferModRq> that changes <RecSeriesEnd> or when a payment is spawned.

6.3.13.2 Transfer Information Aggregate <XferInfo>

The <XferInfo> aggregate is used in messages related to Transfers, Interbank Transfers, and Recurring Transfer Models for Funds Transfers.

Transfers may be single currency transfers, (i.e. the source and target amount is the same) or may be foreign exchange transfer (i.e. source and target amounts have different ISO currency code). The Financial Institution indicates support of foreign exchange transfers by specifying ForEx within <OptSupt> within <XferProf> in the Banking Profile

A customer normally cannot specify the exchange rate for a foreign exchange transfer. Typically, the Financial Institution makes the transfer at the prevailing exchange rate at the time of the transfer, often considering such factors as the amount of transfer and the customer relationship with the Financial Institution, in addition to the interbank exchange rate. Some Financial Institutions make exchange rate commitments (see Exchange Rate message). Typically, these commitments expire after a short period of time. In this case, the customer may specify the exchange rate and reference the <CSPRefId> returned within the Exchange Rate Rs.

Tag	Type	Usage	Description
<DepAcctIdFrom>	Aggregate	Required XOR	Deposit Account Identification Aggregate. Usage is transfer source account.

Tag	Type	Usage	Description
<CardAcctIdFrom>	Aggregate	Required XOR	Credit Account Identification Aggregate. Usage is transfer source account.
<LoanAcctIdFrom>	Aggregate	Required XOR	Loan Account Identification Aggregate. Usage is transfer source account.
<DepAcctIdTo>	Aggregate	Required XOR	Deposit Account Identification Aggregate. Usage is transfer destination account.
<CardAcctIdTo>	Aggregate	Required XOR	Credit Account Identification Aggregate. Usage is transfer destination account.
<LoanAcctIdTo>	Aggregate	Required XOR	Loan Account Identification Aggregate. Usage is transfer destination account.
<CurAmt>	Currency Amount	Required	Currency Amount.
<DueDt>	Date	Optional	Due Date. If not specified, the transfer is to be scheduled as soon as possible.
<Category>	C-40	Optional	Category for this message, modified by the client.
<ImmediateXfer>	Boolean	Optional Profiled support	Immediate Transfer Indicator. If set to TRUE, the transfer should be executed immediately, and not at end of day. Subject to support in Service Profile.
<ForExInfo>	Aggregate	Optional	Foreign Exchange Information Aggregate.

6.3.13.3 Transfer Status Aggregate <XferStatus>

The <XferStatus> is returned in responses to Add or Modify Transfer and Add or Modify Recurring Transfer Model. Note that all elements within this aggregate are assigned by the server and cannot be assigned or modified by the client.

Tag	Type	Usage	Description
<XferStatusCode>	Closed Enum	Required	Transfer Status Code. This identifies the payment processing status.
<EffDt>	Date	Required	Transfer Status Date. The date associated with the state change to the current state.
<StatusModBy>	Open Enum	Optional	

6.3.14 Check Order Record Aggregate

Tag	Type	Usage	Description
<ChkOrdId>	Identifier	Required	Check Order Identifier. Assigned by the server at the time the Check Order is first added.
<ChkOrdInfo>	Aggregate	Required	Check Order Information Aggregate.
<ChkOrdStatus>	Aggregate	Required	Check Order Status.

6.3.14.1 Check Order Information Aggregate <ChkOrdInfo>

Tag	Type	Usage	Description
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<ChkPrint>	Aggregate	Optional	Check Print Aggregate.
<CustName>	Aggregate	Required	Customer Name Aggregate.
<PostAddr>	Aggregate	Required	Customer Address Aggregate.
<Phone>	Phone Number	Optional	Telephone Number.
<Desc>	C-80	Optional	Additional Information to be printer on check.
</ChkPrint>			
<ShipTo>	Aggregate	Optional	Ship To Aggregate.
<CustName>	Aggregate	Required	Customer Name Aggregate.
<PostAddr>	Aggregate	Required	Customer Address Aggregate.
</ShipTo>			
<Count>	Long	Optional	Number of Checks. If this is missing, the number used for the previous order must be used.
<ChkNumStart>	NC-12	Optional	Starting Check Number of the Order.
<ChkBkStyleId>	Identifier	Optional	Checkbook Style Type. Valid values assigned by Financial Institution. This field indicates customer's choice.
<DeliveryMethod>	Open Enum	Optional	Delivery Method. Default is POST. Value must be supported in Service Profile.

6.3.14.2 Check Order Status Aggregate <ChkOrdStatus>

Tag	Type	Usage	Description
<ChkOrdStatusCode>	Closed Enum	Required	Check Order Status Code.
<EffDt>	Date	Required	Check Order Status Date. The date associated with the state change to the current state.
<StatusModBy>	Open Enum	Optional	

6.3.15 Deposit Book Order Record Aggregate <DebBkOrdRec>

Tag	Type	Usage	Description
<DepBkOrdId>	Identifier	Required	Deposit Book Order Identifier. Assigned by the server at the time the Deposit Book Order is first added.
<DepBkOrdInfo>	Aggregate	Required	Deposit Book Order Information Aggregate.
<DepBkOrdStatus>	Aggregate	Required	Deposit Book Order Status.

6.3.15.1 Deposit Book Order Information Aggregate <DepBkOrdInfo>

Tag	Type	Usage	Description
<DepAcctId>	Aggregate	Required	Deposit Account Identification Aggregate.
<Count>	Long	Optional	Number of Deposit Slips. If omitted, the number used for the previous order must be used.
<DepBkStyleId>	Identifier	Optional	Style of Deposit Book.
<DeliveryMethod>	Open Enum	Optional Profiled values	Delivery Method. Used to request the delivery channel for requested information. See Data Dictionary for details. Default is Post. Value selected must be supported in Service profile.

6.3.15.2 Deposit Book Order Status Aggregate <DepBkOrdStatus>

Tag	Type	Usage	Description
<DepBkOrdStatusCode>	Closed Enum	Required	Deposit Book Order Status Code.
<EffDt>	Date	Required	Deposit Book Order Status Date. The date associated with the state change to the current state.
<StatusModBy>	Open Enum	Optional	

6.3.16 Transfer Profile Aggregate <XferProf>

The Transfer Profile Aggregate is used to convey information on the transfer.

Tag	Type	Usage	Description
<OptSupt>	Open Enum	Optional Repeating	Options Supported. Valid values: ForEx, ImmediateXfer, RecCtrl, RecFinalAmt, RecInitialAmt, RecModelNickname, Skip, SchedXfer.
<PrcSched>	Aggregate	Optional	Processing Schedule Aggregate. If omitted, the default processing schedule is assumed.
<RecXferProf>	Aggregate	Optional	Recurring Transfer Profile Aggregate.
<Freq>	Open Enum	Required Repeating	Recurring Model Frequency. Usage is a list of supported frequencies. See Data Dictionary for details.
<ModPendingType>	Closed Enum	Required	Client Modify Pending Type. If set to Always, changes to recurring transfer models are always propagated to pending transfers based on that model. If set to Never, changes to recurring transfer models are never propagated to pending transfers based on that model. If set to IfRequested, then changes to recurring transfer models are (1) propagated to pending transfers based on that model when <ModPending> is set to TRUE in the Modify Recurring Transfer Model request <RecXferModRq> or <RecInterModRq>) and (2) not propagated when <ModPending> is set to FALSE or not specified.
</RecXferProf>			

6.4 Statement and Account Inquiry

6.4.1 Balance Inquiry

A client may use the <BalInqRq> message to retrieve account balances without requesting message detail or a statement.

6.4.1.1 Request <BalInqRq>

The client specifies only the account for which to retrieve balances.

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<IncExtBal>	Boolean	Optional	Include Extended Balances Indicator. If TRUE, the response should also include the <ExtAcctBal> aggregate and return all available balances for the type of account. If FALSE or omitted, the response should only include the standard balances for the account in <AcctBal>.

6.4.1.2 Response <BalInqRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR Echoed	Loan Account Identification Aggregate.
<IncExtBal>	Boolean	Optional Echoed	Include Extended Balances Indicator.
<AcctBal>	Aggregate	Required Repeating	Account Balance Aggregate.
<ExtAcctBal>	Aggregate	Optional Repeating	Extended Account Balance Aggregate.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<MktgInfo>	C-255	Optional	Marketing Information.

6.4.2 Account Inquiry

A client uses the Account Inquiry Message to retrieve more detailed information about a bank account than that provided by the Balance Inquiry Message. Much of the information is returned in an aggregate that is specific to each Account Type.

6.4.2.1 Request <AcctInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Card Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<IncExtBal>	Aggregate	Optional	Include Extended Balances Indicator. If TRUE, the response should include the <ExtAcctBal> aggregate and return all available balances for the type of account. If FALSE or omitted, the response should not include balances.
<IncBal>	Aggregate	Optional	Include Balances Indicator. If TRUE, the response should include the <AcctBal> and <ExtAcctBal> aggregate and return all available balances for the type of account. If FALSE or omitted, the response should not include balances.

6.4.2.2 Response <AcctInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Card Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR Echoed	Loan Account Identification Aggregate.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<IncExtBal>	Aggregate	Optional Echoed	Include Extended Balances Indicator.
<IncBal>	Aggregate	Optional Echoed	Include Balances Indicator.
<UpDt>	DateTime	Required	Account Update DateTime.
<LastTrnDt>	Date	Optional	Last Transaction Date.
<LastStmtDt>	Date	Optional	Last Statement Cycle Date.
<AcctBal>	Aggregate	Optional Repeating	Account Balance Aggregate
<ExtAcctBal>	Aggregate	Optional Repeating	Extended Account Balance Aggregate.
<DepAcctRec>	Aggregate	Optional XOR	Deposit Account Record Aggregate.
<CCAcctRec>	Aggregate	Optional XOR	Credit Card Account Record Aggregate.
<LoanAcctRec>	Aggregate	Optional XOR	Loan Account Record Aggregate.
<LOCAcctRec>	Aggregate	Optional XOR	Line of Credit Account Record Aggregate.
<MortAcctRec>	Aggregate	Optional XOR	Mortgage Account Record Aggregate.

6.4.3 Deposit Account Statement Inquiry

A client may use <DepAcctStmtInqRq> to retrieve account information for a statement period or multiple statement periods. A client may optionally provide a date range to limit the number of Account Statement aggregates that are returned in the response. Note that the statement is based on the normal cutoff cycles and the date range will return statements that *ended* (cutoff or cycled) during the specified range. The client may specify a date range that results in no statement, because there were no statement end cutoffs within the range. If the client does not specify a date range, the server returns as many statements as possible.

The client may request the detail messages associated with each returned Closing Statement by specifying TRUE in the <IncDetail> Boolean.

6.4.3.1 Request <DepAcctStmtInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Card Account Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.

Tag	Type	Usage	Description
<IncDetail>	Boolean	Optional	Include Detail Indicator. If TRUE, the response should include the detail statement messages <DepStmtTrn> for the statements returned. If FALSE or omitted, the detail messages should not be included.

6.4.3.2 Response <DepAcctStmtInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Card Account Aggregate.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<IncDetail>	Boolean	Optional Echoed	Include Detail Indicator.
<DepAcctStmtRec>	Aggregate	Optional Repeating	Deposit Account Statement Record.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier. This is a reference number for the statement.
<NextDt>	DateTime	Optional	Next Closing DateTime.
<AcctBal>	Aggregate	Required Repeating	Account Balance Aggregate. May be used to provide the opening, closing and minimum ledger balance and any other balances to be provided on the statement. The ClosingLedger must be included.
<StmtSummAmt>	Aggregate	Optional Repeating	Statement Summary Amount Aggregate. Used to return all the statement summary totals for this closing statement period.
<StmtSummType>	Open Enum	Required	Summary Types. Used to identify the type of summary data. Defined Values are: Deposits, OtherCredits, Checks, OtherDebits, Fees, IntCharged, IntEarned
<CurAmt>	Currency Amount	Required	Summary Amount.
</StmtSummAmt>			
<StartDt>	Date	Required	Selection Start Date for this statement.
<EndDt>	DateTime	Required	End for this statement.
<MktgInfo>	C-255	Optional	Marketing Information.
<URL>	URL	Optional	Contains bank-rendered statement copy.

Tag	Type	Usage	Description
<DepAcctTrnRec> **</DepAcctStmtRec>**	Aggregate	Optional Repeating	Deposit Account Transaction Record aggregate. Included if the <IncDetail> Boolean is set to TRUE in the request. One record per message for this closing statement period.

6.4.4 Credit Card Statement Inquiry

A client may use <CCAcctStmtInqRq> to retrieve account information for a statement period or multiple statement periods. A client may optionally provide a date range to limit the number of Account Statement aggregates that are returned in the response. Note that the statement is based on the normal cutoff cycles and the date range must return statements that *ended* (cutoff or cycled) during the specified range. The client may specify a date range that results in no statement, because there were no statement end cutoffs within the range. If the client does not specify a date range, the server returns as many statements as possible.

The client may request the detail messages associated with each returned Statement by specifying TRUE in the <IncDetail> Boolean.

6.4.4.1 Request <CCAcctStmtInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CardAcctId>	Aggregate	Required	Credit Card Account Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.
<IncDetail>	Boolean	Optional	Include Detail Indicator. If TRUE, the response should include the detail statement messages (CCSTMTTRN) for the statements returned. If FALSE or omitted, the detail messages should not be included.

6.4.4.2 Response <CCAcctStmtInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CardAcctId>	Aggregate	Required Echoed	Credit Card Account Aggregate.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<IncDetail>	Boolean	Optional Echoed	Include Detail Indicator.
<CCAcctStmtRec>	Aggregate	Optional Repeating	Credit Card Account Statement Record
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

Tag	Type	Usage	Description
<NextDt>	DateTime	Optional	Next Closing DateTime.
<AcctBal>	Aggregate	Required Repeating	Account Balance Aggregate. The CLOSINGOUTSTANDING balance is required.
<ExtAcctBal>	Aggregate	Optional Repeating	Extended Account Balance Aggregate. Used to report other balances for this statement end, such as Period Fee, Cashline, Outstanding Cash Advance Total, Cash Available, and Over Limit Amount
<PmtDueDt>	Date	Optional	Payment Due Date.
<MinAmtDue>	Currency Amount	Optional	Minimum Payment Due.
<StmtSummAmt>	Aggregate	Optional Repeating	Statement Summary Amount Aggregate. Used to return all the statement summary totals for this closing statement period.
<StmtSummType>	Open Enum	Required	Summary Types. Used to identify the type of summary data. Valid Values are: Deposits, OthCredits, Checks, OthDebits, Fees, IntCharged, IntEarned
<CurAmt>	Currency Amount	Required	Summary Amount.
</StmtSummAmt>			
<DelinqAging>	Aggregate	Optional Repeating	Delinquency Aging Aggregate.
<Aging>	Open Enum	Required	Number of Days Delinquent.
<CurAmt>	Currency Amount	Required	Amount Delinquent.
</DelinqAging>			
<StartDt>	DateTime	Required	Selection Start DateTime Provided to allow client to use these dates to retrieve message detail corresponding to this statement.
<EndDt>	DateTime	Required	End DateTime. Provided to allow client to use these dates to retrieve message detail corresponding to this statement.
<MktgInfo>	C-255	Optional	Marketing Information.
<URL>	URL	Optional	Contains bank-rendered statement copy.
<CCAcctTrnRec>	Aggregate	Optional Repeating	Credit Card Transaction Record Aggregate. Included if the <IncDetail> Boolean is set to TRUE in the request.
</CCAcctStmtRec>			

6.4.5 Deposit Account Transaction Inquiry

A client uses the Deposit Account Transaction Inquiry to retrieve transaction detail for a given account. A variety of selection criteria are supported.

6.4.5.1 Request <DepAcctTrnInqRq>

Tag	Type	Usage	Description

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Card Account Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate. Selection criteria are based upon message posting date.
<SelRangeCurAmt>	Aggregate	Optional	Selection Range Amount Aggregate.
<ChkRange>	Aggregate	Optional	Selection Range Check Number Aggregate.
<TrnType>	Open Enum	Optional Repeating	Transaction Type. Defined values: Debit, Credit, Withdrawal, Check, Deposit, Transfer, Payment, Interest, Dividend, DirectDeposit, DirectDebit, RepeatPayment, Fee, ServiceCharge
<TrnSrc>	Open Enum	Optional Repeating	Transaction Source. Defined values: ATM, Teller, POS, VRU, Home, ACH
<CSPRefId>	Identifier	Optional Repeating	Customer Service Provider Reference Identifier. This field is used as a selection criterion.
<SPRefId>	Identifier	Optional Repeating	Service Provider Reference Identifier. This field is used as a selection criterion.

6.4.5.2 Response <DepAcctTrnInqRs>

The Deposit Account Transaction Inquiry Response message contains a list of Deposit Account Transaction Records for those transactions that meet the selection criteria in the request.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Card Account Aggregate.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.

Tag	Type	Usage	Description
<SelRangeCurAmt>	Aggregate	Optional Echoed	Selection Range Amount Aggregate.
<ChkRange>	Aggregate	Optional Echoed	Selection Range Check Number Aggregate.
<TrnType>	Open Enum	Optional Repeating Echoed	Transaction Type.
<TrnSrc>	Open Enum	Optional Repeating Echoed	Transaction Source.
<CSPRefId>	Identifier	Optional Repeating Echoed	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional Repeating Echoed	Service Provider Reference Identifier.
<MktgInfo>	C-255	Optional	Marketing Information.
<DepAcctTrnRec>	Aggregate	Optional Repeating	Deposit Message Record Aggregate. One record per message subject to selection criteria and message records control.

6.4.6 Credit Card Account Transaction Inquiry

A client uses the Credit Card Account Transaction Inquiry to retrieve transaction detail for a given account. A variety of selection criteria are supported.

6.4.6.1 Request <CCAcctTrnInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CardAcctId>	Aggregate	Required	Credit Card Account Aggregate.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range DateTime Aggregate.
<SelRangeCurAmt>	Aggregate	Optional	Selection Range Amount Aggregate.
<TrnType>	Open Enum	Optional Repeating	Transaction Type. See Data Dictionary for details. This field is used as a selection criterion. Defined values: Debit, Credit, Withdrawal, Check, Deposit, Transfer, Payment, Interest, Dividend, DirectDeposit, DirectDebit, RepeatPayment, Fee, ServiceCharge
<TrnSrc>	Open Enum	Optional Repeating	Message Source. Defined values: ATM, Teller, POS, VRU, Home, ACH This field is used as a selection criterion.
<CSPRefId>	Identifier	Optional Repeating	Customer Service Provider Reference Identifier. This field is used as a selection criterion.

Tag	Type	Usage	Description
<SPRefId>	Identifier	Optional Repeating	Service Provider Reference Identifier. This field is used as a selection criterion.

6.4.6.2 Response <CCAcctTrnInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CardAcctId>	Aggregate	Required Echoed	Credit Card Account Aggregate.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Message Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request, and the server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range DateTime Aggregate.
<SelRangeCurAmt>	Aggregate	Optional Echoed	Selection Range Amount Aggregate.
<TrnType>	Open Enum	Optional Repeating Echoed	Message Type.
<TrnSrc>	Open Enum	Optional Repeating Echoed	Message Source.
<CSPRefId>	Identifier	Optional Repeating Echoed	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional Repeating Echoed	Service Provider Reference Identifier. This field is used as a selection criterion.
<MktgInfo>	C-255	Optional	Marketing Information.
<CCAcctTrnRec>	Aggregate	Optional Repeating	Credit Card Transaction Record Aggregate. One record per message subject to selection criteria and message records control.

6.4.7 Bank Account Transaction Image Inquiry

The Bank Account Transaction Image Order Message allows a client to request a copy of a check or sales slip. Note that the client may specify a delivery method picked from those supported in the Service Profile.

6.4.7.1 Request <BankAcctTrnImgInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.

Tag	Type	Usage	Description
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DeliveryMethod>	Open Enum	Optional Profiled values	Delivery Method. Used to request the delivery channel for requested information. See Data Dictionary for details. Value selected must be supported in Service Profile.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<StopChkInfo>	Aggregate	Optional Repeating	Check Description Aggregate. This field is used as a selection criterion.
<CustContact>	Aggregate	Optional	Customer Contact Aggregate.

6.4.7.2 Response <BankAcctTrnImgInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DeliveryMethod>	Open Enum	Optional Profiled values Echoed	Delivery Method.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR Echoed	Loan Account Identification Aggregate.
<StopChkInfo>	Aggregate	Optional Repeating Echoed	Check Description Aggregate.
<CustContact>	Aggregate	Optional Echoed	Customer Contact Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.4.8 Interest Rate Inquiry

6.4.8.1 Request <IntRateInqRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<AcctType>	Open Enum	Required	Account Type. See Data Dictionary for details.
<AcctTaxStatus>	Open Enum	Optional	Account Tax Status. See Data Dictionary for details.
<CurAmt>	Currency Amount	Optional	Currency Amount.
<Term>	Aggregate	Optional	Term Aggregate.

6.4.8.2 Response <IntRateInqRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<AcctType>	Open Enum	Required Echoed	Account Type.
<AcctTaxStatus>	Open Enum	Optional Echoed	Account Tax Status.
<CurAmt>	Currency Amount	Optional Echoed	Currency Amount.
<Term>	Aggregate	Optional Echoed	Term Aggregate. Echo of request.
<IntRateInfo>	Aggregate	Optional Repeating	Interest Rate Aggregate.
<Rate>	Decimal	Required	Interest Rate. Usage is a percentage. (e.g.—a value of 5.2 = 5.2%)
<Desc>	C-80	Optional	Short Description. Explanatory text associated with the interest rate code. Assigned by the Financial Institution.
<IntAPY>	Decimal	Optional	Annual Percentage Yield. Usage is percentage of yield on an annualized basis.
<Term>	Aggregate	Optional	Term Aggregate.
<LowCurAmt>	Currency Amount	Optional	Interest Rate Low Amount.
<HighCurAmt>	Currency Amount	Optional	Interest Rate High Amount.

Tag	Type	Usage	Description
</IntRateInfo>			

6.4.9 Bank Account Taxation Inquiry

6.4.9.1 Request <BankAcctTaxInqRq>

The client must specify a client identifier, the account identifier, and tax details.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<TaxYear>	Year	Required	Tax year.

6.4.9.2 Response <BankAcctTaxInqRs>

If research is required, requests for years other than current and prior years may require <Status> to say that it is accepted for asynchronous processing.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR Echoed	Loan Account Identification Aggregate.
<TaxYear>	Date	Required Echoed	Tax year.
<TaxId>	NC-12	Required	Customer Tax Identifier.
<AcctTaxInfo>	Aggregate	Optional Repeating	Account Tax Information Aggregate.
<Org>	Identifier	Required	Organization. Organization defining this name space. Usage is Tax Authority (e.g., state or country).
<TaxType>	Open Enum	Optional	Tax Type. Qualified by <Org>.

Tag	Type	Usage	Description
<CurAmt>	Currency Amount	Optional	Tax Amount (+ paid by customer, – earned by customer).
<PrevYrCurAmt>	Currency Amount	Optional	Previous Year Tax Amount (+ paid by customer, – earned by customer).
<Rate>	Decimal	Optional	Tax Rate. Usage is a percentage (e.g., a value of 5.2 = 5.2%). (+ paid by customer, – earned by customer)
</AcctTaxInfo>			

6.4.10 Foreign Exchange Rate Inquiry

A client may request an exchange rate or a committed exchange rate for a future message. The Financial Institution may quote the current rate or may commit a rate for some time period. If a commitment is made to honor a rate until a future date, the Financial Institution must return a commitment identifier, which is provided in the <SPRefId> in the <ForExRateInfo>.

6.4.10.1 Request <ForExRateInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<CurAmt>	Currency Amount	Required	Message Amount. Always in the currency of the FROM account.
<CurCode>	NC-3	Required	TO ISO Currency Code. FROM currency is <DepAcctIdFrom> or <CardAcctIdFrom> currency.
<ForExRateType>	Closed Enum	Optional	Requested Rate Type.

6.4.10.2 Response <ForExRateInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctIdFrom>	Aggregate	Required XOR Echoed	Deposit Account Identification.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Account Identification Aggregate.

Tag	Type	Usage	Description
<CurAmt>	Currency Amount	Required Echoed	Message Amount. Always in the currency of the FROM account.
<CurCode>	NC-3	Required Echoed	TO ISO Currency Code. FROM currency is <DepAcctIdFrom> or <CardAcctIdFrom> currency.
<ForExRateType>	Closed Enum	Optional Echoed	Requested Rate Type.
<ForExRateRec>	Aggregate	Required	Foreign Exchange Rate Record Aggregate
<ForExRateId>	Identifier	Optional *but see Description*	Foreign Exchange Rate Identifier. This is required if the server is committing to an exchange rate.
<ForExRateInfo>	Aggregate	Required	Foreign Exchange Rate Information Aggregate. This aggregate provides a Financial Institution commitment to honor this rate for the period of time specified.
</ForExRateRec>			

6.5 Stop Check

6.5.1 Stop Check Add

The client sends a <StopChkAddRq> message to request that a check be stopped. The minimum function server must support identification of the check via check number <ChkNum> within <StopChkInfo>. The server stores the additional data elements within <StopChkInfo> (payee name and amount) to help identify the stop check request for future reference. They are not used to identify the check to be stopped.

6.5.1.1 Request <StopChkAddRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<ChkRange>	Aggregate	Required XOR	Selection Range—Check Number Aggregate. <ChkRange> or <StopChkInfo>, but not both.
<StopChkInfo>	Aggregate	Required XOR	Check Description Aggregate. This field is used as a selection criterion.

6.5.1.2 Response <StopChkAddRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR Echoed	Loan Account Identification Aggregate.
<ChkRange>	Aggregate	Required XOR Echoed	Selection Range—Check Number Aggregate.
<StopChkInfo>	Aggregate	Required XOR Echoed	Stop Check Information Aggregate.
<StopChkRec>	Aggregate	Required Repeating	Stop Check Record. This aggregate contains information about the server-stored stop check information and status.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.5.2 Stop Check Cancel

A client may cancel a Stop Check Payment Request with a Financial Institution.

6.5.2.1 Request <StopChkCanRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<ChkRange>	Aggregate	Required XOR	Selection Range Check Number Aggregate.
<StopChkInfo>	Aggregate	Required XOR	Check Description Aggregate. Usage is selection criteria.

6.5.2.2 Response <StopChkCanRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR Echoed	Loan Account Identification Aggregate.
<ChkRange>	Aggregate	Required XOR Echoed	Selection Range Check Number Aggregate.
<StopChkInfo>	Aggregate	Required XOR Echoed	Stop Check Information Aggregate.
<StopChkRec>	Aggregate	Required Repeating	Stop Check Record.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier. This is the <CSPRefId> for this Cancel Stop Check message.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier. This is the <SPRefId> for this Cancel Stop Check message.

6.5.3 Stop Check Inquiry

The Stop Check Inquiry message allows a client to retrieve records of both current and completed stopped checks. Since volume of stopped checks is expected to be relatively low on a per-customer basis, there is not a separate Stop Check History Inquiry.

6.5.3.1 Request <StopChkInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Detail Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<ChkRange>	Aggregate	Optional XOR	Selection Range—Check Number Aggregate.

Tag	Type	Usage	Description
<StopChkInfo>	Aggregate	Optional Repeating XOR	Check Description Aggregate. This field is used as a selection criterion.
<SelRangeDt>	Aggregate	Optional XOR	Selection Range Date Aggregate. Returns all stop check requests originated in this date range.
<IncToken>	Boolean	Optional	Include Token. If TRUE, a <Token> should be included in the response to set a base for future Audit messages. If FALSE or omitted, no <Token> is returned.

6.5.3.2 Response <StopChkInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Detail Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR Echoed	Loan Account Identification Aggregate.
<ChkRange>	Aggregate	Optional XOR Echoed	Selection Range—Check Number Aggregate.
<StopChkInfo>	Aggregate	Optional XOR Repeating Echoed	Stopped Check Information Aggregate.
<SelRangeDt>	Aggregate	Optional XOR Echoed	Selection Range Date Aggregate.
<StopChkRec>	Aggregate	Optional Repeating	Stopped Check Record Aggregate. One record is returned for each stopped check subject to selection criteria.
<Token>	Identifier	Optional	Token. Server Assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. The token is only included if <IncToken>=TRUE in the request. <Token>=0 is returned if no records are returned within the response.

6.5.4 Stop Check Audit

6.5.4.1 Request <StopChkAudRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate. Date of the Stop Check request.
<ChkRange>	Aggregate	Optional	Selection Range—Check Number Aggregate.

6.5.4.2 Response <StopChkAudRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR Echoed	Loan Account Identification Aggregate.
<SelRangeDt>	Aggregate	Optional Echoed	Stop Check Audit Selection Criteria Aggregate.
<ChkRange>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<StopChkMsgRec>	Aggregate	Optional Repeating	Stop Check Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the <StopChkMsgRec> request.
<MsgRecDt>	Date/Time	Optional	Audit Record Creation Date. Date/time at which the audit record was stored/created by the service provider..

Tag	Type	Usage	Description
<StopChkAddRs>	Aggregate	Required XOR	Stop Check Response Message Aggregate. One record is returned for each Stop Check message, which meets the selection criteria specified in the request.
<StopChkCanRs>	Aggregate	Required XOR	Cancel Stop Check Response Message Aggregate. One record is returned for each Cancel Stop Check message, which meets the selection criteria specified in the request.
</StopChkMsgRec>			

6.5.5 Stop Check Sync

6.5.5.1 Request <StopChkSyncRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<DepAcctId>	Aggregate	Required XOR	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR	Credit Account Identification Aggregate.
<LoanAcctId>	Aggregate	Required XOR	Loan Account Identification Aggregate.
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.

6.5.5.2 Response <StopChkSyncRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<DepAcctId>	Aggregate	Required XOR Echoed	Deposit Account Identification Aggregate.
<CardAcctId>	Aggregate	Required XOR Echoed	Credit Account Identification Aggregate.

Tag	Type	Usage	Description
<LoanAcctId>	Aggregate	Required XOR Echoed	Loan Account Identification Aggregate.
<Token>	Aggregate	Required Echoed	Stop Check Synchronization Aggregate.
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it.
<StopChkMsgRec>	Aggregate	Optional Repeating	Stop Check Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the <StopChkMsgRec> request.
<MsgRecDt>	Date/Time	Optional	Audit Record Creation Date. Date/time at which the audit record was stored/created by the service provider..
<StopChkAddRs>	Aggregate	Required XOR	Stop Check Response Message Aggregate. One record is returned for each Stop Check message, which meets the selection criteria specified in the request.
<StopChkCanRs>	Aggregate	Required XOR	Cancel Stop Check Response Message Aggregate. One record is returned for each Cancel Stop Check message, which meets the selection criteria specified in the request.
</StopChkMsgRec>			

6.6 Single Transfer

This section contains messages for transferring money between accounts within one Financial Institution, or among service providers.

6.6.1 Funds Transfer Add

6.6.1.1 Request <XferAddRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferInfo>	Aggregate	Required	Transfer Information Aggregate.
<DupChkOverride>	Boolean	Optional	Duplicate Check Override Flag. When set to TRUE, requests that the server not perform duplicate checking if any is normally performed. The client is affirming that this is a new transfer being added.

6.6.1.2 Response <XferAddRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.

Tag	Type	Usage	Description
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferInfo>	Aggregate	Required Echoed	Transfer Information Aggregate.
<DupChkOverride>	Boolean	Optional Echoed	Duplicate Check Override Flag.
<XferRec>	Aggregate	Required	Transfer Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.6.2 Funds Transfer Modify

6.6.2.1 Request <XferModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferId>	Identifier	Required	Transfer Identifier. Assigned by the server at the time the Transfer is first added. Cannot be changed by the client.
<XferInfo>	Aggregate	Required	Transfer Information Aggregate.
<DupChkOverride>	Boolean	Optional	Duplicate Check Override Flag. When set to TRUE, requests that the server not perform duplicate checking if any is normally performed. The client is affirming that this is a new transfer being added.

6.6.2.2 Response <XferModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferId>	Identifier	Required Echoed	Transfer Identifier.
<XferInfo>	Aggregate	Required Echoed	Transfer Information Aggregate.
<DupChkOverride>	Boolean	Optional Echoed	Duplicate Check Override Flag. When set to TRUE, requests that the server not perform duplicate checking if any is normally performed. The client is affirming that this is a new transfer being added.
<XferRec>	Aggregate	Required	Transfer Record Aggregate.

Tag	Type	Usage	Description
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.6.3 Funds Transfer Cancel

6.6.3.1 Request <XferCanRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferId>	Identifier	Required	Transfer Identifier. Assigned by the server at the time the Transfer is first added. Cannot be changed by the client.

6.6.3.2 Response <XferCanRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferId>	Identifier	Required Echoed	Transfer Identifier.
<XferRec>	Aggregate	Required	Transfer Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.6.4 Funds Transfer Status Modify

6.6.4.1 Request <XferStatusModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferId>	UUID	Required	Transfer Identifier.
<XferStatus>	Aggregate	Required	Transfer Status Aggregate.

6.6.4.2 Response <XferStatusModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferId>	UUID	Required Echoed	Transfer Identifier.
<XferStatus>	Aggregate	Required Echoed	Transfer Status Aggregate.
<CSPRefId>	Identifier	Optional	Financial Institute Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.6.5 Funds Transfer Inquiry

6.6.5.1 Request <XferInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<XferId>	Identifier	Optional Repeating	Transfer Identifier.
<RecXferId>	Identifier	Optional Repeating	Recurring Transfer Model Identifier. This field is used as a selection criterion to select only transfer instances generated by a Recurring Transfer Model. Inclusion of <RecXferId> must not cause this message to include the recurring transfer model itself. The model may be retrieved using the Recurring Transfer Inquiry.
<CSPRefId>	Identifier	Optional Repeating	Customer Service Provider Reference Identifier. This field is used as a selection criterion.
<SPRefId>	Identifier	Optional Repeating	Service Provider Reference Identifier. This field is used as a selection criterion.
<DepAcctIdFrom>	Aggregate	Optional XOR	Deposit Account Identification. This field is used as a selection criterion.
<CardAcctIdFrom>	Aggregate	Optional XOR	Credit Account Identification. This field is used as a selection criterion.
<LoanAcctIdFrom>	Aggregate	Optional XOR	Loan Account Identification. This field is used as a selection criterion.
<DepAcctIdTo>	Aggregate	Optional XOR	Deposit Account Identification. This field is used as a selection criterion and is subject to server support in the Service profile.

Tag	Type	Usage	Description
<CardAcctIdTo>	Aggregate	Optional XOR	Credit Account Identification. This field is used as a selection criterion and is subject to server support in the Service profile
<LoanAcctIdTo>	Aggregate	Optional XOR	Loan Account Identification. This field is used as a selection criterion and is subject to server support in the Service profile
<XferStatusCode>	Closed Enum	Optional Repeating	Transfer Status Code.
<SelRangeDueDt>	Aggregate	Optional	Selection Range Due Date Aggregate.
<SelRangeCurAmt>	Aggregate	Optional	Selection Range Amount Aggregate.
<IncToken>	Boolean	Optional	Include Token. If TRUE, a <Token> should be included in the response to set a base for future Audit messages. If FALSE or omitted, no <Token> is returned.

6.6.5.2 Response <XferInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<XferId>	Identifier	Optional Repeating Echoed	Transfer Identifier.
<RecXferId>	Identifier	Optional Repeating Echoed	Recurring Transfer Model Identifier. This field is used as a selection criterion to select only transfer instances generated by a Recurring Transfer Model. Inclusion of <RecXferId> must not cause this message to include the recurring transfer model itself. The model may be retrieved using the Recurring Transfer Inquiry.
<CSPRefId>	Identifier	Optional Repeating Echoed	Customer Service Provider Reference Identifier. This field is used as a selection criterion.
<SPRefId>	Identifier	Optional Repeating Echoed	Service Provider Reference Identifier. This field is used as a selection criterion.
<DepAcctIdFrom>	Aggregate	Optional XOR Echoed	Deposit Account Identification. This field is used as a selection criterion.
<CardAcctIdFrom>	Aggregate	Optional XOR Echoed	Credit Account Identification. This field is used as a selection criterion.
<LoanAcctIdFrom>	Aggregate	Optional XOR Echoed	Loan Account Identification. This field is used as a selection criterion.

Tag	Type	Usage	Description
<DepAcctIdTo>	Aggregate	Optional XOR Echoed	Deposit Account Identification. This field is used as a selection criterion and is subject to server support in the Service profile.
<CardAcctIdTo>	Aggregate	Optional XOR Echoed	Credit Account Identification. This field is used as a selection criterion and is subject to server support in the Service profile
<LoanAcctIdTo>	Aggregate	Optional XOR Echoed	Loan Account Identification. This field is used as a selection criterion and is subject to server support in the Service profile
<XferStatusCode>	Closed Enum	Optional Repeating Echoed	Processing Status.
<SelRangeDueDt>	Aggregate	Optional Echoed	Selection Range Due Date Aggregate.
<SelRangeCurAmt>	Aggregate	Optional Echoed	Selection Range Amount Aggregate.
<XferRec>	Aggregate	Optional Repeating	Transfer Record Aggregate. These records are generated by the server and reflect the current state of the customer's Transfers. The records are filtered by the selection criteria specified in the request message. Also, note that transfers may have been generated by a client (using <XferAddRq>), or may have been generated by the server from one of the customer's Recurring Transfer Models.
<Token>	Identifier	Optional *but see Description*	Token. Server Assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. The token is only included if <IncToken>=TRUE in the request. <Token>=0 is returned if no records are returned within the response.

6.6.6 Funds Transfer Audit

6.6.6.1 Request <XferAudRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional	Audit Selection Action. Used to identify actions associated with the object that is being audited (e.g. transfer). This field is used as a selection criterion.
<XferId>	Identifier	Optional	Transfer Identifier. Assigned by the server at the time the Transfer is first added. Cannot be changed by the client. This field is used as a selection criterion.

Tag	*Type*	*Usage*	*Description*
<RecXferId>	Identifier	Optional	Recurring Transfer Model Identifier. Assigned by the server at the time the Recurring Transfer Model is first added. Cannot be changed by the client. This field is used as a selection criterion.

6.6.6.2 Response <XferAudRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Message Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional Echoed	Audit Selection Action. Used to identify actions associated with the object that is being audited (e.g. transfer). This field is used as a selection criterion.
<XferId>	Identifier	Optional Echoed	Transfer Identifier. Assigned by the server at the time the Transfer is first added. Cannot be changed by the client. This field is used as a selection criterion.
<RecXferId>	Identifier	Optional Echoed	Recurring Transfer Model Identifier. Assigned by the server at the time the Recurring Transfer Model is first added. Cannot be changed by the client. This field is used as a selection criterion.
<XferMsgRec>	Aggregate	Optional Repeating	Transfer Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This aggregate identifies the party that actually issued the <XferMsgRec> request.
<MsgRecDt>	Date/Time	Optional	Message Record Creation Date. The date/time at which the message record was stored/created by the service provider.
<XferAddRs>	Aggregate	Required XOR	Transfer Add Response Message Aggregate.
<XferModRs>	Aggregate	Required XOR	Transfer Modify Response Message Aggregate.
<XferStatusModRs>	Aggregate	Required XOR	Transfer Status Modify Response Message Aggregate.
<XferCanRs>	Aggregate	Required XOR	Transfer Cancel Response Message Aggregate.
</XferMsgRec>			

6.6.7 Funds Transfer Sync

6.6.7.1 Request <XferSyncRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.
<DepAcctIdFrom>	Aggregate	Required XOR	Deposit Account Detail Aggregate.
<CardAcctIdFrom>	Aggregate	Required XOR	Credit Account Detail Aggregate.
<LoanAcctIdFrom>	Aggregate	Required XOR	Loan Account Detail Aggregate.

6.6.7.2 Response <XferSyncRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Message Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<Token>	Identifier	Required Echoed	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.
<DepAcctIdFrom>	Aggregate	Required XOR Echoed	Deposit Account Detail Aggregate.
<CardAcctIdFrom>	Aggregate	Required XOR Echoed	Credit Account Detail Aggregate.
<LoanAcctIdFrom>	Aggregate	Required XOR Echoed	Loan Account Detail Aggregate.

Tag	Type	Usage	Description
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. If the client sent a token in the request, the server returns a new token based on this audit message.
<XferMsgRec>	Aggregate	Optional Repeating	Transfer Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This aggregate identifies the party that actually issued the <XferMsgRec> request.
<MsgRecDt>	Date/Time	Optional	Message Record Creation Date. The date/time at which the message record was stored/created by the service provider.
<XferAddRs>	Aggregate	Required XOR	Transfer Add Response Message Aggregate.
<XferModRs>	Aggregate	Required XOR	Transfer Modify Response Message Aggregate.
<XferStatusModRs>	Aggregate	Required XOR	Transfer Status Modify Response Message Aggregate.
<XferCanRs>	Aggregate	Required XOR	Transfer Cancel Response Message Aggregate.
</XferMsgRec>			

6.7 Recurring Transfer Model

6.7.1 Recurring Transfer Model Add

6.7.1.1 Request <RecXferAddRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferInfo>	Aggregate	Required	Transfer Information Aggregate.
<RecModelInfo>	Aggregate	Required	Recurring Model Information Aggregate.
<DupChkOverride>	Boolean	Optional	Duplicate Check Override Flag. When set to TRUE, requests that the server not perform duplicate checking if any is normally performed. The client is affirming that this is a new transfer being added.

6.7.1.2 Response <RecXferAddRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.

Tag	Type	Usage	Description
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<XferInfo>	Aggregate	Required Echoed	Transfer Information Aggregate.
<RecModelInfo>	Aggregate	Required Echoed	Recurring Model Information Aggregate.
<DupChkOverride>	Boolean	Optional Echoed	Duplicate Check Override Flag.
<RecXferRec>	Aggregate	Required	Recurring Transfer Model Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.7.2 Recurring Transfer Model Modify

6.7.2.1 Request <RecXferModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecXferId>	Identifier	Required	Recurring Transfer Model Identifier. Assigned by the server at the time the Recurring Transfer Model is first added. Cannot be changed by the client.
<XferInfo>	Aggregate	Required	Transfer Information.
<RecModelInfo>	Aggregate	Required	Recurring Model Information.
<ModPending>	Boolean	Optional Profiled support	Modify Pending Flag. If allowed by profile and set by client, any changes to the recurring model must be propagated to pending transfers previously spawned from the model. Regardless of this field, any instances spawned in the future must be based on the changed model.

6.7.2.2 Response <RecXferModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecXferId>	Identifier	Required Echoed	Recurring Transfer Model Identifier.
<XferInfo>	Aggregate	Required Echoed	Transfer Information.

Tag	Type	Usage	Description
<RecModelInfo>	Aggregate	Required Echoed	Recurring Model Information.
<ModPending>	Boolean	Optional Profiled support Echoed	Modify Pending Flag.
<RecXferRec>	Aggregate	Required	Recurring Transfer Model Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.7.3 Recurring Transfer Model Cancel

6.7.3.1 Request <RecXferCanRq>

***Note**—Cancel Recurring Transfer Model always cancels pending transfers based on the model.*

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecXferId>	Identifier	Required	Recurring Transfer Model Identifier. Assigned by the server at the time the Recurring Transfer Model is first added. Cannot be changed by the client.
<CascadeDel>	Boolean	Optional	Cascade Delete. If TRUE, server must delete all dependent objects when this object is deleted. If FALSE or omitted, the recurring model must not be deleted if dependent transfers exist.

6.7.3.2 Response <RecXferCanRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecXferId>	Identifier	Required Echoed	Recurring Transfer Model Identifier.
<CascadeDel>	Boolean	Optional Echoed	Cascade Delete.
<RecXferRec>	Aggregate	Required XOR	Recurring Transfer Model Record Aggregate.
<DependentType>	Open Enum	Required XOR Repeating	An aggregate that would contain a list of depending object types that exist for the recurring model.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.7.4 Recurring Transfer Model Inquiry

6.7.4.1 Request <RecXferInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<RecXferId>	Identifier	Optional Repeating	Recurring Transfer Model Identifier. Assigned by the server at the time the Recurring Transfer Model is first added. Cannot be changed by the client. This field is used as a selection criterion.
<DepAcctIdFrom>	Aggregate	Optional XOR	Deposit Account Identification Aggregate. This field is used as a selection criterion.
<CardAcctIdFrom>	Aggregate	Optional XOR	Credit Account Identification Aggregate. This field is used as a selection criterion.
<LoanAcctIdFrom>	Aggregate	Optional XOR	Loan Account Identification Aggregate. This field is used as a selection criterion.
<SelRangeCurAmt>	Aggregate	Optional	Selection Range Amount Aggregate.
<CSPRefId>	Identifier	Optional Repeating	Customer Service Provider Reference Identifier. This field is used as a selection criterion.
<SPRefId>	Identifier	Optional Repeating	Service Provider Reference Identifier. This field is used as a selection criterion.
<IncToken>	Boolean	Optional	Include Token. If TRUE, a <Token> should be included in the response to set a base for future Audit messages. If FALSE or omitted, no <Token> is returned.

6.7.4.2 Response <RecXferInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.

Tag	Type	Usage	Description
<RecXferId>	Identifier	Optional Repeating Echoed	Recurring Transfer Model Identifier.
<DepAcctIdFrom>	Aggregate	Optional XOR Echoed	Deposit Account Detail Aggregate.
<CardAcctIdFrom>	Aggregate	Optional XOR Echoed	Credit Account Detail Aggregate.
<LoanAcctIdFrom>	Aggregate	Optional XOR Echoed	Loan Account Detail Aggregate.
<SelRangeCurAmt>	Aggregate	Optional Echoed	Selection Range Amount Aggregate.
<CSPRefId>	Identifier	Optional Repeating Echoed	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional Repeating Echoed	Service Provider Reference Identifier.
<RecXferRec>	Aggregate	Optional Repeating	Recurring Transfer Model Record Aggregate. One record is returned for each of the customer's Recurring Transfer Models that meet the selection criteria specified in the request message.
<Token>	Identifier	Optional *but see Description*	Token. Server Assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. The token is only included if <IncToken>=TRUE in the request. <Token>=0 is returned if no records are returned within the response.

6.7.5 Recurring Transfer Model Audit

6.7.5.1 Request <RecXferAudRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Message Records Control Input Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional	Action. Used to identify actions associated with the object that is being audited (e.g. payment, transfer, etc.). This field is used as a selection criterion.
<RecXferId>	Identifier	Optional	Recurring Transfer Model Identifier. Assigned by the server at the time the Recurring Transfer Model is first added. Cannot be changed by the client. This field is used as a selection criterion.

6.7.5.2 Response <RecXferAudRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional Echoed	Action. Used to identify actions associated with the object that is being audited (e.g. payment, transfer, etc.). This field is used as a selection criterion.
<RecXferId>	Identifier	Optional Echoed	Recurring Transfer Model Identifier. Assigned by the server at the time the Recurring Transfer Model is first added. Cannot be changed by the client. This field is used as a selection criterion.
<RecXferMsgRec>	Aggregate	Optional Repeating	Recurring Transfer Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This aggregate identifies the party that actually issued the <RecXferMsgRec> request.
<MsgRecDt>	Date/Time	Optional	Message Record Creation Date. The date/time at which the message record was stored/created by the service provider.
<RecXferAddRs>	Aggregate	Required XOR	Recurring Transfer Add Response Message Aggregate.
<RecXferModRs>	Aggregate	Required XOR	Recurring Transfer Modify Response Message Aggregate.
<RecXferCanRs>	Aggregate	Required XOR	Recurring Transfer Cancel Response Message Aggregate.
</RecXferMsgRec>			

6.7.6 Recurring Transfer Model Sync

6.7.6.1 Request <RecXferSyncRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Message Records Control Input Aggregate.

Tag	Type	Usage	Description
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.
<DepAcctIdFrom>	Aggregate	Required XOR	Deposit Account Detail Aggregate.
<CardAcctIdFrom>	Aggregate	Required XOR	Credit Account Detail Aggregate.
<LoanAcctIdFrom>	Aggregate	Required XOR	Loan Account Detail Aggregate.

6.7.6.2 Response <RecXferSyncRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<DepAcctIdFrom>	Aggregate	Required XOR	Deposit Account Detail Aggregate.
<CardAcctIdFrom>	Aggregate	Required XOR	Credit Account Detail Aggregate.
<LoanAcctIdFrom>	Aggregate	Required XOR	Loan Account Detail Aggregate.
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it.
<RecXferMsgRec>	Aggregate	Optional Repeating	Recurring Transfer Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This aggregate identifies the party that actually issued the <RecXferMsgRec> request.
<MsgRecDt>	Date/Time	Optional	Message Record Creation Date. The date/time at which the message record was stored/created by the service provider.
<RecXferAddRs>	Aggregate	Required XOR	Recurring Transfer Add Response Message Aggregate.
<RecXferModRs>	Aggregate	Required XOR	Recurring Transfer Modify Response Message Aggregate.
<RecXferCanRs>	Aggregate	Required XOR	Recurring Transfer Cancel Response Message Aggregate.
</RecXferMsgRec>			

6.8 Customer Communications

6.8.1 Check Order

6.8.1.1 Request <ChkOrdAddRq>

The client sends a <ChkOrdAddRq> to place an order for more checks.

> ***Note:*** *the customer must determine the supported values for <ChkCount> and <ChkBkStyleId> through an out-of-band process.*

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<ChkOrdInfo>	Aggregate	Required	Check Order Information Aggregate

6.8.1.2 Response <ChkOrdAddRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<ChkOrdInfo>	Aggregate	Required Echoed	Check Order Information Aggregate.
<ChkOrdRec>	Aggregate	Required	Check Order Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.8.2 Deposit Book Order

The Deposit Book Order message allows a client to order a Deposit Book from a Financial Institution.

6.8.2.1 Request <DepBkOrdAddRq>

The client must specify a client identifier, account identifier, number of deposit slips, deposit book style, and delivery method when ordering a Deposit Book.

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.

Tag	Type	Usage	Description
<DepAcctId>	Aggregate	Required	Deposit Account Identification Aggregate.
<Count>	Long	Optional	Number of Deposit Slips. If omitted, the number used for the previous order must be used.
<DepBkStyleId>	Identifier	Optional	Style of Deposit Book.
<DeliveryMethod>	Open Enum	Optional Profiled values	Delivery Method. Used to request the delivery channel for requested information. See Data Dictionary for details. Default is Post. Value selected must be supported in Service profile.

6.8.2.2 Response <DepBkOrdAddRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<DepAcctId>	Aggregate	Required Echoed	Deposit Account Identification Aggregate.
<Count>	Long	Optional Echoed	Number of Deposit Slips.
<DepBkStyleId>	Identifier	Optional Echoed	Style of Deposit Book.
<DeliveryMethod>	Open Enum	Optional Profiled values Echoed	Delivery Method.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

6.9 Banking Service Profile <BankSvcProfInfo>

The Profile for the Banking Service <BankSvcProfInfo> is defined below. This profile is returned to the client in <SvcProfInqRq> and provides information on how the client should use the Banking Service.

Tag	Type	Usage	Description
<SvcCore>	Aggregate	Required	Service Core Aggregate. Information specified for every service.
<MsgSupt>	Open Enum	Optional Repeating	Supported Messages. This is a list of messages that are supported for Banking. The convention is to use the name of the message without the Rq or Rs so that each message is only listed once. Defined values: BalInq, AcctInq, DepAcctStmtInq, CCAcctStmtInq, DepAcctTrnInq, CCAcctTrnInq, BankAcctTrnImgInq, IntRateInq, BankAcctTaxInq, ForExRateInq, StpChkAdd, StpChkCan, StpChkInq, StpChkAud, StpChkSync, XferAdd, XferMod, XferCan, XferStatusMod, XferInq, XferAud, XferSync, RecXferAdd, RecXferMod, RecXferCan, RecXferInq, RecXferAud, RecXferSync, ChkOrdAdd, DepBkOrdAdd

Tag	*Type*	*Usage*	*Description*
<OptSupt>	Open Enum	Optional Repeating	Options Supported. Defined values: RecCtrl.
<DeliveryMethod>	Open Enum	Optional Repeating	Delivery Method. Defined values: Channel, Home, Post, UPS, Courier. Note that although these are valid values for this element, they may not be appropriate for a particular message and may result in rejection. In this case, a response must be sent to the customer with an appropriate Status Code.
<StopChkProf>	Aggregate	Optional	Stop Check Profile Aggregate. Included if supported.
<OptSupt>	Open Enum	Optional Repeating	Options Supported. Valid values: StopChkRangeSC.
<PrcSched>	Aggregate	Optional	Processing Schedule Aggregate. If omitted, the default processing schedule is assumed.
<FeeCurAmt>	Currency Amount	Optional	Default Check Stop Fee.
</StopChkProf>			
<XferProf>	Aggregate	Optional *but see Description*	Transfer Profile Aggregate. Must be included if funds transfers are supported.
<InterXferProf>	Aggregate	Optional *but see Description*	Interbank Transfer Profile Aggregate. Must be included if interbank transfers are supported
<XferProf>	Aggregate	Optional	Transfer Profile Aggregate.
<DomXferFeeCurAmt>	Currency Amount	Optional	Standard fee for a domestic interbank transfer.
<IntlXferFeeCurAmt>	Currency Amount	Optional	Standard fee for an international interbank transfer.
<USA.ACHProf>	Aggregate	Optional	ACH Profile Aggregate. This is for use in the United States.
<DaysWith>	Long	Required	Number of Days Before Withdrawal. Number of days before processing date that funds are withdrawn.
<DfltDaysToPay>	Long	Required	Default Days to Pay.
<CanWind>	Long	Required	Cancellation Window—Refers to number of days after a transfer is initiated using ACH during which the transfer may be cancelled.
</USA.ACHProf>			
</InterXferProf>			

7 The Pay Service <PaySvc>

The IFX Specification defines all functionality related to consumer and business payment in the Pay Service <PaySvc>.

Clients use the Pay Service to manage a list of payees, schedule individual payments, and define recurring payment models that automatically generate fixed-amount payments at regular intervals to a given payee. Payments may be made to organizations (Billers) with which the CPP (also described in this chapter as the customer's Pay provider, and may be a Financial Institution or Service Provider) has contracts for payment consolidation. Payments may also be made to individuals or organizations that have no such relationship with the customer's Pay provider (Fully-Specified Payees) or to accounts at other Financial Institutions (Interbank Transfer Payees).

7.1 Description

7.1.1 Payees

7.1.1.1 Types of Payees

As mentioned previously, a payee may be one of four types: a Standard Payee, a Fully-Specified Payee, a Biller Payee, or a Transfer Payee.

A *Standard Payee* is a merchant or payment recipient whose remittance attributes are well known to the CPP. Typically remittance attributes include remittance method (ACH-CIE, ACH-CTX, RPS, Epay, etc.,) concentrator, lockbox, posting exceptions contact, etc. A standard payee is uniquely identified within a CPP with a payee identifier <StdPayeeId>. The customer need only reference the Standard Payee. A customer should not be able to modify the <StdPayeeId> of a standard payee. Standard Payees are typically national, regional, or large local companies or organizations (such as billers) that have contracted with the CPP to consolidate payments from multiple consumers and small businesses. Because the CPP consolidates payments, it already has sufficient remittance information about the biller in its systems, and the customer does not need to enter all the information required for a Fully-Specified Payee. In some countries, Standard Payees are assigned unique identifiers on a national basis, while in others a biller identifier may only be significant to a single CPP. IFX allows for both cases by qualifying a biller number with the name of the organization that assigned it (e.g., the <StdPayeeId> is the Biller as known by the CPP, and the <BillerId> is the Biller as known by the Biller or BSP). Note that anytime a <StdPayeeId> exists outside of a payment message, it must include the CPP <SPName> to scope its value.

A *Fully-Specified Payee* is a payee for which a customer must enter all information needed for his or her Pay provider to identify the payee and payment destination, such as the payee's full name and address. Some Pay providers may also require that customers enter a telephone number for the payee. Most individuals and small businesses are likely to be Fully-Specified Payees.

A *Biller Payee* is one where the details about that payee (name, address, acctid, remittance information, payment instruments) have been obtained from a BSP as a result of a Biller Inquiry <BillerInqRq> or Presentment Service/Account Link Add <SvcAcctAddRq>.

A *Transfer Payee* is a payee for which the customer knows the Financial Institution and Account information where a payment may be made using a funds transfer.

It should be noted that the customer is permitted to change name/address or destination account information about the *Fully-Specified Payee* and *Transfer Payee,* but the customer cannot change *Biller* name and address information. Only Financial Institutions or Service Providers may change *Biller* information.

7.1.1.2 Common Payee Information

Although a customer must specify some different information for each type of payee, there are a number of attributes common to all types of payees.

A *Payee Name* and, optionally, a *Payee Nickname* may be assigned to each payee. The name is typically the payee's legal name and may be used by the Pay provider as well as the client. A payee nickname may be assigned by the customer and is only used as a user-friendly name for ease of recall for later use.

All Pay providers should allow a *Customer Payee Account* number to be stored as part of each payee record. This number is then sent to the Payee with a payment to allow the payee to correlate the payment with one of their customer accounts. While this element is not required for all payments, it should be included if it is known. The *Customer Payee Account* within the Payee definition is used as a default for all payments to that payee. If a *Customer Payee Account* is specified as part of the payment, it must be used for that payment, but *must not* modify the C*ustomer Payee Account* within the payee definition.

While not explicitly supported within the IFX specification, a customer may have multiple accounts with the payee. For example, a customer who purchases auto and homeowners insurance from the same insurance company may want to set up payments for both policies. The customer adds the insurance company to his payee list as two independent payees, each with it's own customer account number, based on instructions from the insurance company ("use account 11023732 for auto insurance payments, and account 97584324 for homeowners insurance payments.") The customer may use the <Nickname> field to differentiate between payees, e.g., ABC Insurance—Home and ABC Insurance—Auto.

Default Payment Information may also be assigned to a payee so that when a customer is adding a payment or a recurring payment model for that payee, some fields may be pre-populated with default values for the customer's convenience. The customer may always elect to change the values for a particular payment, but the defaults are useful to help a customer quickly enter payment information for common payments. Payment defaults, which may be assigned to a payee, are the funding account for payments, the category to which payments are assigned, and the memo that is associated with the payment.

7.1.1.3 Customer Payee Management—Customer Payee List and <CustPayeeId>

Some Pay Service Providers require an individual list of Payees for each customer, while others do not. Pay Service Providers within the United States typically support this feature, while Canadian and European systems generally do not support it. A Service Profile option is provided to specify whether this feature is supported or not.

Each customer maintains a list of payees on the Pay provider's server. This list contains payees that the customer has added, either explicitly using the Payee Add message <PayeeAddRq> or implicitly using the Payment Add <PmtAddRq> or Recurring Payment Model Add <RecPmtAddRq> messages (which allow a payee to be added along with a payment). In any case, the customer's Payee list may contain any or all of the four types of payees and only contains payees that the customer has added. Each payee on the customer's payee list is assigned a unique identifier <CustPayeeId> that is used for unambiguous communications between the client and server. The <CustPayeeId> is unique only for that particular customer; another customer may use the same <CustPayeeId> value to refer to a different payee.

Once a payee appears on the customer's payee list, the client may use the Customer Payee Modify <CustPayeeModRq> and Customer Payee Delete <CustPayeeDelRq> messages to keep the payee information up-to-date or to remove the payee from the customer's payee list.

A client may request a current view of the customer's payee list at any time by using the Customer Payee Inquiry <CustPayeeInqRq> message. Clients that keep a local copy of a customer's payee list may use this message to refresh their copy of the list. They may also request a "playback" of the messages that affected the customer's payee list since a given time by using the Customer Payee Audit <CustPayeeAudRq> and/or Customer Payee Synchronization <CustPayeeSyncRq> messages.

7.1.1.4 Pay Provider Payee Management—Duplicate Checking and Payee Type Conversion

Pay providers also manage customer Payee Lists. Two common practices among Pay providers are duplicate checking and payee type conversion.

Some Pay providers perform *Duplicate checking* when a Payee record is added to or modified on a customer's Payee List. In this case, the Pay provider verifies that the Payee is not already on the customer's payee list.

Payee type conversion to Biller allows payments to Billers to be paid effectively. A customer may be unaware that a payee is actually considered a biller by his or her Pay provider and may enter a payment to a Fully Specified Payee by entering the payee's complete name and address. The Pay provider may check additions or modifications to a customer's Payee List against its biller database and convert the Fully Specified Payee to a biller payee if a match is detected. This checking may be performed when the Customer Payee is added or modified, or done in a background process, which converts the Fully-Specified payee to a Biller payee at some time after it is added to or modified on the payee list. The customer may obtain information about the payee conversion the next time a Payee Inquiry message is sent, or when the client performs a Customer Payee Synchronization message to update its records.

Payee type conversion to Fully Specified Payee may occur if a Biller terminates a relationship with the customer's Pay provider. In this case, a Fully-Specified Payee Detail aggregate may be substituted for the Biller ID in the Payee record on the customer's Payee List.

7.1.1.5 Standard Payee Inquiry

Although the list of Standard Payees with contractual arrangements with a Pay provider may be very long, it is sometimes desirable for a client to search for a particular Standard Payee. A client may use the Standard Payee Inquiry <StdPayeeInqRq> message for this purpose. If the client finds a biller that the customer is interested in adding to the payee list, the client may add it simply by using the Customer Payee Add <CustPayeeAddRq> message with the Standard Payee's Identifier <StdPayeeId>.

7.1.2 Payments

7.1.2.1 Funding Accounts

The Pay Service uses the <DepAcctIdFrom> and <CardAcctIdFrom> aggregates to identify accounts as funding accounts for payment. When a customer schedules a Payment or creates a Recurring Payment Model, he or she specifies which account should be used as the funding account for that Payment, whether or not it is the default funding account.

A user may activate multiple funding accounts for paying bills and making other payments. If a single funding account is specified during Service Activation, then the User Interface may use that funding account as a default account. The funding account must be specified in <PmtInfo> for each individual payment message .

7.1.2.2 Payment Processing Flow

When a customer decides to make a payment, several dates are important. The date he or she transmits the request to the server is the payment set-up date. The customer may, depending on the Pay Service Profile, either supply the payment processing date <PrcDt> or payment due date <DueDt> in the <PmtInfo> aggregate in the request.

The processing date is the date by which the service must begin processing it for payment. In the case where a customer's CPP actually provides payment support directly, the processing date is the date by which the CPP must retrieve the payment request from its warehouse, generate remittance data, and possibly initiate funds transfers. The processing date is not necessarily the same as the date the funds are withdrawn from the customer's funding account. The actual funds withdrawal date varies depending on who actually processes the funds transfer requests.

The due date is the date the customer's payment needs to be in the hands of the payee. The customer establishes the date when he or she scheduled the payment. If supported by the CSP model, the customer may enter the due date for a payment and the service must calculate the processing date.

A customer processing date may be adjusted by a service for FI holidays and other non-processing days. A processing date may be adjusted either forward or backward depending on the Service Profile. Once the original processing date is adjusted, it is known as the adjusted processing date and must be the actual processing date.

If a user interface or client wishes to allow a customer to enter a due date even though the CSP model is processing date, the client must calculate a processing date for the customer. To aid in that calculation, for each biller payee, <DaysToPay> is provided in <StdPayeeRec>. <DaysToPay> is the number of business days it takes for the CPP to deliver the payment to the payee. This data is provided by the CPP and is returned when the client uses a Standard Payee Inquiry to obtain the Standard Payees. For fully specified payees and transfer payees, the <DfltDaysToPay> (for Fully-Specified payees) and <DfltXferDaysToPay> (for Transfer Payees) are provided in the Service Profile.

To be strictly accurate, a client would also need to know the SPs' holidays, using <HolInqRs>, and non-processing day schedule, via Service Profile <PrcDaysOff>, as well. But in general, the client may subtract the number of business days it would take for payment to arrive (<DaysToPay>, <DfltDaysToPay> or <DfltXferDaysToPay> depending on payee type) from the entered due date, and use that as the processing date. Business day is another term for processing day (the inverse of the <PrcDaysOff> for an FI or its service provider). If the FI holidays were available, the client may also check for and adjust for holidays.

Once a payment is processed, two deliveries take place: remittance data delivery and the funds delivery. Remittance data is information provided to payees about the payment, such as the customer's name, address, the payment amount, and account number with the payee <BillingAcct>. The delivery mechanism for this data may be electronic, such as using the Remittance Add <RemitAddRq> message or another electronic delivery channel, or paper, and may be separate from the actual delivery of the funds. The funds delivery consists of a withdrawal from the customer's funding account and a credit or deposit to the payee's account or a designated account at a BPP for that payee.

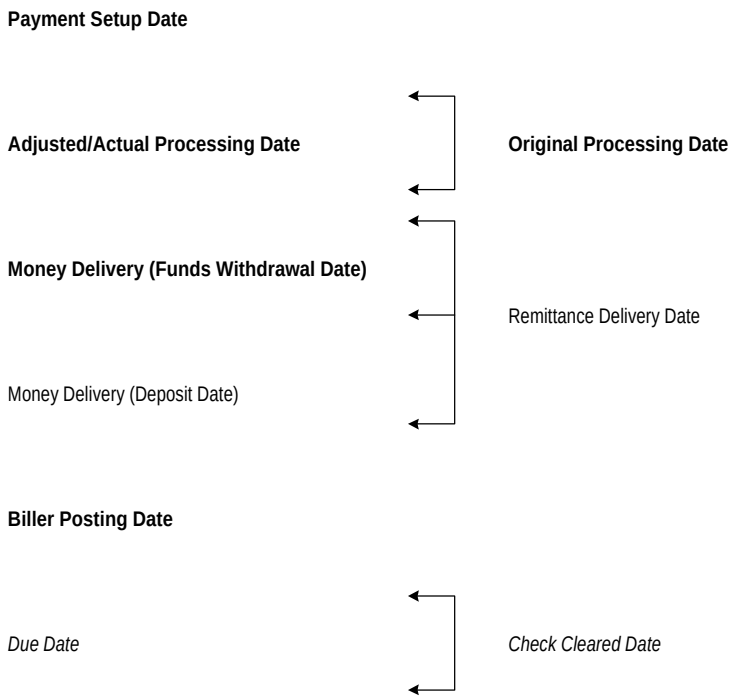
A payee may post a payment to the customer's account <BillingAcct> either when they receive the remittance data or when they receive the funds, at the discretion of the payee. The date that posting occurs is called the Biller Posting Date and is reflected in IFX as the Effective Date <EffDt> associated with the Payment Status of *Posted*.

The Biller Posting Date should ideally be earlier than or equal to the Due Date, although this is rarely guaranteed by a CPP due to variations in postal service and billers' accounts receivables systems.

Billers may consider the payment to be "on-time" if it is received by the Due Date, even if they require additional days to process and actually post the payment. If the billing cycle occurs prior to posting, such that the next bill is received without reflecting the prior payment, a customer care issue may arise. In order to facilitate customer care at the CSP/CPP, the biller may provide information to their BSP about the number of days they require to post an electronic payment after it is received <DaysToEPost>. The CSP may then retrieve this data from the BSP, using a Biller Inquiry. <DaysToEPost> is returned within <BillerRec>.

If the payee was paid with a draft drawn on the customer's actual funding account, then the funds withdrawal date mentioned above would be the date the debit cleared the customer's account. However, if the drafts are drawn on an FI consolidation account or if the payee is being paid electronically, the funds may be withdrawn (to go into the consolidation account) at the bank's discretion. This typically takes place once the CPP receives the funds transfer requests generated during payment processing. How a CPP's customers know when their funds are actually withdrawn is typically up to the CPP to disclose when the customer is signing up for the service.

Payment Processing Flow



The diagram illustrates the flow of a payment using its significant event dates as briefly introduced in the earlier text. The dates on the right hand side represent floating events. For example, the Remittance Delivery may occur before Funds Withdrawal, before Deposit to the payee, or after the deposit to the payee.

In the above diagram, the dates in bold are dates the customer usually cares about. In the service provider scenario, the customer cares about the Processing Date or Adjusted Processing Date because it may be the only date the service provider may supply about a processed/completed payment. Typically, service providers firmly know the Payment Setup Date, Processing Date, and Adjusted Processing Date.

The other dates in the chart are usually only known by a service provider if the actual money or remittance delivery organization (such as the customer's FI) notified the service provider of them. The same is also true for Biller Posting Date; i.e., a service provider would only know this date if it was supplied by the payees receiving payments and posting them to customer accounts <BillingAcct>.

Due Date and Check Cleared Date are in italics to show they may or may not be of importance to the customer. Due Date may be of importance if the customer enters or is more comfortable with the Due Date model. Check Cleared Date is only of importance if the payment was made via paper check and if the customer is in dispute with their payee about proper credit being given them for a payment.

7.1.2.3 Payment Invoices for Businesses

The IFX Service allows a customer to indicate which invoices are included in a payment, including line item detail. A customer may create a payment that pays one or more invoices or may choose to pay specific line items from one or more invoices.

7.1.2.4 Recurring Payments

By specifying the normal parameters associated with a Payment (funding account, payee, amount, date, etc.) and adding a Recurring Message Detail aggregate, a client may define a Recurring Payment Model. The

Recurring Payment Model may be either closed-ended (with a specified total number of payments or last payment date) or open-ended (continue until canceled).

Once a customer has defined Recurring Payment Models, the Pay provider must automatically generate payments based on those models at whatever frequency is specified in the Recurring Message Detail aggregate.

7.1.2.5 Payment Life Cycle and Payment Status

The IFX Specification contains a payment status reporting aggregate <PmtStatus> that is based on a typical payment life cycle. This life cycle may be described as follows:

A payment is *Scheduled*, either by a customer using the Payment Add Request message or by the Pay provider using one of the customer's Recurring Payment Models. Once scheduled, a customer may cancel a payment using the Payment Cancel Request message up until the time that payment is submitted for processing, resulting in a <PmtStatusCode> of *Cancelled*. A customer may request that a payment that has been generated from a recurring payment model be skipped (<PmtStatusCode> of *Skip*).

Payment processing has a number of possible outcomes. A payment may be *processed* with a successful result. A *Processed* payment may be *Returned*. The payment may be posted, rejected or refused by the BPP or Biller, indicated by <PmtStatusCode> values of *Posted*, *RemitRejected*, or *RemitRefused*, respectively. Those Pay providers that request a Payment Authorization from the customer's Financial Institution before payment may indicate additional payment states, such as *PmtAuthRejected* or *PmtAuthHeld*.

Since a Pay provider may not have access to the customer's bank account data, its records may show *Processed* as the final state for a successful payment. In the case where the Pay provider issues a paper check to a payee on behalf of the customer, the Pay provider may indicate to the customer the check has cleared (<PmtStatusCode> *CheckCleared*).

7.1.2.6 Additional Payment Information

Several elements are provided within the payment instruction to communicate other payment-related information back to the payee. These include the <BillRefInfo> and <Memo> elements, the <InvoiceInfo> aggregate, and the <PmtSummAmt> aggregate.

The <BillRefInfo> element is biller-defined text from the bill summary that may be sent back with the payment to assist in the biller's accounts receivable reconciliation. This may be equivalent to scanline information included with a biller's paper remittance stub. Note that some payment processing systems currently handle a maximum of 22 characters for this field, so an implementation must consider data truncation of larger inputs.

The <Memo> element may be used by CPPs that convert electronic payment instructions into paper checks.

The <InvoiceInfo> aggregate, like the <Memo> element, is information that may be delivered to the payee, either with the payment or through some other means, including email or as a separate paper item. This element may be used to allow a small business to identify, along with a payment, which invoices this payment "pays" and may include other characteristics such as discounts, credit adjustments, and/or free-form text.

The <PmtSummAmt> aggregate is used to indicate additional information about the total payment amount, as specified by the <CurAmt> in <RemitInfo>. It is comprised of the <BillSummAmtId> and the associated currency amount <CurAmt>. The <BillSummAmtId> is obtained from the <BillInfo> on a presented bill and is a tag that identifies the payment category to the biller. This element is used to indicate which payment amount is being paid when multiple payable amounts were presented, or to allocate a supplemental payment to one or more categories, or to designate a breakdown of the total amount into sub-amount categories. Examples of how this data may be presented and used may be found in the text above the <BillInfo> aggregate (see Section 8.3.3.1).

7.2 Pay Service Message Summary

Message / Message Name	Req.	Comments

Message / Message Name	*Req.*	*Comments*
Standard Payee Inquiry **<StdPayeeInqRq>** **<StdPayeeInqRs>**		Allows a client to inquire about payees that are well known to the CPP, which may be used to assist customers in selecting payees to be added to the customer's personal payee list <CustPayeeId>.
Customer Payee Add **<CustPayeeAddRq>** **<CustPayeeAddRs>**		Allows a client to create a new Payee record on the customer's Payee List. This Payee may be one of four types: a Biller Payee specified using <StdPayeeId>, a Fully-Specified Payee using the <FSPayee> aggregate, or an Interbank Transfer Payee using the <XferPayee> aggregate.
Customer Payee Modify **<CustPayeeModRq>** **<CustPayeeModRs>**		Allows a client to modify an existing Payee record on a customer's Payee List.
Customer Payee Type Modify **<CustPayeeTypeModRq>** **<CustPayeeTypeModRs>**		Allows a CSP to request a change in a Customer's Payee Type.
Customer Payee Delete **<CustPayeeDelRq>** **<CustPayeeDelRs>**		Allows a client to delete an existing Payee record on a customer's Payee List.
Customer Payee Inquiry **<CustPayeeInqRq>** **<CustPayeeInqRs>**		Allows a client to view a summary of current Payee records associated with the current customer. (Payee List)
Customer Payee Audit **<CustPayeeAudRq>** **<CustPayeeAudRs>**		Allows client to play back the Customer Payee messages associated with the current customer since some past point in time.
Customer Payee Synchronization **<CustPayeeSyncRq>** **<CustPayeeSyncRs>**		Allows client to play back the Customer Payee messages associated with the current customer since some past point in time.
Payment Add **<PmtAddRq>** **<PmtAddRs>**	Yes	Allows a client to create a new Payment associated with the current customer. The Payee is automatically added to the customer's Payee List if it is not already there.
Payment Modify **<PmtModRq>** **<PmtModRs>**		Allows a client to modify pending Payment records.
Payment Status Modify **<PmtStatusModRq>** **<PmtStatusModRs>**		Allows an SP to request a modification in the status of a Payment.
Payment Cancel **<PmtCanRq>** **<PmtCanRs>**	Yes	Allows a client to cancel pending Payment records.
Payment Inquiry **<PmtInqRq>** **<PmtInqRs>**	Yes	Allows a client to view pending Payment records.
Payment Audit **<PmtAudRq>** **<PmtAudRs>**		Allows a client to play back the Payment messages associated with the current customer since some past point in time. Note that an audit must return payment messages for both pending and completed payments.

Message / Message Name	*Req.*	*Comments*
Payment Synchronization **<PmtSyncRq>** **<PmtSyncRs>**		Allows a client to play back the Payment messages associated with the current customer since some past point in time. Note that a sync must return payment messages for both pending and completed payments.
Payment Authorization Add **<PmtAuthAddRq>** **<PmtAuthAddRs>**		Allows a client to request a new Payment Authorization.
Payment Authorization Modify **<PmtAuthModRq>** **<PmtAuthModRs>**		Allows a client to modify existing Payment Authorization records.
Payment Authorization Cancel **<PmtAuthCanRq>** **<PmtAuthCanRs>**	Yes	Allows a client to cancel existing Payment Authorization records.
Payment Authorization Inquiry **<PmtAuthInqRq>** **<PmtAuthInqRs>**	Yes	Allows a client to view existing Payment Authorization records.
Payment Authorization Audit **<PmtAuthAudRq>** **<PmtAuthAudRs>**		Allows a client to play back the Payment Authorization messages since some past point in time. Note that an audit must return payment messages for both pending and completed authorizations.
Payment Authorization Synchronization **<PmtAuthSyncRq>** **<PmtAuthSyncRs>**		Allows a client to play back the Payment Authorization messages since some past point in time. Note that a synchronization must return payment messages for both pending and completed authorizations.
Remittance Add **<RemitAddRq>** **<RemittAddRs>**		Allows a client to create a new Remittance object.
Remittance Modify **<RemitModRq>** **<RemitModRs>**		Allows a client to modify existing Remittance records.
Remittance Status Modify **<RemitStatusModRq>** **<RemitStatusModRs>**		Allows an SP to request a modification in the status of a Remittance Advice.
Remittance Delete **<RemitDelRq>** **<RemitDelRs>**		Allows a client to delete existing Remittance records.
Remittance Inquiry **<RemitInqRq>** **<RemitInqRs>**		Allows a client to view existing Remittance records.
Remittance Audit **<RemitAudRq>** **<RemitAudRs>**		Allows a client to play back the Remittance messages since some past point in time. Note that an audit must return payment messages for both pending and completed remittances.
Remittance Synchronization **<RemitSyncRq>** **<RemitSyncRs>**		Allows a client to play back the Payment messages since some past point in time. Note that a synchronization must return payment messages for both pending and completed remittances.

Message / Message Name	*Req.*	*Comments*
Recurring Payment Model Add **<RecPmtAddRq>** **<RecPmtAddRs>**		Allows a client to create a new Recurring Payment Model associated with the current customer. The first Payment may be generated when the Recurring Payment Model is added. The Payee is automatically added to the customer's Payee List if it is not already there.
Recurring Payment Model Modify **<RecPmtModRq>** **<RecPmtModRs>**		Allows a client to modify a Recurring Payment Model associated with the current customer. Pending payments generated from the modified Recurring Payment Model may also be modified if supported by the Pay Provider.
Recurring Payment Model Cancel **<RecPmtCanRq>** **<RecPmtCanRs>**		Allows a client to delete a Recurring Payment Model associated with the current customer. Pending payments generated from the deleted Recurring Payment Model may also be deleted.
Recurring Payment Model Inquiry **<RecPmtInqRq>** **<RecPmtInqRs>**		Allows a client to view a summary of Recurring Payment Models associated with the current customer.
Recurring Payment Model Audit **<RecPmtAudRq>** **<RecPmtAudRs>**		Allows a client to play back the Recurring Payment Model messages associated with the current customer since some past point in time.
Recurring Payment Model Synchronization **<RecPmtSyncRq>** **<RecPmtSyncRs>**		Allows a client to play back the Recurring Payment Model messages associated with the current customer since some past point in time.

7.3 Pay Service Common Aggregates

7.3.1 Standard Payee Record <StdPayeeRec>

Tag	*Type*	*Usage*	*Description*
<StdPayeeId>	Aggregate	Required	Customer's Payee Identifier. The Standard Payee as known to the CPP.
<StdPayeeInfo>	Aggregate	Required	Standard Payee Information Aggregate

7.3.1.1 Standard Payee ID <StdPayeeId>

The <StdPayeeId> aggregate is used to describe a *Predefined Payee*, such as a Biller. The <StdPayeeId> is the Biller as known by the CPP. <StdPayeeId> is a synonym of <BillerId>, which is the Biller identification as known by the Biller or BSP.

Tag	*Type*	*Usage*	*Description*
<SPName>	Identifier	Required	Service Provider Name. Used to qualify <BillerNum>. This is the name of the organization that assigned <BillerNum>, i.e., the CPP (or agent of the CPP).
<BillerNum>	Identifier	Required	Biller Number. Assigned by the CPP.

7.3.1.2 Standard Payee Information <StdPayeeInfo>

Tag	Type	Usage	Description
<Name>	C-40	Optional	Payee Name.
<PostAddr>	Aggregate	Optional	Standard Payee Address Aggregate.
<AcctMask>	C-32	Optional Repeating	Account Mask. Zero or more edit masks to facilitate <BillingAcct> entry and editing.
<IndustId>	Aggregate	Optional	Industry Identifier
<DaysToPay>	Long	Required	Days to Pay. Minimum number of business days needed to process. Assigned by the Pay provider. Cannot be changed by the client.

7.3.2 Customer Payee Record Aggregate <CustPayeeRec>

Tag	Type	Usage	Description
<CustPayeeId>	Identifier	Required	Customer's Payee Identifier. If the server supports Payee Lists, the server assigns <CustPayeeId> at the time the Payee is first added.
<CustPayeeInfo>	Aggregate	Required	New Payee Information. This is an instance of <CustPayeeInfo> that provides information about the payee after any adjustment has been made
<BillerContact>	Aggregate	Optional	Biller Contact Information Aggregate.
<DaysToPay>	Long	Required	Days to pay. Number of days required to complete the payment.

7.3.2.1 Customer Payee Information Aggregate <CustPayeeInfo>

The <CustPayeeInfo> aggregate is used in most messages related to Payees and is optionally used in <PmtAddRq> and <RecPmtAddRq> if a new Payment or Recurring Payment Model is being added for a new Customer Payee.

> ***Note:*** *name is optional below. The following table provides guidelines for including <Name> within implementations.*

Payee Type	Usage	Explanation
Fully Specified Payee	Required	Required in order to issue the check.
Transfer Payee	Recommended	Recommended for problem resolution.
Standard Payee	Recommended	Recommended for problem resolution.
Biller Payee	Required	Required, since it is always returned as part of <BillerRec>.

Tag	Type	Usage	Description

Tag	Type	Usage	Description
<Name>	C-40	Optional *but see Description*	Payee Name. Initially assigned by the customer when the Payee is added. May be modified by the client. Should contain Biller Name <BusName> in the case of a Biller. Does not affect the behavior of interbank transfer payments, but recommended usage is destination account name for interbank transfer payees.
<Nickname>	C-40	Optional	Payee Nickname. Optionally assigned by the customer. Server always echoes it in immediate response if received in a request. Pay provider may indicate support for longer-term storage of nicknames in the Service Profile. May be modified by the client.
<StdPayeeId>	Aggregate	Required XOR	Standard Payee Identification Aggregate.
<FSPayee>	Aggregate	Required XOR	Fully-Specified Payee Aggregate.
<XferPayee>	Aggregate	Required XOR	Transfer Payee Aggregate.
<BillerPayee>	Aggregate	Required XOR	Biller Payee Aggregate.
<BillingAcct>	C-32	Optional	Customer Account Number with Payee. This is the customer's identification with this Payee. For Transfer Payees, the account number for the target account is carried in the <XferPayee> aggegate, rather than in this element.
<DfltPmtInfo>	Aggregate	Optional	Default Payment Information Aggregate. Used to add default payment information when adding a new payee. This would not be included when paying a bill unless the user was also adding a new payee at the same time.
<DepAcctIdFrom>	Aggregate	Optional XOR Profiled support	Deposit Account Aggregate. The default funding account must be used when the Payment definition does not specify the funding account. The client may modify the default funding account. Either <DepAcctIdFrom> or <CardAcctIdFrom> may be specified. A default funding account for bill presentment may be specified during service activation when used within <PmtInfo>.
<CardAcctIdFrom>	Aggregate	Optional XOR Profiled support	Credit Account Aggregate. Used when the default funding account is a Credit Card account.
<Category>	C-40	Optional	Default Category for Payments. May be modified by the client.
<Memo>	C-255	Optional	Default Memo for Payments. May be modified by the client.
</DfltPmtInfo>			

7.3.2.2 Fully-Specified Payee Aggregate <FSPayee>

The <FSPayee> aggregate is used to provide complete information about a Fully-Specified Payee.

Tag	Type	Usage	Description
<PostAddr>	Aggregate	Required	Payee Address Aggregate.

Tag	Type	Usage	Description
<OrgContact>	Aggregate	Required	Payee contact information.

7.3.2.3 Biller Payee <BillerPayee>

The <BillerPayee> aggregate is used to provide information for payment to a Biller.

Tag	Type	Usage	Description
<BillerId>	Aggregate	Required	Biller Identification Aggregate.
<BillerContact>	Aggregate	Optional	Biller Contact Information.
<HistRetentionDays>	Long	Optional	Number of days that Bill Summary and Bill Detail information is available for inquiries. The Bill Detail information may be available for a longer period of time.
<BillerPayInfo>	Aggregate	Optional	Biller Pay Information Aggregate.

7.3.2.4 Transfer Payee Aggregate <XferPayee>

The <XferPayee> aggregate is used to provide complete information about a Transfer Payee.

Tag	Type	Usage	Description
<BankInfo>	Aggregate	Optional	Bank Information.
<DepAcctIdTo>	Aggregate	Required	Deposit Account Aggregate. In this context, <DepAcctIdTo> specifies the target account for the transfer.

7.3.3 Payment Record Aggregate <PmtRec>

Tag	Type	Usage	Description
<PmtId>	Identifier	Required	Payment Identifier. Assigned by the server at the time the Payment is first added.
<RecPmtId>	Identifier	Optional *but see Description*	Recurring Payment Model Identifier. Assigned by the server at the time the Recurring Payment Model is first added. Returned if payment was generated from a Recurring Payment Model.
<PmtInfo>	Aggregate	Required	Payment Record Aggregate.
<PmtStatus>	Aggregate	Required	Payment Status Aggregate.
<PmtRemitName>	C-40	Optional	Payment Remittance Name. The name of the organization to which payments are remitted.
<PmtRemitAddr>	Aggregate	Optional	Payment Remittance Address Aggregate.
<CreatedDt>	Date	Optional	Message Date. Date payment was actually processed and entered history.
<OrigPmtPrcDt>	Date	Optional XOR	Original Payment Processing Date.
<OrigPmtDueDt>	Date	Optional XOR	Original Payment Due Date.
<FIDebitTrcNum>	NC-7	Optional	FI Debit Trace Number.

Tag	Type	Usage	Description
<FICreditTrcNum>	NC-7	Optional	FI Credit Trace Number.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier. This is the <CSPRefId> of the message that created or last modified the payment.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier. This is the <SPRefId> of the message that created or last modified the payment.
<ClientApp>	Aggregate	Optional	Client Application Aggregate.

7.3.3.1 Payment Information Aggregate <PmtInfo>

The <PmtInfo> aggregate is used in most messages related to Payments and Recurring Payments.

Tag	Type	Usage	Description
<RemitInfo>	Aggregate	Required	Remittance Information Aggregate.
<DepAcctIdFrom>	Aggregate	Required XOR	Deposit Account Aggregate. The funding account for this payment.
<CardAcctIdFrom>	Aggregate	Required XOR	Credit Account Aggregate. Used for credit card payments and transfers.
<Category>	C-40	Optional	Payment Category. The customer assigns categories.
<PrcDt>	Date	Required XOR Profiled support	Payment Processing Date. The customer enters this date as the original requested processing date, not corrected for holidays and non-processing days. May be modified by the customer.
<DueDt>	Date	Required XOR Profiled support	Payment Due Date. The customer enters this date to represent the date the payment is due to arrive at the payee. May be modified by the customer.
<ImmediatePmt>	Boolean	Optional Profiled support	Immediate Payment Indicator. If set to TRUE, the payment must be executed immediately, and not at end of day. Subject to support in Service Profile.

7.3.3.2 Payment Status Aggregate <PmtStatus>

The <PmtStatus> is returned in responses to Add or Modify Payment and Add or Modify Recurring Payment Model. Note that all elements within this aggregate are assigned by the server and cannot be assigned or modified by the client.

Tag	Type	Usage	Description

Tag	*Type*	*Usage*	*Description*
<PmtStatusCode>	Closed Enum	Required	Payment Status Code. This identifies the payment processing status. Valid Values are: • Scheduled – The payment has been scheduled. • Processed – The payment has been processed by the SP. • FundAcctDebited – The funding account has been debited for this payment. • Skip – The client has requested that this payment, which was generated from a recurring payment model, be skipped. • Cancelled – The payment has been cancelled either by a <PmtCanRq> or as a cascade delete from a <CustPayeeDelRq> or a <RecPmtDelRq>. • Failed – The payment processing failed and no additional information is available. May include data integrity problems. • PmtAuthHeld – A payment authorization has been requested & resulted in the payment being held by the FI, typically for the next processing day. • PmtAuthNoFunds – A payment authorization has been requested, but was rejected by the FI due to insufficeint funds in the funding account. • PmtAuthInactive – A payment authorization has been requested, but was rejected due to the funding account being inactive. • PmtAuthClosed – A payment authorization has been requested, but was rejected due to the funding account being closed. • Posted – The payment has been posted by the Biller. • CheckCleared – The check used to remit the funds to the biller has cleared the account on which it is drawn. • Returned – The payment has been returned due to insufficient funds. • RemitPending – The remittance has been sent to the Biller or BPP and is pending processing. • RemitRefused – The Remittance has been refused by the Biller. They located the Billing Account but cannot accept the payment as remitted. • RemitRejected – The Biller has rejected the Remittance. This may happen if the billing account could not be located.
<EffDt>	Date	Required	Payment Status Date. The date associated with the state change to the current state.
<PmtAuthId>	Identifier	Optional	Payment Authorization Identifier. If present, payment authorization was obtained for the payment and the authorizing SP assigned this identifier. If omitted, no authorization was obtained.

Tag	Type	Usage	Description
<RemitId>	Identifier	Optional	Remittance Idenitifier. If present, indicates the identifier of the remittance advice associated with this payment. May apply for <PmtStatusCode> of Posted, RemitPending, RemitRejected, or RemitRefused
<RecPmtMod>	Boolean	Optional	Recurring Payment Modified Indicator. Required to be TRUE if a <PmtModRq> has subsequently modified a payment generated from a Recurring Payment Model so that it no longer matches the Recurring Payment Model. This may only be supplied for recurring payment instances.
<PmtAuthCount>	Long	Optional	Payment Authorization Count. For Pay providers that require an authorization from the customer's CPP. This is the number of times that the payment has been submitted for authorization.
<PmtMethod>	Open Enum	Optional	Payment Method. Indicates how the payment was transmitted to the payee; e.g., via check or electronically.
<ChkNum>	NC-12	Optional	Check Number. Assigned by the Pay provider or the CPP if payment is by paper check.
<Memo>	C-255	Optional	Memo. Intended to be optionally used by a CPP to store any necessary text associated with the payment.

7.3.3.3 Recurring Payment Record <RecPmtRec>

Tag	Type	Usage	Description
<RecPmtId>	Identifier	Required	Recurring Payment Model Identifier. Assigned by the server at the time the Recurring Payment Model is first added.
<PmtInfo>	Aggregate	Required	Payment Record Aggregate.
<RecModelInfo>	Aggregate	Required	Recurring Model Information Aggregate.
<RemainingInsts>	Long	Required	Remaining Instance Count. The server must calculate this number as the number of actual payments to be made plus the number of instances to skip based on the customer-entered <RecSeriesEnd>. Server must calculate on <RecPmtAddRq> and return in response. Server must recalculate in case of an <RecPmtModRq> that changes <RecSeriesEnd> or when a payment is spawned. The server is not required to generate a <RecPmtModRs> on each change to <RemainingInsts>.

7.3.4 Payment Authorization Record Aggregate <PmtAuthRec>

Tag	Type	Usage	Description
<PmtAuthId>	Identifier	Required	Payment Authorization Identifier
<PmtAuthInfo>	Aggregate	Required	Payment Authorization Information Aggregate.
<PmtAuthStatus>	Aggregate	Required	Payment Authorization Status Aggregate

7.3.4.1 Payment Authorization Information Aggregate <PmtAuthInfo>

Tag	Type	Usage	Description
<DepAcctIdFrom>	Aggregate	Required XOR	Deposit Account Aggregate. The funding account for this payment.
<CardAcctIdFrom>	Aggregate	Required XOR	Credit Account Aggregate. Used for credit card payments and transfers. A funding account must be specified.
<CurAmt>	Aggregate	Required	Currency Amount. The amount of funds being requested for a payment authorization.
<Name>	C-80	Optional	Payee name.

7.3.4.2 Payment Authorization Status Aggregate <PmtAuthStatus>

Tag	Type	Usage	Description
<PmtAuthStatusCode>	Open Enum	Required	Payment Authorization Status Code Defined values: Authorized, Cleared
<EffDt>	DateTime	Required	Effective date of this status
<StatusModBy>	Open Enum	Optional	Status modified by.

7.3.5 Remittance Record Aggregate <RemitRec>

Tag	Type	Usage	Description
<RemitId>	Identifier	Required	Remittance Identifier. Assigned by the server at the time the Remittance is first added.
<RemitInfo>	Aggregate	Required	Remittance Record Aggregate.
<RemitStatus>	Aggregate	Required	Remittance Status Aggregate.

7.3.5.1 Remittance Information Aggregate <RemitInfo>

The <RemitInfo> aggregate is used in most messages related to Remittance of Payments and Recurring Payments. <RemitInfo> contains the elements that pertain to remittance.

Tag	Type	Usage	Description
<CustPayeeId>	Identifier	Required XOR	Customer's Payee Identifier. If the server supports Payee Lists, the server assigns <CustPayeeId> at the time the Payee is first added.
<CustPayeeInfo>	Aggregate	Required XOR	Payee Information Aggregate. Must be used if <CustPayeeId> is unassigned.
<BillRefInfo>	NC-80	Optional	Biller Reference Information. Biller-defined text from <BillRec>, for the biller's Accounts Receivable reconciliation. Depending on the Payment Provider and the payment network and message used to make the payment, <BillRefInfo> may be truncated to 22 characters.
<BillId>	UUID	Optional	Bill Identifier. Bill Service Provider defined universally-unique identifier, to be provided if the payment is related to an IFX-presented bill.

Tag	*Type*	*Usage*	*Description*
<PmtId>	Identifier	Optional	Payment Identifier. The identifier of the payment associated with this remittance. This is included to enable the BPP to inform the CPP of payment posting against this remittance.
<Memo>	C-255	Optional	Memo for Payment. From Customer to Payee. Depending on the Payment Provider and the payment network and message used to make the payment, <Memo> may be truncated or dropped.
<BillingAcct>	C-32	Optional *but see Description*	Customer Account Number with Payee. Required if the Service Provider does not support <CustPayeeId>. Inclusion here must not modify the <BillingAcct> at the CPP.
<CurAmt>	Currency Amount	Required	Currency Amount. The total amount being paid.
<PmtSummAmt>	Aggregate	Optional Repeating	Payment Summary Amount. Used to designate the identifier for the amount being paid, or to specify the allocation of the total amount being paid or a portion of the total amount.
<BillSummAmtId>	Identifier	Required	Bill Summary Amount Identifier. Biller's identifier for this currency amount, which is obtained in the <BillInfo> for a presented bill. The consumer can return this in the payment messages to identify to the biller the type of amount being paid or allocated by the consumer.
<CurAmt>	Currency Amount	Required	Currency Amount. Used to specify the amount being associated with the identifier specified by <BillSummAmtId>. This may be the same as the total amount being paid or a portion of the total amount (such as a supplemental payment amount or sub-amount allocation).
</PmtSummAmt>			
<InvoiceInfo>	Aggregate	Optional Repeating	Invoice Information Aggregate. Provides information about which Invoices and optionally Line Items are paid by this payment. The amount of the payment is always <CurAmt> above, which may or may not match the total of invoices or line items listed here.

7.3.5.2 Remittance Status Aggregate <RemitStatus>

Tag	*Type*	*Usage*	*Description*
<RemitStatusCode>	Closed Enum	Required	Remit Status Code. This identifies the payment processing status. Valid values are: • Pending • Posted • Refused • Rejected • Returned
<EffDt>	Date	Required	Processing Status Date. The date associated with the state change to the current state.
<Memo>	C-255	Optional	Memo. Intended to be optionally used by a BPP to store any necessary text associated with the remittance status.

7.3.6 Invoice Information Aggregate <InvoiceInfo>

Tag	*Type*	*Usage*	*Description*
<InvoiceNum>	C-32	Required	Invoice Number. The client enters this identifier since it does not serve as a database key on the server.
<TotalCurAmt>	Currency Amount	Required	Invoice Due Currency Amount.
<PaidCurAmt>	Currency Amount	Optional	Invoice Paid Currency Amount.
<EffDt>	Date	Required	Invoice Date.
<Desc>	C-80	Optional	Invoice Description.
<InvoiceVouchNum>	C-80	Optional	Invoice Voucher Number. The invoice recipient's internal voucher number confirming receipt of the goods listed on the invoice.
<LateFeeCurAmt>	Currency Amount	Optional	Invoice Late Fee Amount
<InvoicePremium>	Decimal	Optional	Invoice Premium.
<Discount>	Aggregate	Optional	Discount Information Aggregate.
<Rate>	Decimal	Required XOR	Discount Percentage.
<CurAmt>	Currency Amount	Required XOR	Discount Currency Amount.
<EffDt>	Date	Optional	Discount Date.
<Desc>	C-80	Optional	Discount Description.
</Discount>			
<InvoiceAdj>	Aggregate	Optional	Adjustment Aggregate.
<InvoiceAdjNum>	C-32	Optional	Adjustment Number.
<CurAmt>	Currency Amount	Required	Adjustment Currency Amount.
<EffDt>	Date	Optional	Adjustment Date.
<Desc>	C-80	Optional	Adjustment Description.
</InvoiceAdj>			
<InvoiceLineItem>	Aggregate	Optional Repeating	Invoice Line Item Aggregate.
<InvoiceLineItemNum>	C-12	Required	Line Item Number
<CurAmt>	Currency Amount	Required	Line Item Amount.
<Desc>	C-80	Optional	Line Item Description.
</InvoiceLineItem>			

7.4 Standard Payee

7.4.1 Standard Payee Inquiry

The Standard Payee Inquiry message may be used to assist customers in selecting payees to be added to the customer's personal payee list <CustPayeeId>. The response returns one or more well-known payees that match the selection criteria of the request.

7.4.1.1 Request <StdPayeeInqRq>

If the client omits all search elements in the <StdPayeeInqRq>, the client is requesting a complete directory of payees. Otherwise, the client wants to filter results based on the included elements.

For each payee that matches the elements in the request, the CPP returns the complete name and address of the payee, the payee ID, and the CPP name.

This inquiry request is similar to the <BillerInqRq>, which returns billers serviced by a BSP. Note that billers known by a <BillerId> to a BSP may be well-known merchants to a CPP and therefore will have a corresponding <StdPayeeId>. In fact, an entity playing both BSP and CSP roles may use the same actual value, but implementers must never count on this.

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<StdPayeeId>	Aggregate	Optional Repeating	Payee Identification. This field is used as a selection criterion.
<Name>	C-40	Optional Repeating	Payee Name. This field is used as a selection criterion.
<BillingAcct>	Aggregate	Optional Repeating	Customer (Billing) Account Number with the Payee. Useful for determining appropriate merchant instance via account scheme.
<PostalCode>	C-11	Optional Repeating	Postal Code. This is the postal code of the billing account. This field is used as a selection criterion. (1) It may used to limit the search to billers doing business within a limited geography, or (2) It may be used to identify the correct legal/entity or remittance address such as the cable company for a specific city, e.g. TCI—Sunnyvale CA.
<IndustId>	Aggregate	Optional Repeating	Industry Identifier

7.4.1.2 Response <StdPayeeInqRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<StdPayeeId>	Aggregate	Optional Repeating Echoed	Payee Identification.

Tag	Type	Usage	Description
<Name>	C-40	Optional Repeating Echoed	Payee Name. This field is used as a selection criterion.
<BillingAcct>	Aggregate	Optional Repeating Echoed	Customer (Billing) Account Number with the Payee. Useful for determining appropriate merchant instance via account scheme.
<PostalCode>	C-11	Optional Repeating Echoed	Postal Code. This is the postal code of the billing account. This field is used as a selection criterion. (1) It may used to limit the search to billers doing business within a limited geography, or (2) It may be used to identify the correct legal/entity or remittance address such as the cable company for a specific city, e.g. TCI—Sunnyvale CA.
<IndustId>	Aggregate	Optional Repeating Echoed	Industry Identifier
<StdPayeeRec>	Aggregate	Optional Repeating	Standard Payee Record Aggregate. One record is returned for each Payee defined for this customer who meets the selection criteria in the request.

7.5 Customer Payee

7.5.1 Customer Payee Add

The client uses the Customer Payee Add message to add a Payee to a Customer Payee list on the Pay provider's server.

7.5.1.1 Request <CustPayeeAddRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustPayeeInfo>	Aggregate	Required	Payee Record Aggregate.

7.5.1.2 Response <CustPayeeAddRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustPayeeInfo>	Aggregate	Required Echoed	Payee Record Aggregate.
<CustPayeeRec>	Aggregate	Required	Customer Payee Record.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.5.2 Customer Payee Modify

The Modify Payee message allows a client to change information about a payee on a customer's Payee list on the Pay provider's server.

7.5.2.1 Request <CustPayeeModRq>

The <CustPayeeModRq> message is sent by a client to modify a Payee record on a customer's Payee list on the server.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustPayeeId>	Identifier	Required	Customer's Payee Identifier. If the server supports Payee Lists, the server assigns <CustPayeeId> at the time the Payee is first added.
<CustPayeeInfo>	Aggregate	Required	Payee Info Aggregate.
<ModPending>	Boolean	Optional Profiled support	Modify Pending Payments indicates whether to propagate the payee change to pending single payments. Changes to payees are always propagated to payment models. If TRUE, changes to payees are propagated to pending payments If FALSE, or omitted, changes are not propagated. This option must be ignored when <ModPendingType> ≠ IfRequested in the Pay Service Profile.

7.5.2.2 Response <CustPayeeModRs>

The server sends the <CustPayeeModRs> message in response to a <CustPayeeModRq> message from a client.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustPayeeId>	Identifier	Required Echoed	Customer's Payee Identifier.
<CustPayeeInfo>	Aggregate	Required Echoed	Payee Record Aggregate.
<ModPending>	Boolean	Optional Profiled support Echoed	Modify Pending Payments Flag.
<CustPayeeRec>	Aggregate	Required	Customer Payee Record.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.5.3 Customer Payee Type Modify

A user with special authorization, typically a CSR, may send a request to the Payment Service Provider to change the payee type.

7.5.3.1 Request <CustPayeeTypeModRq>

The user must specify security details, the payee identifier, the payee type, and the new payee type.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustPayeeId>	Identifier	Required	Customer Payee Identifier. Assigned by the server at the time the Payee is first added. Cannot be modified by the client.
<PayeeType>	Closed Enum	Required	Payee Type. May be Standard Payee, Biller, Fully-Specified Payee or Interbank Transfer. Valid values: Biller, FSPayee, Xfer, Std.

7.5.3.2 Response <CustPayeeTypeModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustPayeeId>	Aggregate	Required Echoed	Payee Identifier.
<PayeeType>	Closed Enum	Required Echoed	Payee Type.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.5.4 Customer Payee Delete

A client uses the Customer Payee Delete message to delete a Payee from a Customer Payee list on the server. If the <CascadeDel> element is not set to TRUE, an error must be returned if a payee delete is attempted when payments are still pending. If this error is returned, the types of dependent objects must be communicated to the client in the response message.

7.5.4.1 Request <CustPayeeDelRq>

A client initiates the Delete Payee message by sending a <CustPayeeDelRq> message to the server.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustPayeeId>	Identifier	Required	Customer's Payee Identifier. If the server supports Payee Lists, the server assigns <CustPayeeId> at the time the Payee is first added.
<CascadeDel>	Boolean	Optional	Cascade Delete. If TRUE, server must delete all dependent objects when this object is deleted. If FALSE or omitted, the customer payee must not be deleted if dependent objects exist. For a customer payee, a dependant objects are recurring models or payments.

7.5.4.2 Response <CustPayeeDelRs>

The server responds to the client's <CustPayeeDelRq> message by sending a <CustPayeeDelRs> message.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<CustPayeeId>	Identifier	Required Echoed	Customer's Payee Identifier.
<CascadeDel>	Boolean	Optional Echoed	Cascade Delete.
<DependentType>	Open Enum	Optional Repeating	An aggregate that would contain a list of depending object types that exist for the customer payee. Defined Values: RecPmt, Pmt
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.5.5 Customer Payee Inquiry

A client uses the Payee Inquiry message to obtain records from the Customer Payee list on a server. Clients that store a local copy of a Customer Payee list may use this message to "refresh" the Payee list.

7.5.5.1 Request <CustPayeeInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.

Tag	Type	Usage	Description
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<DeliveryMethod>	Open Enum	Optional Profiled values	Delivery Method. Used to request the delivery channel for requested information. Included here primarily to allow customer using an IVR (interactive voice response) unit to request a delivery option for their Payee list. See Data Dictionary for details. Default value is Channel
<CustPayeeId>	Identifier	Optional Repeating	Customer's Payee Identifier. This field is used as a selection criterion. If Payee Lists are supported by the server, <CustPayeeId> is assigned by the server at the time the Payee if first added.
<PayeeType>	Closed Enum	Optional Repeating	Payee Type. This field is used as a selection criterion.
<Name>	C-40	Optional Repeating	Payee Name. This field is used as a selection criterion.
<Nickname>	C-40	Optional Repeating	Payee Nickname. Optionally assigned by the customer. Only stored by the Pay provider if indicated in the Service Profile. This field is used as a selection criterion.
<IncToken>	Boolean	Optional	Include Token. If TRUE, a <Token> must be included in the response, if the Service Profile indicates support for customer payee synchronization, to set a base for future synchronization messages. If FALSE or omitted, the <Token> may be omitted in the response.

7.5.5.2 Response <CustPayeeInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<DeliveryMethod>	Open Enum	Optional Profiled values Echoed	Delivery Method.
<CustPayeeId>	Identifier	Optional Repeating Echoed	Customer's Payee Identifier.
<PayeeType>	Closed Enum	Optional Repeating Echoed	Payee Type. This field is used as a selection criterion.
<Name>	C-40	Optional Repeating Echoed	Payee Name.

Tag	Type	Usage	Description
<Nickname>	C-40	Optional Repeating Echoed	Payee Nickname.
<CustPayeeRec>	Aggregate	Optional Repeating	Payee List Record Aggregate. One record is returned for each Payee defined for this customer who meets the selection criteria in the request.
<Token>	Identifier	Optional *but see Description*	Token. Server Assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. The token must be provided if <IncToken> = TRUE in the request and the server supports synchronization. <Token> = 0 is returned if no records are returned within the response.

7.5.6 Customer Payee Audit

7.5.6.1 Request <CustPayeeAudRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional	Payee Method. This field is used as a selection criterion.
<CustPayeeId>	Identifier	Optional	Customer's Payee Identifier. This field is used as a selection criterion. If the server supports Payee Lists, the server assigns <CustPayeeId> at the time the Payee is first added.

7.5.6.2 Response <CustPayeeAudRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Message Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.

Tag	Type	Usage	Description
<Method>	Closed Enum	Optional Echoed	Payee Method. This field is used as a selection criterion.
<CustPayeeId>	Identifier	Optional Echoed	Customer's Payee Identifier. This field is used as a selection criterion. If the server supports Payee Lists, the server assigns <CustPayeeId> at the time the Payee is first added.
<CustPayeeMsgRec>	Aggregate	Optional Repeating	Payee Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the <CustPayeeMsgRec> request.
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<CustPayeeAddRs>	Aggregate	Required XOR	Customer Payee Add Response Message Aggregate.
<CustPayeeModRs>	Aggregate	Required XOR	Customer Payee Modify Response Message Aggregate.
<CustPayeeTypeModRs>	Aggregate	Required XOR	Customer Payee Type Modify Response Message Aggregate.
<CustPayeeDelRs>	Aggregate	Required XOR	Customer Payee Delete Response Message Aggregate.
</CustPayeeMsgRec>			

7.5.7 Customer Payee Synchronization

The Customer Payee Synchronization message is used by a client for retrieving a list of changes to a customer's Payee list. This message may be used by clients that keep local copies of a customer's Payee messages to synchronize their local databases of Payee messages against the databases of Payee messages kept by Pay providers. Typically results of this message tells a client what the customer has done using other clients since they last used this one, but it may also inform a client about changes the Pay provider made (e.g., converted a Payee from a Fully-Specified Payee to a Biller).

7.5.7.1 Request <CustPayeeSyncRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.

7.5.7.2 Response <CustPayeeSyncRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Message Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<Token>	Identifier	Required Echoed	Token.
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. If the client sent a token in the request, the server returns a new token based on this message.
<CustPayeMsgRec>	Aggregate	Optional Repeating	Customer Payee Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issued the <CustPayeeMsgRec> request.
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<CustPayeeAddRs>	Aggregate	Required XOR	Customer Payee Add Response Message Aggregate.
<CustPayeeModRs>	Aggregate	Required XOR	Customer Payee Modify Response Message Aggregate.
<CustPayeeTypeModRs>	Aggregate	Required XOR	Customer Payee Type Modify Response Message Aggregate.
<CustPayeeDelRs>	Aggregate	Required XOR	Customer Payee Delete Response Message Aggregate.
</CustPayeeMsgRec>			

7.6 Single Payment

7.6.1 Payment Add

The Payment Add message allows a client to schedule a single payment, where the amount is input by the customer or from a presented bill. The Payment Add message may reference an existing payee or add a new one, by specifying the information within <RemitInfo> <PayeeInfo>. If the Payment service provider supports <CustPayeeId>, the client must specify an existing <CustPayeeId> or include the <CustPayeeInfo> aggregate, but not both. Whether or not the Payment service provider supports <CustPayeeId>, the <CustPayeeInfo> aggregate may specify an existing standard payee or create a new fully specified or transfer payee. It is not possible to modify an existing payee within an Add Payment message. The customer may modify a payee via <CustPayeeModRq>.

7.6.1.1 Request <PmtAddRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtInfo>	Aggregate	Required	Payment Record Aggregate.
<DupChkOverride>	Boolean	Optional	Duplicate Check Override Flag. When set to TRUE, requests that the server not perform duplicate checking if any is normally performed. The client is affirming that this is a new payment being added.

7.6.1.2 Response <PmtAddRs>

The <PmtAddRs> message is used to provide an acknowledgement to a customer-initiated <PmtAddRq>. It is also used in the Payment Audit Response <PmtAudRs> and Payment Synchronization Response <PmtSyncRs> to communicate to the client that payments have been added by the customer using <PmtAddRq> and by the Pay provider using the customer's Recurring Payment Models.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtInfo>	Aggregate	Required Echoed	Payment Record Aggregate.
<DupChkOverride>	Boolean	Optional Echoed	Duplicate Check Override Flag.
<PmtRec>	Aggregate	Required	Payment Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.6.2 Payment Modify

The Payment Modify message allows a client to modify the current information about a Payment that was set up using the Payment Add message or a Payment that was generated from a Recurring Payment Model. For information on Conventions for Modification of Server-Based Data, see Chapter 2. The Payment Modify message may neither add a new payee nor modify the definition of an existing payee. The client may change the previously defined payee, to whom the payment is intended, by using either the <CustPayeeId>, if the Payment service provider supports its use, or the <StdPayeeId> within <CustPayeeInfo> within <PmtInfo>.

7.6.2.1 Request <PmtModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtId>	Identifier	Required	Payment Identifier. Assigned by the server at the time the Payment is first added. Cannot be modified by the client.

Tag	Type	Usage	Description
<PmtInfo>	Aggregate	Required	Payment Record Aggregate. If the server supports Customer Payee lists, the client must specify the Payee using <CustPayeeId> within <PmtInfo>. Adding new Payees using the <PmtModRq> message is not supported. ***NOTE**: Values of <InvoiceInfo>s included in this aggregate must be interpreted as replacing all existing <InvoiceInfo>s associated with this payment.*
<UpdatePayee>	Boolean	Optional	Update Payee—Update Payee to current level. Used if a previous CustPayeeMod was done without propagating the changes to all pending payments, and the client would like the payee changes propagated to this particular payment. If TRUE, the Payee information for this payment should be updated to match the current level of payee information known to the server. If FALSE, or omitted, do not update the Payee information to the current level. Note that the rest of the <CustPayeeInfo> aggregate may still be used to update the Payee information for this payment, either within a <PmtModRq> or <CustPayeeModRq> as appropriate.
<DupChkOverride>	Boolean	Optional	Duplicate Check Override Flag. When set to TRUE, requests that the server not perform duplicate checking if any is normally performed. The client is affirming that this is a distinct payment, even if the modification creates a duplicate of another payment.

7.6.2.2 Response <PmtModRs>

The <PmtModRs> message is used to provide an acknowledgement to a customer-initiated <PmtModRq>. It is also used in the Payment Audit Response <PmtAudRs> and Payment Synchronization Response <PmtSyncRs> to communicate to the client that the customer has modified payments using <PmtModRq> or that pending payments have undergone status changes.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtId>	Identifier	Required Echoed	Payment Identifier.
<PmtInfo>	Aggregate	Required Echoed	Payment Record Aggregate.
<UpdatePayee>	Boolean	Optional Echoed	Update Payee Flag.
<DupChkOverride>	Boolean	Optional Echoed	Duplicate Check Override Flag.
<PmtRec>	Aggregate	Required	Payment Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.6.3 Payment Status Modify

A user with special authorization, typically a CSR, may send a request to the SP Server to modify the payment status of a specific payment instance. Customers may also send a request to the SP Server to modify the payment status in order to undo a Skip applied to one or more payments. The only fields that may be changed are within the payment status aggregate <PmtStatus>.

7.6.3.1 Request <PmtStatusModRq>

The user must specify the payment identifier along with the payment details to be changed.

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtId>	Identifier	Required	Payment Identifier Aggregate. Assigned by the server at the time the Payment is first added. Cannot be modified by the client. Used to reference a specific payment.
<PmtStatus>	Aggregate	Required	Payment Status Aggregate.

7.6.3.2 Response <PmtStatusModRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtId>	Identifier	Required Echoed	Payment Identifier Aggregate.
<PmtStatus>	Aggregate	Required Echoed	Payment Status Aggregate.
<CSPRefId>	Identifier	Optional	Financial Institution Reference Number.
<SPRefId>	Identifier	Optional	Service Provider Reference Number.

7.6.4 Payment Cancel

The Cancel Payment message allows a client to cancel a Payment that was set up using the Payment Add message or generated from a Recurring Payment Model.

7.6.4.1 Request <PmtCanRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.

Tag	Type	Usage	Description
<PmtId>	Identifier	Required	Payment Identifier. Assigned by the server at the time the Payment is first added. Cannot be modified by the client.

7.6.4.2 Response <PmtCanRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests. Server must return <RqUID> if provided by client in a <PmtCanRq>.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtId>	Identifier	Required Echoed	Payment Identifier.
<PmtRec>	Aggregate	Required	Payment Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.6.5 Payment Inquiry

The Payment Inquiry message allows a client to get a list of the customer's pending, completed, or failed payments on the CPP's server. The client may select on a number of criteria. Clients that keep local copies of a customer's payments may use this message to "refresh" their payment list. This message must not return a customer's recurring payment models but must return any pending, completed, or failed payments that were generated from recurring payment models.

The <FIDebitTrcNum> and <FICreditTrcNum> elements exist to support the case where the CPP submits a payment request to the FI (or CSP) that actually sends the payment messages. For example, a CPP may send a file of ACH-style messages that the FI forwards as actual ACH messages. The FI/CSP may edit the file; e.g., process their in-house messages and forward the rest. As part of this process the FI/CSP may assign reference numbers that correlate the messages to their system of record. There are both credit and debit fields to handle cases like "US on US" where two messages are spawned by the same payment request. Note that the debit and credit trace numbers are received offline but are added to support message histories. Also note that in this scenario all 4 message IDs are used: <SPRefId> is the Pay engine (CPP) reference to the payment and <CSPRefId> is the FI/CSP reference to the payment. One or both were probably returned to the client as confirmation number(s) when the payment was added or modified. <FIDebitTrcNum> and <FICreditTrcNum> are references to the messages that actually move funds to and from accounts.

7.6.5.1 Request <PmtInqRq>

A client initiates the Payment Inquiry message by sending a <PmtInqRq> message.

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.

Tag	*Type*	*Usage*	*Description*
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<PmtType>	Closed Enum	Optional Repeating	Payment Type. Valid values: Pmt, RecPmt This field is used as a selection criterion.
<PmtId>	Identifier	Optional Repeating	Payment Identifier. Assigned by the server at the time the Payment is first added. Cannot be modified by the client. This field is used as a selection criterion.
<RecPmtId>	Identifier	Optional Repeating	Recurring Payment Model Identifier. Assigned by the server at the time the Recurring Payment Model is first added. Cannot be modified by the client. This field is used as a selection criterion.
<DepAcctIdFrom>	Aggregate	Optional XOR	Deposit Account Aggregate. Source account for payment. This field is used as a selection criterion.
<CardAcctIdFrom>	Aggregate	Optional XOR	Credit Account Aggregate. Source account for payment. This field is used as a selection criterion.
<Category>	C-40	Optional Repeating	Payment Category. The customer assigns categories. This field is used as a selection criterion.
<Memo>	C-255	Optional Repeating	Memo for Payment. From Customer to Payee. This field is used as a selection criterion.
<ChkNum>	NC-12	Optional Repeating	Check Number. Assigned by the Pay provider or the CPP if payment is by paper check.
<ChkClrDt>	Date	Optional Repeating	Check Cleared Date. The date when the check cleared the account on which it was drawn.
<FIDebitTrcNum>	NC-7	Optional Repeating	FI Debit Trace Number.
<FICreditTrcNum>	NC-7	Optional Repeating	FI Credit Trace Number.
<PmtMethod>	Closed Enum	Optional Repeating	Payment Method. Intended to provide general method of payment to customer. Valid Values: Check, Electronic
<CustPayeeId>	Identifier	Optional Repeating	Customer's Payee Identifier. This field is used as a selection criterion.
<BillingAcct>	C-32	Optional Repeating	Customer Account Number with Payee. This field is used as a selection criterion.
<Name>	C-40	Optional Repeating	Payee Name.
<StdPayeeId>	Aggregate	Optional Repeating	Customer's Payee Identifier Aggregate. <StdPayeeId> is a synonym for <BillerId>.
<Nickname>	C-40	Optional Repeating	Payee Nickname.
<PmtStatusCode>	Closed Enum	Optional Repeating	Payment Status Code.
<SelRangeDueDt>	Aggregate	Optional	Selection Range Due Date Aggregate.

Tag	Type	Usage	Description
<SelRangePrcDt>	Aggregate	Optional	Selection Range Processing Date Aggregate. Usage is range for actual processing date, possibly adjusted for holidays and non-processing days—as opposed to original customer-entered processing date if using the processing date model.
<SelRangeCurAmt>	Aggregate	Optional	Selection Range Amount Aggregate.
<LastNRec>	Long	Optional	Last N Records.
<DeliveryMethod>	Open Enum	Optional	Delivery Method. Used to request the delivery channel for requested information. Included here primarily to allow customer using an IVR (interactive voice response) unit to request a delivery option for their payment list. See Data Dictionary for details. Default value is Channel
<CSPRefId>	Identifier	Optional Repeating	Customer Service Provider Reference Identifier. <CSPRefId> is used to inquire about a payment corresponding to a confirmation number that was returned to the client when the payment was added or modified. When a payment has been modified, only the <CSPRefId> received in the most recent PMPMODRS is valid. The use of an <CSPRefId> from an earlier response is likely to result in a "payment not found" response.
<SPRefId>	Identifier	Optional Repeating	Service Provider Reference Identifier. Same usage as <CSPRefId>.
<IncToken>	Boolean	Optional	Include Token. If TRUE, a <Token> must be included in the response, if the Service Profile indicates support for payment synchronization, to set a base for future synchronization messages. If FALSE or omitted, the <Token> may be omitted in the response.
<IncHistory>	Boolean	Optional	Include History. If TRUE, the response should include payments that have already occurred, as well as those that are scheduled to occur.

7.6.5.2 Response <PmtInqRs>

The server responds to a <PmtInqRq> message by returning a <PmtInqRs> message to the client.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<PmtType>	Closed Enum	Optional Repeating Echoed	Payment Type. This field is used as a selection criterion.
<PmtId>	Identifier	Optional Repeating Echoed	Payment Identifier. Assigned by the server at the time the Payment is first added. Cannot be modified by the client. This field is used as a selection criterion.

Tag	*Type*	*Usage*	*Description*
<RecPmtId>	Identifier	Optional Repeating Echoed	Recurring Payment Model Identifier. Assigned by the server at the time the Recurring Payment Model is first added. Cannot be modified by the client. This field is used as a selection criterion.
<DepAcctIdFrom>	Aggregate	Optional XOR Echoed	Deposit Account Aggregate. Source account for payment. This field is used as a selection criterion.
<CardAcctIdFrom>	Aggregate	Optional XOR Echoed	Credit Account Aggregate. Source account for payment. This field is used as a selection criterion.
<Category>	C-40	Optional Repeating Echoed	Payment Category. The customer assigns categories. This field is used as a selection criterion.
<Memo>	C-255	Optional Repeating Echoed	Memo for Payment. From Customer to Payee. This field is used as a selection criterion.
<ChkNum>	NC-12	Optional Repeating Echoed	Check Number. Assigned by the Pay provider or the CPP if payment is by paper check.
<ChkClrDt>	Date	Optional Repeating Echoed	Check Cleared Date. The date when the check cleared the account on which it was drawn.
<FIDebitTrcNum>	NC-7	Optional Repeating Echoed	FI Debit Trace Number.
<FICreditTrcNum>	NC-7	Optional Repeating Echoed	FI Credit Trace Number.
<PmtMethod>	Closed Enum	Optional Repeating Echoed	Payment Method. Intended to provide general method of payment to customer.
<CustPayeeId>	Identifier	Optional Repeating Echoed	Customer's Payee Identifier. This field is used as a selection criterion.
<BillingAcct>	C-32	Optional Repeating Echoed	Customer Account Number with Payee. This field is used as a selection criterion.
<Name>	C-40	Optional Repeating Echoed	Payee Name.
<StdPayeeId>	Aggregate	Optional Repeating Echoed	Customer's Payee Identifier Aggregate. <StdPayeeId> is a synonym for <BillerId>.
<Nickname>	C-40	Optional Repeating Echoed	Payee Nickname.
<PmtStatusCode>	Closed Enum	Optional Repeating Echoed	Payment Status Code.
<SelRangeDueDt>	Aggregate	Optional Echoed	Selection Range Due Date Aggregate.
<SelRangePrcDt>	Aggregate	Optional Echoed	Selection Range Processing Date Aggregate.

Tag	Type	Usage	Description
<SelRangeCurAmt>	Aggregate	Optional Echoed	Selection Range Amount Aggregate.
<LastNRec>	Long	Optional Echoed	Last N Records.
<DeliveryMethod>	Open Enum	Optional Echoed	Delivery Method.
<CSPRefId>	Identifier	Optional Repeating Echoed	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional Repeating Echoed	Service Provider Reference Identifier.
<IncToken>	Boolean	Optional Echoed	Include Token.
<IncHistory>	Boolean	Optional Echoed	Include History.
<PmtRec>	Aggregate	Optional Repeating	Payment Record Aggregate. One record is returned for each of the customer's payments that meets the selection criteria specified in the request message. Note that payments may have been generated by a client (using <PmtAddRq> or may have been generated by the Pay provider from one of the customer's Recurring Payment Models.
<Token>	Identifier	Optional *but see Description*	Token. Server Assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. The token is only included if <IncToken>=TRUE in the request. <Token>=0 is returned if no records are returned within the response.

7.6.6 Payment Audit

7.6.6.1 Request <PmtAudRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional	Payment Action. Valid Values: Add, Mod, Can This field is used as a selection criterion.
<PmtId>	Identifier	Optional	Payment Identifier. Assigned by the server at the time the Payment is first added. Cannot be modified by the client. This field is used as a selection criterion.

7.6.6.2 Response <PmtAudRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional Echoed	Payment Action.
<PmtId>	Identifier	Optional Echoed	Payment Identifier. Assigned by the server at the time the Payment is first added. Cannot be modified by the client. This field is used as a selection criterion.
<PmtMsgRec>	Aggregate	Optional Repeating	Payment Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issues the <PmtMsgRec> request.
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<PmtAddRs>	Aggregate	Required XOR	Payment Add Response Message Aggregate.
<PmtModRs>	Aggregate	Required XOR	Payment Modify Response Message Aggregate.
<PmtStatusModRs>	Aggregate	Required XOR	Payment Status Modify Response Message Aggregate.
<PmtCanRs>	Aggregate	Required XOR	Payment Cancel Response Message Aggregate.
</PmtMsgRec>			

7.6.7 Payment Sync

A client uses the Payment Sync message to retrieve a list of changes that have occurred to a customer's payments. This message may be used to enable a client that keeps local copies of a customer's payment messages to synchronize its local database with the Pay provider. The results of this message tell a client what the customer has done using other clients since they last used this one. It may also inform a client about changes the Pay provider made (i.e., generated another payment instance for a recurring payment model, completed a payment or rejected a payment).

Some examples of changes the Pay provider would communicate to a customer include:

- New pending payment generated from a Recurring Payment Model—returned with a <PmtAddRs>;
- Successful payment processing—returned with <PmtModRs> with new <PmtStatus> indicating success;
- Unsuccessful payment processing—returned with <PmtModRs> with new <PmtStatus> indicating status and reason for failure, if available;

- Payment assigned a check number—returned with <PmtModRs> with new <PmtStatus> including <ChkNum>; and
- Payment check cleared—returned with <PmtModRs> with new <PmtStatus> indicating status.

7.6.7.1 Request <PmtSyncRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.

7.6.7.2 Response <PmtSyncRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional	Records Control Output Aggregate.
<Token>	Identifier	Required Echoed	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it.
<PmtMsgRec>	Aggregate	Optional Repeating	Payment Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issues the <PmtMsgRec> request.
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<PmtAddRs>	Aggregate	Required XOR	Payment Add Response Message Aggregate.
<PmtModRs>	Aggregate	Required XOR	Payment Modify Response Message Aggregate.

Tag	Type	Usage	Description
<PmtStatusModRs>	Aggregate	Required XOR	Payment Status Modify Response Message Aggregate.
<PmtCanRs>	Aggregate	Required XOR	Payment Cancel Response Message Aggregate.
</PmtMsgRec>			

7.7 Payment Authorization

7.7.1 Payment Authorization Add

7.7.1.1 Request <PmtAuthAddRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtAuthInfo>	Aggregate	Required	Payment Authorization Information Aggregate.

7.7.1.2 Response <PmtAuthAddRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtAuthInfo>	Aggregate	Required Echoed	Payment Authorization Information Aggregate.
<PmtAuthRec>	Aggregate	Required	Payment Authorization Record Aggregate

7.7.2 Payment Authorization Modify

7.7.2.1 Request <PmtAuthModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.

Tag	Type	Usage	Description
<PmtAuthId>	Identifier	Required	Payment Authorization Identifier.
<PmtAuthInfo>	Aggregate	Required	Payment Authorization Information Aggregate.

7.7.2.2 Response <PmtAuthModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtAuthId>	Identifier	Required Echoed	Payment Authorization Identifier
<PmtAuthInfo>	Aggregate	Required Echoed	Payment Authorization Information Aggregate.
<PmtAuthRec>	Aggregate	Required	Payment Authorization Record Aggregate

7.7.3 Payment Authorization Cancel

7.7.3.1 Request <PmtAuthCanRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtAuthId>	Identifier	Required	Payment Authorization Identifier.

7.7.3.2 Response <PmtAuthCanRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtAuthId>	Identifier	Required Echoed	Payment Authorization Identifier.
<CurAmt>	Aggregate	Optional Repeating Echoed	Currency Amount. Used as a selection criterion.

Tag	Type	Usage	Description
<PmtInst>	Aggregate	Optional Repeating Echoed	Payment Instrument. Used as a selection criterion.

7.7.4 Payment Authorization Inquiry

7.7.4.1 Request <PmtAuthInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtAuthId>	Identifier	Optional Repeating	Payment Authorization Identifier. Used as a selection criterion.
<CurAmt>	Aggregate	Optional Repeating	Currency Amount. Used as a selection criterion.
<PmtInst>	Aggregate	Optional Repeating	Payment Instrument. Used as a selection criterion.
<SelRangeDt>	Aggregate	Optional	Date Selection Range. Used as a selection criterion.

7.7.4.2 Response <PmtAuthInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtAuthId>	Identifier	Optional Repeating Echoed	Payment Authorization Identifier.
<CurAmt>	Aggregate	Optional Repeating Echoed	Currency Amount. Used as a selection criterion.
<PmtInst>	Aggregate	Optional Repeating Echoed	Payment Instrument. Used as a selection criterion.
<SelRangeDt>	Aggregate	Optional Echoed	Date Selection Range. Used as a selection criterion.
<PmtAuthRec>	Aggregate	Optional Repeating	Payment Authorization Record. One aggregate for each record matching the selection criteria in the request.

7.7.5 Payment Authorization Audit

7.7.5.1 Request <PmtAuthAudRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional	Payment Action. This field is used as a selection criterion.
<PmtAuthId>	Identifier	Optional	Payment Authorization Identifier. Assigned by the server at the time the Payment Authorization is first added. Cannot be modified by the client. This field is used as a selection criterion.

7.7.5.2 Response <PmtAuthAudRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional Echoed	Payment Action. This field is used as a selection criterion. If not present, defaults to ALL.
<PmtAuthId>	Identifier	Optional Echoed	Payment Authorization Identifier. Assigned by the server at the time the Payment Authorization is first added. Cannot be modified by the client. This field is used as a selection criterion.
<PmtAuthMsgRec>	Aggregate	Optional Repeating	Payment Authorization Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issues the <PmtAuthMsgRec> request.
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<PmtAuthAddRs>	Aggregate	Required XOR	Add Payment Authorization Response Message Aggregate.

Tag	Type	Usage	Description
<PmtAuthModRs>	Aggregate	Required XOR	Modify Payment Authorization Response Message Aggregate.
<PmtAuthCanRs>	Aggregate	Required XOR	Cancel Payment Authorization Response Message Aggregate.
</PmtAuthMsgRec>			

7.7.6 Payment Authorization Synchronization

7.7.6.1 Request <PmtAuthSyncRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.

7.7.6.2 Response <PmtAuthSyncRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional	Records Control Output Aggregate.
<Token>	Identifier	Required Echoed	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first time requests.
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it.
<PmtAuthMsgRec>	Aggregate	Optional Repeating	Payment Authorization Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issues the <PmtAuthMsgRec> request.

Tag	Type	Usage	Description
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<PmtAuthAddRs>	Aggregate	Required XOR	Add Payment Authorization Response Message Aggregate.
<PmtAuthModRs>	Aggregate	Required XOR	Modify Payment Authorization Response Message Aggregate.
<PmtAuthCanRs>	Aggregate	Required XOR	Cancel Payment Authorization Response Message Aggregate.
</PmtAuthMsgRec>			

7.8 Remittance

7.8.1 Remittance Add

The Remittance Add message allows a client to transmit a remittance advice to a BPP or BSP for posting.

7.8.1.1 Request <RemitAddRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<EffDt>	Date	Required	Effective Date. The date the payment was processed by the CPP. It is used for remittance advice from CPP to BPP. The date can be used by the BPP to associate the date the payment was processed with the posting date. This date should match the date settlement to the BPP/Biller was initiated.
<PmtId>	Identifier	Optional	Payment Identifier. The identifier of the payment associated with this remittance. This is included to enable the BPP to inform the CPP of payment posting against this remittance.
<RemitInfo>	Aggregate	Required	Remittance Record Aggregate.
<DupChkOverride>	Boolean	Optional	Duplicate Check Override Flag. When set to TRUE, requests that the server not perform duplicate checking if any is normally performed. The client is affirming that this is a new remittance being added.

7.8.1.2 Response <RemitAddRs>

The <RemitAddRs> message is used to provide an acknowledgement to a CPP-initiated <RemitAddRq>. It is also used in the Remittance Audit Response <RemitAudRs> to communicate to the client that remittances have been added by the CPP using <RemitAddRq>.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.

Tag	Type	Usage	Description
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<EffDt>	Date	Required Echoed	Effective Date. The date the payment was processed by the CPP. It is used for remittance advice from CPP to BPP. The date can be used by the BPP to associate the date the payment was processed with the posting date. This date should match the date settlement to the BPP/Biller was initiated.
<PmtId>	Identifier	Optional Echoed	Payment Identifier. The identifier of the payment associated with this remittance. This is included to enable the BPP to inform the CPP of payment posting against this remittance.
<RemitInfo>	Aggregate	Required Echoed	Remittance Record Aggregate.
<DupChkOverride>	Boolean	Optional Echoed	Duplicate Check Override Flag.
<RemitRec>	Aggregate	Required	Remittance Record Aggregate.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.8.2 Remittance Modify

The Remittance Modify message allows a client to modify the current information about a Remittance that was set up using the Remittance Add message.

7.8.2.1 Request <RemitModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RemitId>	Identifier	Required	Remittance Identifier. Assigned by the server at the time the Remittance is first added. Cannot be modified by the client.
<EffDt>	Date	Required	Effective Date. The date the payment was processed by the CPP. It is used for remittance advice from CPP to BPP. The date can be used by the BPP to associate the date the payment was processed with the posting date. This date should match the date settlement to the BPP/Biller was initiated.
<PmtId>	Identifier	Optional	Payment Identifier. The identifier of the payment associated with this remittance. This is included to enable the BPP to inform the CPP of payment posting against this remittance.
<RemitInfo>	Aggregate	Required	Remittance Information Aggregate.
<DupChkOverride>	Boolean	Optional	Duplicate Check Override Flag. When set to TRUE, requests that the server not perform duplicate checking if any is normally performed. The client is affirming that this is a new remittance being added.

7.8.2.2 Response <RemitModRs>

The <RemitModRs> message is used to provide an acknowledgement to a CSP-initiated <RemitModRq>. It is also used in the Remittance Audit Response <RemitAudRs> to communicate to the client that remittances have been modified by the customer using <RemitModRq> or that pending payments have undergone status changes.

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<EffDt>	Date	Required Echoed	Effective Date. The date the payment was processed by the CPP. It is used for remittance advice from CPP to BPP. The date can be used by the BPP to associate the date the payment was processed with the posting date. This date should match the date settlement to the BPP/Biller was initiated.
<PmtId>	Identifier	Optional Echoed	Payment Identifier. The identifier of the payment associated with this remittance. This is included to enable the BPP to inform the CPP of payment posting against this remittance.
<RemitId>	Identifier	Required Echoed	Remittance Identifier.
<RemitInfo>	Aggregate	Required Echoed	Remittance Record Aggregate.
<DupChkOverride>	Boolean	Optional Echoed	Duplicate Check Override Flag.
<RemitRec>	Aggregate	Required	Remittance Record Aggregate.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.8.3 Remittance Status Modify

7.8.3.1 Request <RemitStatusModRq>

The client must specify the remittance identifier along with the status.

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RemitId>	Identifier	Required	Remittance Identifier Aggregate. Assigned by the server at the time the Remittance is first added. Cannot be modified by the client. Used to reference a specific remittance.
<RemitStatus>	Aggregate	Required	Remittance Status Aggregate.

7.8.3.2 Response <RemitStatusModRs>

Tag	*Type*	*Usage*	*Description*

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtId>	Identifier	Required Echoed	Remittance Identifier Aggregate.
<RemitStatus>	Aggregate	Required Echoed	Remittance Status Aggregate.
<CSPRefId>	Identifier	Optional	Financial Institution Reference Number.
<SPRefId>	Identifier	Optional	Service Provider Reference Number.

7.8.4 Remittance Delete

The Delete Remittance message allows a client to delete a Remittance that was set up using the Remittance Add message.

7.8.4.1 Request <RemitDelRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RemitId>	Identifier	Required	Remittance Identifier. Assigned by the server at the time the Remittance is first added. Cannot be modified by the client.

7.8.4.2 Response <RemitDelRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests. Server must return <RqUID> if provided by client in a <PmtCanRq>.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RemitId>	Identifier	Required Echoed	Remittance Identifier
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.8.5 Remittance Inquiry

7.8.5.1 Request <RemitInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RemitId>	Identifier	Optional Repeating	Remittance Identifier. Used as a selection criterion.
<CurAmt>	Aggregate	Optional Repeating	Currency Amount. Used as a selection criterion.
<SelRangeDt>	Aggregate	Optional	Date Selection Range. Used as a selection criterion.

7.8.5.2 Response <RemitInqRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RemitId>	Identifier	Optional Repeating Echoed	Remittance Identifier.
<CurAmt>	Aggregate	Optional Repeating Echoed	Currency Amount. Used as a selection criterion.
<SelRangeDt>	Aggregate	Optional Echoed	Date Selection Range. Used as a selection criterion.
<RemitRec>	Aggregate	Required	Remittance Record Aggregate.

7.8.6 Remittance Audit

Remittance Audit allows a client to audit Remittance Add/Modify/Delete messages associated with the current customer. When the <RemitStatus> changes, the server must generate an Rs message to the Rq that created the pending state. The <Status> <Severity> must always be Info.

7.8.6.1 Request <RemitAudRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.

Tag	Type	Usage	Description
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional	Remittance Method. Valid Values: Add, Mod, Del This field is used as a selection criterion.
<RemitId>	Identifier	Optional	Remittance Identifier. Assigned by the server at the time the remittance is first added. Cannot be modified by the client. This field is used as a selection criterion.

7.8.6.2 Response <RemitAudRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional	Records Control Output Aggregate.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional Echoed	Remittance Method.
<RemitId>	Identifier	Optional Echoed	Remittance Identifier. Assigned by the server at the time the remittance is first added. Cannot be modified by the client. This field is used as a selection criterion.
<RemitMsgRec>	Aggregate	Optional Repeating	Remittance Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issues the <PmtMsgRec> request.
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<RemitAddRs>	Aggregate	Required XOR	Remittance Add Response Message Aggregate.
<RemitModRs>	Aggregate	Required XOR	Remittance Modify Response Message Aggregate.
<RemitStatusModRs>	Aggregate	Required XOR	Remittance Status Modify Response Message Aggregate.
<RemitDelRs>	Aggregate	Required XOR	Remittance Delete Response Message Aggregate.
</RemitMsgRec>			

7.8.7 Remittance Synchronization

A client uses the Remittance Synchronization message to retrieve a list of changes that have occurred to a client's remittance. This message will commonly be used to enable a client to synchronize with the server to get up-to-date status information about a remittance. The results of this message typically will status change of a remittance advice.

7.8.7.1 Request <RemitSyncRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or "0" for first time requests.

7.8.7.2 Response <RemitSyncRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional	Records Control Output Aggregate.
<Token>	Identifier	Required Echoed	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or "0" for first time requests.
<NewToken>	Identifier	Required	New Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it.
<RemitMsgRec>	Aggregate	Optional Repeating	Remittance Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issues the <PmtMsgRec> request.
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<RemitAddRs>	Aggregate	Required XOR	Remittance Add Response Message Aggregate.

Tag	Type	Usage	Description
<RemitModRs>	Aggregate	Required XOR	Remittance Modify Response Message Aggregate.
<RemitStatusModRs>	Aggregate	Required XOR	Remittance Status Modify Response Message Aggregate.
<RemitDelRs>	Aggregate	Required XOR	Remittance Delete Response Message Aggregate.
</RemitMsgRec>			

7.9 Recurring Payment Model

7.9.1 Recurring Payment Model Add

The Recurring Payment Model Add message allows a client to set up a recurring or repeating payment where the payment amount is the same. Examples of these types of payments are mortgages, car loans, equity loans, etc. The initial and/or final payment amount may be different from the normal recurring payment amount if supported by the Pay provider.

7.9.1.1 Request <RecPmtAddRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtInfo>	Aggregate	Required	Payment Information Aggregate.
<RecModelInfo>	Aggregate	Required	Recurring Model Information Aggregate.
<DupChkOverride>	Boolean	Optional Profiled support	Duplicate Check Override Flag. When set to TRUE, requests that the server not perform duplicate checking if any is normally performed. The client is affirming that this is a new recurring payment model being added.

7.9.1.2 Response <RecPmtAddRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtInfo>	Aggregate	Required Echoed	Payment Information Aggregate.
<RecModelInfo>	Aggregate	Required Echoed	Recurring Model Information Aggregate.

Tag	Type	Usage	Description
<DupChkOverride>	Boolean	Optional Profiled support Echoed	Duplicate Check Override Flag.
<RecPmtRec>	Aggregate	Required	Recurring Payment Model Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.9.2 Recurring Payment Model Modify

The Modify Recurring Payment Model message allows a client to modify the current definition of a recurring or repeating payment. Whether a modification to the model affects any current pending payment instance(s) from the model depends on profile settings and the message contents.

7.9.2.1 Request <RecPmtModRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecPmtId>	Identifier	Required	Recurring Payment Model Identifier. Assigned by the server at the time the Recurring Payment Model is first added. Cannot be modified by the client.
<PmtInfo>	Aggregate	Required	Payment Record Aggregate. If the server supports Customer Payee lists, the Payee must be specified using <PmtInfo> <CustPayeeId>.
<RecModelInfo>	Aggregate	Required	Recurring Model Information Aggregate.
<ModPending>	Boolean	Optional Profiled support	Modify Pending Indicator. If set to TRUE, all pending payment instances that were automatically generated from the recurring model that is being modified must also be modified. If absent or set to FALSE, only the model is modified; any pending payments must remain unaffected by the modification of the model. This element must be ignored unless <ModPendingType> in the Pay Service Profile = IfRequested.

7.9.2.2 Response <RecPmtModRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<PmtInfo>	Aggregate	Required Echoed	Payment Record Aggregate.

Tag	Type	Usage	Description
<RecModelInfo>	Aggregate	Required Echoed	Recurring Model Information Aggregate.
<DupChkOverride>	Boolean	Optional Profiled support Echoed	Duplicate Check Override Flag.
<RecPmtRec>	Aggregate	Required	Recurring Payment Model Record Aggregate.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.9.3 Recurring Payment Model Cancel

Allows a client to cancel a customer's Recurring Payment Model. Cancellation of a Recurring Payment Model always also cancels any pending Payments that were generated from that model.

7.9.3.1 Request <RecPmtCanRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecPmtId>	Identifier	Required	Recurring Payment Model Identifier. Assigned by the server at the time the Recurring Payment Model is first added. Cannot be modified by the client.
<CascadeDel>	Boolean	Optional	Cascade Delete. If TRUE, server must delete all dependent payments when this model is deleted. If FALSE or omitted, the model must not be deleted if dependent payments exist.

7.9.3.2 Response <RecPmtCanRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecPmtId>	Identifier	Required Echoed	Recurring Payment Model Identifier.
<CascadeDel>	Boolean	Optional Echoed	Cascade Delete.
<RecPmtRec>	Aggregate	Required XOR	Recurring Payment Record
<DependentType>	Open Enum	Required XOR Repeating	An aggregate that would contain a list of depending object types that exist for the model.
<CSPRefId>	Identifier	Optional	Customer Service Provider Reference Identifier.

Tag	Type	Usage	Description
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

7.9.4 Recurring Payment Model Inquiry

The Recurring Payment Inquiry message allows a client to get a list of the current Recurring Payment Models. Clients that keep local copies of a customer's Recurring Payment Models may use this message to "refresh" their copies of the customer's Recurring Payment Models.

7.9.4.1 Request <RecPmtInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<RecPmtId>	Identifier	Optional Repeating	Recurring Payment Model Identifier. Assigned by the server at the time the Recurring Payment Model is first added. Cannot be modified by the client. This field is used as a selection criterion.
<CustPayeeId>	Identifier	Optional Repeating	Customer's Payee Identifier. This field is used as a selection criterion. If the server supports Payee Lists, the server assigns <CustPayeeId> at the time the Payee is first added.
<DepAcctIdFrom>	Aggregate	Optional XOR	Deposit Account Aggregate. Source account for payment. This field is used as a selection criterion.
<CardAcctIdFrom>	Aggregate	Optional XOR	Credit Account Aggregate. Source account for payment. This field is used as a selection criterion.
<Category>	C-40	Optional Repeating	Payment Category. The customer assigns categories. This field is used as a selection criterion.
<Memo>	C-255	Optional Repeating	Memo for Payment. From Customer to Payee. This field is used as a selection criterion.
<BillingAcct>	C-32	Optional Repeating	Customer Account Number with Payee. This field is used as a selection criterion.
<SelRangeCurAmt>	Aggregate	Optional	Selection Range Amount Aggregate.
<CSPRefId>	Identifier	Optional Repeating	Customer Service Provider Reference Identifier. This field is used as a selection criterion.
<SPRefId>	Identifier	Optional Repeating	Service Provider Reference Identifier. This field is used as a selection criterion.
<IncToken>	Boolean	Optional	Include Token. If TRUE, a <Token> should be included in the response to set a base for future Sync messages. If FALSE or omitted, no <Token> is returned.

7.9.4.2 Response <RecPmtInqRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<RecPmtId>	Identifier	Optional Repeating Echoed	Recurring Payment Model Identifier.
<CustPayeeId>	Identifier	Optional Repeating Echoed	Customer's Payee Identifier.
<DepAcctIdFrom>	Aggregate	Optional XOR Echoed	Deposit Account Aggregate.
<CardAcctIdFrom>	Aggregate	Optional XOR Echoed	Credit Account Aggregate.
<Category>	C-40	Optional Repeating Echoed	Payment Category.
<Memo>	C-255	Optional Repeating Echoed	Memo for Payment.
<BillingAcct>	C-32	Optional Repeating Echoed	Customer Account Number with Payee.
<SelRangeCurAmt>	Aggregate	Optional Echoed	Selection Range Amount Aggregate.
<CSPRefId>	Identifier	Optional Repeating Echoed	Customer Service Provider Reference Identifier.
<SPRefId>	Identifier	Optional Repeating Echoed	Service Provider Reference Identifier.
<RecPmtRec>	Aggregate	Optional Repeating	Recurring Payment Model Record Aggregate. One record is returned for each of the customer's Recurring Payment Models that meets the selection criteria specified in the request message
<Token>	NC-32	Optional *but see Description*	Token. Server Assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is significant only to the server that originally assigned it. The token is only included if <IncToken>=TRUE in the request. <Token>=0 is returned if no records are returned within the response.

7.9.5 Recurring Payment Model Audit

7.9.5.1 Request <RecPmtAudRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Message Records Control Input Aggregate.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional	Recurring Payment Model Method. Valid Values: Add, Mod, Can This field is used as a selection criterion.
<RecPmtId>	Identifier	Optional	Recurring Payment Model Identifier. Assigned by the server at the time the Recurring Payment Model is first added. This field is used as a selection criterion.

7.9.5.2 Response <RecPmtAudRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<Method>	Closed Enum	Optional Echoed	Recurring Payment Model Method.
<RecPmtId>	Identifier	Optional Echoed	Recurring Payment Model Identifier.
<RecPmtMsgRec>	Aggregate	Optional Repeating	Recurring Payment Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issues the <RecPmtMsgRec> request.
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<RecPmtAddRs>	Aggregate	Required XOR	Add Recurring Payment Model Response Message Aggregate.
<RecPmtModRs>	Aggregate	Required XOR	Modify Recurring Payment Model Response Message Aggregate.

Tag	Type	Usage	Description
<RecPmtCanRs>	Aggregate	Required XOR	Cancel Recurring Payment Model Response Message Aggregate.
</RecPmtMsgRec>			

7.9.6 Recurring Payment Model Sync

The Recurring Payment Model Sync message allows clients to retrieve a list of changes that have occurred to a customer's Recurring Payment Models. This message may be used to enable a client that keeps local copies of a customer's Recurring Payment messages to synchronize its database against the one kept by the Pay provider. The results of this message tell a client what the customer has done using other clients since they last used this one.

7.9.6.1 Request <RecPmtSyncRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Message Records Control Input Aggregate.
<Token>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first-time requests.

7.9.6.2 Response <RecPmtSyncRs>

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<Token>	Identifier	Required Echoed	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. This is a token that has been previously sent by the server, or zero for first-time requests.

Tag	Type	Usage	Description
<NewToken>	Identifier	Required	Token. Server assigned. Client should make no assumptions about the value of this token relative to others that it may have received. The token is only significant to the server that originally assigned it. If the client sent a token in the request, the server returns a new token based on this audit message.
<RecPmtMsgRec>	Aggregate	Optional Repeating	Recurring Payment Message Record Aggregate.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. Identifies the party that actually issues the <RecPmtMsgRec> request.
<MsgRecDt>	Timestamp	Optional	Audit Record Creation Date. Date when this audit record is created.
<RecPmtAddRs>	Aggregate	Required XOR	Add Recurring Payment Model Response Message Aggregate.
<RecPmtModRs>	Aggregate	Required XOR	Modify Recurring Payment Model Response Message Aggregate.
<RecPmtCanRs>	Aggregate	Required XOR	Cancel Recurring Payment Model Response Message Aggregate.
</RecPmtMsgRec>			

7.10 Pay Service Profile <PaySvcProfInfo>

The profile for the Pay Service is defined below. This profile is returned to the client in the Service Profile Inquiry <SvcProfInqRs> response and provides information on how the client should use the Pay Service.

Tag	Type	Usage	Description
<SvcCore>	Aggregate	Required	Service Core Aggregate. Information specified for every service and version.
<MsgSupt>	Open Enum	Required Repeating	Supported Messages. This is a list of messages that are supported for Pay. The convention is to use the name of the message without the Rq or Rs so that each message is only listed once. Defined values: StdPayeeInq, CustPayeeAdd, CustPayeeMod, CustPayeeTypeMod, CustPayeeDel, CustPayeeInq, CustPayeeAud, CustPayeeSync, PmtAdd, PmtMod, PmtStatusMod, PmtCan, PmtInq, PmtAud, PmtSync, PmtAuthAdd, PmtAuthMod, PmtAuthCan, PmtAuthInq, PmtAuthAud, PmtAuthSync, RemitAdd, RemitMod, RemitStatusMod, RemitDel, RemitInq, RemitAud, RemitSync, RecPmtAdd, RecPmtMod, RecPmtCan, RecPmtInq, RecPmtAud, RecPmtSync
<OptSupt>	Open Enum	Optional Repeating	Options Supported. Defined values: AcctNickname, BillerPayee, CustPayeeNickName, ForExCommit, FSPayee, ImmediatePmt, LineItem, RecCtrl, RecFinalCurAmt, RecInitialCurAmt, Skip, XferPayee
<PrcSched>	Aggregate	Optional	Processing Schedule Aggregate. If omitted, the default processing schedule is assumed.
<BankAcctFromProf>	Aggregate	Required	Bank Account From Profile Aggregate. Usage is payment source account profile.

Tag	*Type*	*Usage*	*Description*
<BankAcctToProf>	Aggregate	Optional	Bank Account To Profile Aggregate. Usage is payment destination account profile. Only included if Interbank Transfer Payments are supported.
<PmtModel>	Closed Enum	Required	Payment Model. Indicates whether the client must enter the date that the server should initiate processing of the payment <PrcDt> or the date that the payment is due <PmtDueDt> within the Payment Record Aggregate <PmtInfo>. Valid values: DueDt, PrcDt.
<DfltDaysToPay>	Long	Required	Default Days to Pay. The default number of days required to complete the payment by check. Does not include transfers.
<DfltXferDaysToPay>	Long	Required	Default Transfer Days to Pay. The default number of days required to complete the payment by transfer.
<DaysWith>	Long	Required	Withdrawal Date Offset. Used in determination of date to withdraw funds from customer account. Usage is <DueDt>—<DaysToPay> + <DaysWith> provides withdrawal date. NOTE—If the value of <DaysWith> is –1 then the withdrawal date is the same as <DueDt>.
<HistRetentionDays>	Long	Required	History Retention Days. Number of days after a message is processed that it is available for inquiries.
<DeliveryMethod>	Open Enum	Optional Repeating	Delivery Method. Defined values: Channel, HomeBank, Post, UPS, Courier. Note that although these are valid values for this element, they may not be appropriate for a particular message and may result in rejection. In that case, a response must be sent to the customer with an appropriate Status Code.
<ModPendingType>	Closed Enum	Required	Payee Modify Pending specifies the rules used by the server to propagate changes to payees to pending single payments. Changes to payees are always propagated to payment models. Valid values: Always, Never or IfRequested. If set to Always, changes to payees are always propagated to pending payments. If set to Never, changes to payees are never propagated to pending payments. If set to IfRequested, changes to payees are propagated to pending payments when <ModPending> is set to TRUE in the Customer Payee Modify request <CustPayeeModRq> and not propagated when <ModPending> is set to FALSE or not specified.
<RecPmtProf>	Aggregate	Optional	Recurring Payment Profile Aggregate.
<Freq>	Open Enum	Required Repeating	Recurring Model Frequency. Usage is a list of supported frequencies. See Data Dictionary for details.
<ModPendingType>	Closed Enum	Required	Client Modify Pending Type. Valid values: Always, Never, IfRequested. If set to Always, changes to recurring payment models are always propagated to pending payments based on that model. If set to Never, changes to recurring payment models are never propagated to pending payments based on that model. If set to IfRequested, changes to recurring payment models are propagated to pending payments based on that model when <ModPending> is set to TRUE in the Recurring Payment Model Modify request <RecPmtModRq> and not propagated when <ModPending> is set to FALSE or not specified.

Tag	*Type*	*Usage*	*Description*
</RecPmtProf>			

8 The Bill Presentment Service <PresSvc>

Bill Presentment is the electronic delivery of a bill from a biller to a customer. While this chapter focuses on the presentation of bills, this service may be used for the presentation of other documents such as notices, statements, and invoices. A company that distributes bills often has a variety of information that is to be presented to a customer, including payment information, announcements of new services, and changes in the terms and conditions of service. A biller also needs to know that they and their service providers have been able to get the bills to the customers. They often need to know that the customer has seen the bill, statement, or notice, and having information about the customer's actions returned to the biller and their service provider should facilitate business.

To support the widest intersection of customers who want bills, and billers that want bill delivery, the concept of the biller directory has been developed. Basically, the flow is that a customer's agent (the CSP) locates billers through the Biller Directory, uses the customer, customer/service link, and service/account link messages to enable presentment and payment; obtains the current list of bills available for the consumer; and allows the consumer to initiate payment. This chapter covers the methods for finding billers, obtaining bills, and updating bill statuses.

Although some billers may provide Bill Presentment service themselves, many may choose to work with a BSP that provides Bill Presentment service on behalf of many billers. For this reason, Bill Presentment focuses on connecting customers to BSPs.

8.1 Description

This section summarizes the process of receiving bills electronically, starting with the steps required to find a BSP and set up Bill Presentment service.

To receive bills electronically, a CSP determines which BSPs provide Bill Presentment service for the billers. Subsequently, the customer:

- Finds one or more billers by searching a biller directory server. The location and access to such servers is not currently defined within IFX.
- Determines which BSPs provide Bill Presentment service for the billers.
- Enables the bill presentment service with a CSP, who forwards the message on to the BSP or Biller.
- Enrolls with a CPP for a Payment service (if not already enrolled).
- Activates bill presentment with the BSP or Biller for one or more accounts with one or more billers.
- Requests electronic bills from the BSP via the CSP.
- Optionally pays the bills.

8.1.1 Biller Directory

To find billers, the client sends a <BillerInqRq> request. A <BillerInqRs> response is returned.

8.1.1.1 Search Arguments

If the client omits all elements in the <BillerInqRq>, the client is requesting a complete directory of billers. Otherwise, the client wants to filter results based on the included elements.

8.1.2 Activate Bill Delivery

Once the customer has located a biller, the customer enables the bill presentment service using the <CustSvcAddRq>. After enabling the service, the customer may activate presentment accounts for one or more

billers at that BSP using the <SvcAcctAddRq> in order to receive bills. Account Activation may be an off-line process for many billers, and in such situations the <SvcAcctStatusCode> must return *xxx*Pend with an optional estimated decision date in the message. <SvcAcctInqRq> is used to request the current account status.

8.1.3 Account Inquiry

The <SvcAcctInqRs> response must return a <PresAcctId> aggregate and <SvcAcctStatus> <StatusDesc> for each of the customer's accounts with the billers at that BSP. The response may list only those accounts that have been activated for Bill Presentment service; it is not required to list all available accounts.

Unlike a financial institution, BSPs generally will not have information about all the accounts of its supported billers. Billers that also serve as their own BSPs may be able return available accounts as well as activated accounts.

8.1.4 Bill Presentment

The customer requests bills that are ready for presentment using the <BillInqRq>. The responding <BillInqRs> includes bill summaries. Bill Summary is information from a biller that is essential for a customer to understand what is owed, which may include, but is not limited to, Amount Owed, Date Due, Biller, and Customer's Account Number with the Biller.

A customer may also elect to view an electronic version of the detail of the bill. Bill Detail is information from a biller that provides invoice line level information to a customer, such as credit card charges, telephone calls, or kilowatts used.

IFX provides an <ImageURL>, included in <StmtImage>, to indicate the location where bill detail information may be found and retrieved. The definition of structured data for communicating Bill Detail in email or other message formats is planned for future releases of the specification.

The bill record aggregate <BillRec> provides the ability to deliver bill-related information, including bill summary information about a single bill, a billing statement, a notification (textual information sent from the biller to communicate information about the bill presentment service), or an invoice, depending on the value in the <BillType> element. The Bill Summary may include such information as amount due, date due, and pointers to more information; e.g., a set of URLs that may used to access bill detail and other information. The actual elements used are likely to vary depending on the Bill Type. For example, with <BillType> Notice, the biller may send <Memo> data but none of the other optional elements.

A Biller may use the <Memo> element within the Bill Summary to send human-readable text to the customer, regardless of bill type. This may include, for example, special instructions for accessing information within the bill detail when sent with a bill, or regulatory or other announcements when sent as part of a bill or statement or alone as part of a notice.

The bill date, <BillDt>, is the bill's "as of" date. It is sometimes referred to as "invoice date" or "statement date" on a paper bill. It is often the date that the biller generated the bill. It is not the date on which the BSP received the bill for publication or the date that the payment is due. <PmtDueDt> is the date used by the biller to indicate when payment is to be received according to the terms and conditions of the account.

The Bill Summary Amount <BillSummAmt> aggregate is used to specify any currency values for the bill, including "amount due" when applicable. If <BillType> = Bill, at least one occurrence of this aggregate must be present where the amount is a payable amount (i.e., <BillAmtType> is <Payable>).

Amount Type: The <BillSummAmt> aggregate is designed to allow maximum flexibility in specifying amounts on a bill summary. The requirements vary greatly by industry, to the extent that a single "amount due" is not sufficient. Some currency values on a bill represent options on an amount to pay (specified by the <BillSummAmtType> = Payable). Some values on the bill summary may be for information only and are not an amount to be paid (specified by the <BillSummAmtType> = InfoOnly). Other values may be presented to denote a category for a supplemental or overpayment by the consumer (specified by the <BillSummAmtType> = Supplemental). Examples for each type are:

- **Payable**

a) Amount Due and Minimum Amount Due on a credit card

b) One-month, Three-month, and Six-month premium on an insurance policy

c) Amount Due and Late Amount Due on a utility bill.

- **InfoOnly**
  a) Discount amount for an early payment of a utility bill

  b) Current Charges, Current Credits, Statement Balance, and Finance Charges on a credit card.

- **Supplemental**
  a) Extra Principal and Extra Escrow on a mortgage

  b) Tip and Charitable Contribution on a newspaper bill.

Note: *When multiple Payable amounts are specified within a bill summary, they must be treated as mutually exclusive; i.e., only one Payable amount may be selected for payment. Supplemental amounts are not mutually exclusive; i.e., the consumer may select more than one Supplemental amount for allocating an overpayment.*

Sub-Amounts: Another type of amount that may be presented is a breakdown of a total amount, typically a payable amount. The <BillSummSubAmt> aggregate allows for this breakdown, and associates the breakdown with a specific amount within a <BillSummAmt> aggregate. The amounts within the sub-amount aggregate may be designated as a category that may be selected by the consumer for allocation of their total payment (<AllocateAllowed> = TRUE), or the amounts within this breakdown may be information only (<AllocateAllowed> = FALSE).

Sub-Amount Example: A consolidated insurance bill may include Payable amounts of 3-month premium and 6-month premium, at the <BillSummAmt> level. Within the <BillSummAmt> aggregate for the 3-month premium, the biller may include two sub-amount aggregates, one for the life insurance 3-month premium and one for the health insurance 3-month premium. A similar breakdown may be included for the 6-month premium. If the biller wants the consumer to be able to specify what portion of their total payment (perhaps if they were paying something other than the amount billed, for example) was for life and what portion was for health, he would designate these sub-amounts as <AllocateAllowed> = TRUE. If the sub-amounts were included for information only, the biller would designate these amounts as <AllocateAllowed> = FALSE.

The table below demonstrates some possible industry examples for using the amount types and sub-amount aggregate.

Amount ID: The <BillSummAmt> aggregate also allows for the BSP or biller to specify an identifier, or tag, for each amount specified. This identifier may be included in a payment message to return not only the amount paid, but also the specific amount category being paid. Using this identifier allows the consumer to specify the total amount paid, as well as how the allocation of the payment is intended.

Description Data: Each amount within the <BillSummAmt> aggregate must carry with it a short description <ShortDesc>. This is the description that may be displayed to the consumer to correspond with the currency amount specified, where a shortened description is needed, such as in a grid box. A more detailed description <Desc> may be specified for clarity, when the <ShortDesc> is insufficient to properly define the amount. The <Memo> tag may be used to provide Help text or terms that pertain to an amount. In addition, the aggregate allows the Biller to categorize the amount into a type, specified by an open enum list, when possible. This would allow the CSP to have a machine-readable understanding of the amount being specified.

IMPORTANT NOTES ABOUT BILL SUMMARY AMOUNTS:

- All values for <CurAmt> are valid, including 0 and negative amounts.
- A CSP is not obligated to display all amounts specified within the bill summary. The CSP should display the bill summary amounts, beginning with the first occurrence in the message, proceeding in order and giving priority to Payable amounts, versus InfoOnly or Supplemental amounts or amounts specified within

the <SubAmt> aggregate. Therefore, each occurrence of the <BillSummAmt> aggregate should be included in the message in priority order, with the first being the most important amount to display to the consumer, the second being the next highest amount in importance, and so forth, with the last occurrence being the least important. Although it may vary by client, billers should consider that amounts specified as <BillSummAmtType> = Supplemental or specified within <BillSummSubAmt> might only be displayed when the bill is selected for payment.

The CSP or CPP may choose not to perform any edit checks on the amount actually paid by the consumer.

EXAMPLES - Using <BillSummAmt> Aggregate:

Industry Example	*Short Desc*	*Payable*	*InfoOnly*	*Supplemental*	*SubAmt*	*AllocateAllowed*
Utilities	Amount Due	×				
	Gas				×	False
	Electric				×	False
	Charity Donation			×		
	Late Charge		×			
	Late Amount Due	×				
	Early Discount		×			
	Discnt Amt Due	×				
	Early Gas				×	False
	Early Electric				×	False
Newspaper	Monthly Amount	×				
	3 Month Saver	×				
	6 Month Super-Saver	×				
	Tip			×		
	Charity Donation			×		
Mortgage	Principal		×			
	Interest		×			
	RE Taxes		×			
	Insurance		×			
	Escrow Total		×			
	PMI		×			
	Total Amnt Due	×				
	Extra Principal			×		
	Extra Escrow			×		
Credit Card	Current Balance	×				
	Minimum Amt Due	×				
	Revolving Minimum				×	True
	Long Term Minimum				×	True
	New Charges		×			
	New Credits		×			
	Finance Charges		×			
	Late Charges		×			

Insurance	3-month Premium	×				
	3-month Life				×	True
	3-month Health				×	True
	6-month Premium	×				
	6-month Life				×	True
	6-month Health				×	True
	Non-Smoker Disc%		×			

8.1.5 Bill Payment

The customer may pay the bill using the payment messages specified in Chapter ***. Note that the <BillerId> used to identify a biller may be different for presentment and payment, even if the CPP and BSP are the same organization. Note also that the <BillRec> aggregate contains both the biller presentment account number and a <BillRefInfo> data element, which are returned with the payment to facilitate correct posting of the payment by the biller's accounts receivable.

8.2 Bill Presentment Service Message Summary

Function / Message Name	*Req.*	*Comments*
Biller Inquiry **<BillerInqRq>** **<BillerInqRs>**	Yes	Allows client to view a summary of current Biller records maintained by the Pay or Presentment service provider. Payment billers may or may not have been previously added to the customer's Payee List.
Bill Inquiry **<BillInqRq>** **<BillInqRs>**	Yes	Allows client to retrieve bills from the biller.
Bill Status Modification **<BillStatusModRq>** **<BillStatusModRs>**		Allows a client (Customer/CSP/CPP) to notify the BSP/Biller that the status of a bill or payment for the bill has changed.

8.3 Presentment Service Common Aggregates

8.3.1 Biller Record Aggregate <BillerRec>

The <BillerRec> aggregate is widely used to provide summary-level information about a Biller.

The elements, <BillerStatus>, <EffDt>, <BSPReferTo>, <SPName>, and <OrgContact> support the case where a Biller discontinues usage of a BSP and includes "forwarding information" if available.

Tag	*Type*	*Usage*	*Description*
<BillerId>	Aggregate	Required Repeating	Biller Identification Aggregate.
<BillerInfo>	Aggregate	Required	
<BillerStatus>	Aggregate	Optional	
<BSPReferTo>	Aggregate	Optional	BSP Refer to Aggregate. The new BSP for this Biller, if known.
<SPName>	Identifier	Optional	Service Provider Name. Name of replacement BSP.
<OrgContact>	Aggregate	Optional	Contact information for replacement BSP.

Tag	Type	Usage	Description
</BSPReferTo>			

8.3.1.1 Biller Identification <BillerId>

The <BillerId> aggregate is widely used to uniquely identify a Biller within a BSP (i.e., the <SPName> would be the BSP).

The <StdPayeeId> is the Biller's ID as known to the CPP; i.e., the <SPName> would be the CPP. Billers may be known differently by different organizations, so when both <BillerId> and <StdPayeeId> are used in a message (such as in those messages that use the <PresAcctId> aggregate) the <BillerId> is the Biller's ID as known by the BSP and the <StdPayeeId> is the Biller's ID as known by the CPP. The presence of these two IDs may help in resolving the identification of Billers known differently by other organizations.

Tag	Type	Usage	Description
<SPName>	Identifier	Required	Service Provider Name. Used to qualify <BillerNum>. This is the name of the BSP that assigned <BillerNum>.
<BillerNum>	Identifier	Required	Biller Number. Assigned by the Pay/Presentment provider. Cannot be changed by the client.

8.3.1.2 Biller Information Aggregate <BillerInfo>

Tag	Type	Usage	Description
<Name>	C-40	Required	Biller Name. Assigned by the service provider. Cannot be changed by the client.
<BillerContact>	Aggregate	Optional	Biller Contact Information.
<IndustId>	Aggregate	Optional	Industry Identifier.
<SecretPrompt>	Aggregate	Optional Repeating AND	Secret Prompt Aggregate. If omitted, the biller does not require the customer to enter any secrets for client enrollment.
<CryptType>	Open Enum	Optional Repeating AND	Encryption type to indicate encryption used for transmitting authentication information. The Biller may specify one or more encryption types that it accepts. Note that NONE is a valid value for <CryptType>.
<HistRetentionDays>	Long	Optional	Number of days that Bill Summary and Bill Detail information is available for inquiries. The Bill Detail information may be available for a longer period of time.
<CSPCustInfoReq>	Boolean	Optional	CSP Customer name and address information Required. If TRUE, the Biller requires that the Customer name and address be sent with an account activation. IF FALSE or omitted, the customer name and address are not required in the account activation.
<BillerPayInfo>	Aggregate	Optional	Biller Pay Information Aggregate.
<LogoURL>	URL	Optional	Logo URL. URL of the biller's logo.
<Logo>	Binary	Optional	Biller Logo. If the client requested images, the logo should be included here in this response.

Tag	Type	Usage	Description
<BillerEnrollURL>	URL	Optional	Biller Information URL. URL of human-readable description of additional information the biller would like the customer to have with regard to signing up. The resource may also include an interactive session to verify a customer's identity and eligiblity to receive bills for an account. If successful, the session results in a security token or password that the customer may use in response to a <SecretPrompt> when activating the account for bill presentment.
<BillerAcctIdInfo>	Aggregate	Optional	Biller Account Information Aggregate. It provides additional information to the customer to assist in entry of the customer's account numbers with the biller. See Section 13.3.6.
<DiscReqd>	Boolean	Optional	Disclosure Presentment Required. If TRUE, the biller requires that their disclosure be presented to the customer prior to account activation. If FALSE, the presentment of the disclosure is not required prior to the account activation request. This may be FALSE if no disclosure is required or when the biller wishes to return the disclosure in the account activation response.
<DiscDt>	Date	Optional	Date Last Disclosure Change.
<OptSupt>	Open Enum	Optional Repeating	Options Supported. Defined values:

8.3.1.3 Biller Status Aggregate <BillerStatus>

The <BillerRec> aggregate is widely used to provide summary-level information about a Biller.

The elements, <BillerStatus>, <EffDt>, <BSPReferTo>, <SPName>, and <OrgContact> support the case where a Biller discontinues usage of a BSP and includes "forwarding information" if available.

Tag	Type	Usage	Description
<BillerStatusCode>	Closed Enum	Required	Biller Status Code. Valid Values: Available, AvailPend, Deleted, DelPend.
<EffDt>	DateTime	Optional *but see Description*	Date Time Effective. The Date the <BillerStatus> was changed or, in the case of DeletePending, the date the delete should take effect. Required if <BillerStatusCode> = *xxx*Pend.
<StatusModBy>	Open Enum	Optional	Status Modified By. If present, indicates who modified the Biller Status Code. Possible values are: Customer, CSP, CSPSR, BSP, BSPSR, CPP, CPPSR, BPP, and BPPSR.
<Memo>	C-255	Optional	Memo Information. To provide additional information about the status.

8.3.2 Biller Account Identification Information Aggregate <BillerAcctIdInfo>

The <BillerAcctIdInfo> aggregate specifies the format of valid account numbers acceptable to the biller.

Tag	Type	Usage	Description
<AcctFormat>	NC-1024	Optional	Account Format. Regular expression describing the account number format. The definition and behavior of "Regular Expression" is per IEEE Std 1003.2-1992 (POSIX.2). General definition may be found at http://www.ciser.cornell.edu/info/regex.html

Tag	Type	Usage	Description
<AcctMask>	C-32	Optional Repeating	Account Mask. String describing the edit mask to identify a valid billing account number <BillingAcct> for that biller. The client uses the account edit mask to assist the user in entering the account number.
<AcctHelpMsg>	C-	Optional	Account Help Message. Human-readable message that the client may display to assist the customer in entering his or her account number.
<AcctRestrictMsg>	C-1024	Optional	Account Restriction Message. Human-readable description of any restrictions on who may sign up with this biller. Enforcement of any restrictions is by the biller or the biller's agent during the <SvcAcctAddRq>. Other service providers are only responsible for communicating this information.
<AcctValidateURL>	URL	Optional	URL for Account Validation. URL for validation. The client application may use this to validate the customer's account number.

8.3.2.1 Account Number Validation

Servers may implement a lightweight CGI (or equivalent) to validate account numbers. The URL provided in the <AcctValidateURL> may be accessed with an HTTP GET with three arguments: <BillerId>, <AcctId> and <PostalCode>. The URL should respond with a text file that includes the following values:

Status: (Required)

- Error: An error condition (wrong number of parameters, Database error, etc.). Clarifying text may accompany the error status.
- PASS: The account number is in an acceptable form for this biller (this does not guarantee that the account will be accepted for the service).
- FAIL: The account number does not correspond to an acceptable account number for this biller. Clarifying text may accompany the failed status.

Account: (Optional) The preferred format or version of the account number presented in the request.

Heading: (Optional) Additional text to help explain problems to end-users.

Example:

```
<AcctValidateURL> = http://testit.com/validate.cgi

Client application uses HTTP GET with
"http://testit.com/validate.cgi?billerid=5454&accountnumber=123-456-
7890&customerpostalcode=12345"
```

The server would respond with one of these:

- Error:

  ```
  Content-type: text/plain

  <Status>Error</Status>

  <Heading>The server is unable to process your request at this time. Please
  resubmit.</Heading>
  ```
- Failure:

```
Content-type: text/plain

<Status>FAIL</Status>

<Heading>123-456-7890 does not appear to be a valid account number</Heading>
```

- Passed:

```
Content-type: text/plain

<Status>passed</Status>

<AcctId>1234567890</AcctId>
```

8.3.3 Bill Record <BillRec>

The bill record aggregate <BillRec> provides the ability to deliver bill-related information, including bill summary information about a single bill, a billing statement, a notification (textual information sent from the biller to communicate information about the bill presentment service), or an invoice, depending on the value in the <BillType> element. The Bill Summary may include such information as amount due, date due, and pointers to more information; e.g., a set of URLs that may used to access bill detail and other information. The actual elements used are likely to vary depending on the Bill Type. For example, with <BillType> Notice, the biller may send <Memo> data but none of the other optional elements.

Tag	*Type*	*Usage*	*Description*
<BillId>	UUID	Required	Identifier for this bill within the BSP.
<BillInfo>	Aggregate	Required	Bill Information Aggregate.
<BillType>	Open Enum	Required	Bill Type. Defined values: Bill, Statement, Notice, Invoice.
<PresAcctId>	Aggregate	Required	Bill Presentment Account Detail Aggregate.
<Memo>	C-255	Optional	Biller-information displayed to user by the biller. This may be used for notice information when <BillType> = Notice.
<BillSummAmt>	Aggregate	Optional Repeating	Bill Summary Amount Aggregate. Used for any currency amounts that are being presented in the Bill Summary. Note: If <BillType> = Bill, at least one occurrence of this aggregate must be present where the amount is a payable amount (<BillAmtType> = Payable).
<BillSummAmtId>	Identifier	Optional	Bill Summary Amount Identifier. Biller's identifier for this currency amount. This may be returned in the payment message to identify the type of amount being paid by the consumer. This value needs to be provided by the Biller if the intent is to allow the consumer to designate the particular amount being paid (using <PaySummAmt> in <PmtInfo> aggregate).
<BillSummAmtCode>	Open Enum	Optional	Bill Summary Amount Type. Indicates the type of amount being specified in <CurAmt>, using a standard list of billing amounts that are machine-readable by the client and may be used for internal processing. The short description <ShortDesc> should be used to describe the amount in a display to the consumer. Defined values: TotalAmtDue, MinAmtDue, MaxAmtDue, LateAmtDue, DiscAmtDue, MonthlyAmt, QuarterlyAmt, SemiAnnualAmt, AnnualAmt, ExtdAmt, PrevBal, Charges, Credits, StmtBal, LateChg, FinanceChg, Tip, Principal, Interest, Escrow, PMI, Donation
<ShortDesc>	C-15	Required	Short Description. A short description of the amount specified in <CurAmt> within this aggregate. To be used for display to the consumer.

Tag	*Type*	*Usage*	*Description*
<Desc>	C-80	Optional	Description. A longer description of the amount specified in <CurAmt>. To be used when the short description <ShortDesc> may be insufficient to clearly describe the amount.
<Memo>	C-255	Optional	Memo. Additional information about the amount specified in <CurAmt>. This may be used to further describe terms or instructions that may apply to the amount specified.
<CurAmt>	Currency Amount	Required	Currency Amount. The amount being specified as described by the short description. This amount may be a zero or negative value.
<BillSummAmtType>	Closed Enum	Required	Bill Summary Amount Type. Indicates the type of amount specified in <CurAmt>. Note that the client must treat amounts that are Payable as mutually exclusive (i.e., the consumer may select only one). Amounts that are Supplemental are not mutually exclusive (i.e., the consumer may select more than one for indicating breakdown of an additional payment). Valid values: • Payable—an amount that may be paid by the consumer; the consumer may use this category on a payment to indicate the intent of the amount being paid, when multiple payable amounts are presented) • Supplemental—additional payment; a category that may be used for assigning an additional payment beyond the payable amount presented) • InfoOnly—an amount that is being presented for information only and is not used in the payment.
<BillSummSubAmt>	Aggregate	Optional Repeating	Bill Summary Sub-Amount. Allows for the specification of a breakdown of an amount specified in <CurAmt> within the higher level aggregate <BillSummAmt>.
<BillSummAmtId>	Identifier	Required	Bill Summary Amount Identifier. Biller's identifier for this currency sub-amount. Used in a payment message to allocate a portion of a total payment amount to this sub-amount.
<ShortDesc>	C-15	Required	Short Description. A short description of the amount specified in <CurAmt> within this aggregate. To be used for display to the consumer.
<Desc>	C-80	Optional	Description. A longer description of the amount specified in <CurAmt>. To be used when the short description <ShortDesc> may be insufficient to clearly describe the amount.
<Memo>	C-255	Optional	Memo. Additional information about the amount specified in <CurAmt>. This may be used to further describe terms or instructions that may apply to the amount specified.
<CurAmt>	Currency Amount	Required	Currency Amount. The amount being specified as described by the short description. This amount may be a zero or negative value.
<AllocateAllowed>	Boolean	Optional	Allocation Allowed Indicator. If TRUE, the <BillSummAmtId> for the amount specified may be used to indicate the allocation breakdown of the total payment amount within the <PayInfo> aggregate in a payment message. If FALSE or omitted, the amount specified is for information only.
</BillSummSubAmt>			
</BillSummAmt>			
<PmtDueDt>	Date	Optional	Payment due date.
<BillDt>	Date	Required	Bill date.
<OpenDt>	Date	Optional	Opening statement date.

Tag	Type	Usage	Description
<CloseDt>	Date	Optional	Closing statement date.
<PmtInst>	Aggregate	Optional Repeating	Payment Instrument Aggregate. Types of payment that the biller accepts via the Pay provider. This aggregate may be used to override types of payments options indicated previously in the <BillerRec>. Note that a restriction here may result in no valid payment type being available to the customer, which may result in the CSP sending a <BillStatusModRq> to the BSP indicating that the bill is Unpayable.
<NotifyReqd>	Boolean	Required	If TRUE, the server requires that the client send a <BillStatusModRq> for each change of status of either the bill or its payment. The server may not send TRUE if the client did not set <NotifyWilling> to TRUE in <BillInqRq>.
<MustViewDtlToPay>	Boolean	Optional	If TRUE, the customer must view the bill detail information, in addition to the bill summary information, prior to initiating payment for this bill. If FALSE or omitted, the customer may initiate payment after viewing the bill summary information. Bill detail is available through the <URL> specified within the <StmtImage> aggregate. If <MustViewDtlToPay> is TRUE, the CSP must require the customer to "click" on the <StmtImage> <URL>, before the customer is permitted to "click" on a "pay button."
<StmtImage>	Aggregate	Optional	Statement image aggregate.
<BillRefInfo>	NC-80	Optional	Biller-defined text to include with the payment, for the biller's Accounts Receivable reconciliation. It is sent with electronic payment requests.
</BillInfo>			
<BillStatus>	Aggregate	Optional	Bill Status Aggregate.
<BillPmtStatus>	Aggregate	Optional	Bill Payment Status Aggregate.

8.3.3.1 Bill Information Aggregate <BillInfo>

Tag	Type	Usage	Description
<BillType>	Open Enum	Required	Bill Type. Defined values: Bill, Statement, Notice, Invoice.
<PresAcctId>	Aggregate	Required	Bill Presentment Account Detail Aggregate.
<Memo>	C-255	Optional	Biller-information displayed to user by the biller. This may be used for notice information when <BillType> = Notice.
<BillSummAmt>	Aggregate	Optional Repeating	Bill Summary Amount Aggregate. Used for any currency amounts that are being presented in the Bill Summary. Note: If <BillType> = Bill, at least one occurrence of this aggregate must be present where the amount is a payable amount (<BillAmtType> = Payable).
<BillSummAmtId>	Identifier	Optional	Bill Summary Amount Identifier. Biller's identifier for this currency amount. This may be returned in the payment message to identify the type of amount being paid by the consumer. This value needs to be provided by the Biller if the intent is to allow the consumer to designate the particular amount being paid (using <PaySummAmt> in <PmtInfo> aggregate).

Tag	*Type*	*Usage*	*Description*
<BillSummAmtCode>	Open Enum	Optional	Bill Summary Amount Type. Indicates the type of amount being specified in <CurAmt>, using a standard list of billing amounts that are machine-readable by the client and may be used for internal processing. The short description <ShortDesc> should be used to describe the amount in a display to the consumer. Defined values: TotalAmtDue, MinAmtDue, MaxAmtDue, LateAmtDue, DiscAmtDue, MonthlyAmt, QuarterlyAmt, SemiAnnualAmt, AnnualAmt, ExtdAmt, PrevBal, Charges, Credits, StmtBal, LateChg, FinanceChg, Tip, Principal, Interest, Escrow, PMI, Donation
<ShortDesc>	C-15	Required	Short Description. A short description of the amount specified in <CurAmt> within this aggregate. To be used for display to the consumer.
<Desc>	C-80	Optional	Description. A longer description of the amount specified in <CurAmt>. To be used when the short description <ShortDesc> may be insufficient to clearly describe the amount.
<Memo>	C-255	Optional	Memo. Additional information about the amount specified in <CurAmt>. This may be used to further describe terms or instructions that may apply to the amount specified.
<CurAmt>	Currency Amount	Required	Currency Amount. The amount being specified as described by the short description. This amount may be a zero or negative value.
<BillSummAmtType>	Closed Enum	Required	Bill Summary Amount Type. Indicates the type of amount specified in <CurAmt>. Note that the client must treat amounts that are Payable as mutually exclusive (i.e., the consumer may select only one). Amounts that are Supplemental are not mutually exclusive (i.e., the consumer may select more than one for indicating breakdown of an additional payment). Valid values: • Payable—an amount that may be paid by the consumer; the consumer may use this category on a payment to indicate the intent of the amount being paid, when multiple payable amounts are presented) • Supplemental—additional payment; a category that may be used for assigning an additional payment beyond the payable amount presented) • InfoOnly—an amount that is being presented for information only and is not used in the payment.
<BillSummSubAmt>	Aggregate	Optional Repeating	Bill Summary Sub-Amount. Allows for the specification of a breakdown of an amount specified in <CurAmt> within the higher level aggregate <BillSummAmt>.
<BillSummAmtId>	Identifier	Required	Bill Summary Amount Identifier. Biller's identifier for this currency sub-amount. Used in a payment message to allocate a portion of a total payment amount to this sub-amount.
<ShortDesc>	C-15	Required	Short Description. A short description of the amount specified in <CurAmt> within this aggregate. To be used for display to the consumer.
<Desc>	C-80	Optional	Description. A longer description of the amount specified in <CurAmt>. To be used when the short description <ShortDesc> may be insufficient to clearly describe the amount.
<Memo>	C-255	Optional	Memo. Additional information about the amount specified in <CurAmt>. This may be used to further describe terms or instructions that may apply to the amount specified.

Tag	Type	Usage	Description
<CurAmt>	Currency Amount	Required	Currency Amount. The amount being specified as described by the short description. This amount may be a zero or negative value.
<AllocateAllowed>	Boolean	Optional	Allocation Allowed Indicator. If TRUE, the <BillSummAmtId> for the amount specified may be used to indicate the allocation breakdown of the total payment amount within the <PayInfo> aggregate in a payment message. If FALSE or omitted, the amount specified is for information only.
</BillSummSubAmt>			
</BillSummAmt>			
<PmtDueDt>	Date	Optional	Payment due date.
<BillDt>	Date	Required	Bill date.
<OpenDt>	Date	Optional	Opening statement date.
<CloseDt>	Date	Optional	Closing statement date.
<PmtInst>	Aggregate	Optional Repeating	Payment Instrument Aggregate. Types of payment that the biller accepts via the Pay provider. This aggregate may be used to override types of payments options indicated previously in the <BillerRec>. Note that a restriction here may result in no valid payment type being available to the customer, which may result in the CSP sending a <BillStatusModRq> to the BSP indicating that the bill is Unpayable.
<NotifyReqd>	Boolean	Required	If TRUE, the server requires that the client send a <BillStatusModRq> for each change of status of either the bill or its payment. The server may not send TRUE if the client did not set <NotifyWilling> to TRUE in <BillInqRq>.
<MustViewDtlToPay>	Boolean	Optional	If TRUE, the customer must view the bill detail information, in addition to the bill summary information, prior to initiating payment for this bill. If FALSE or omitted, the customer may initiate payment after viewing the bill summary information. Bill detail is available through the <URL> specified within the <StmtImage> aggregate. If <MustViewDtlToPay> is TRUE, the CSP must require the customer to “click” on the <StmtImage> <URL>, before the customer is permitted to “click” on a “pay button.”
<StmtImage>	Aggregate	Optional	Statement image aggregate.
<BillRefInfo>	NC-80	Optional	Biller-defined text to include with the payment, for the biller’s Accounts Receivable reconciliation. It is sent with electronic payment requests.

8.3.3.1.1 Statement Image <StmtImage>

The <StmtImage> aggregate provides one or more URLs that point to a fully rendered image of the bill, in HTML. The <URL> should include enough information for the HTTP server to authenticate the client requesting the statement image. For security reasons, the authentication information embedded within the URLs should expire after a period of time deemed prudent by the service provider. Information about the authentication information expiration date and time is included in <ExpDt>.

<ImageURL> accesses the complete bill image. This URL may contain navigation to other sites or to other pages of bill images at the same site.

To support off-line viewing of the bill, the server may provide one or more additional URLs. Each <PrefetchURL> points to a local Web page.

Tag	Type	Usage	Description

Tag	Type	Usage	Description
<ImageURL>	URL	Required	URL address for retrieving an image of the complete bill encoded as HTML. This may be cached by the client for later display, or it may be viewed live directly from the Web.
<PrefetchURL>	URL	Optional Repeating	List of URLs required in order to display an HTML image of the complete bill, to support off-line viewing.
<ExpDt>	DateTime	Optional	Date/Time after which embedded authentication token expires. If absent, the embedded token never expires.

8.3.3.2 Bill Status Aggregate <BillStatus>

Tag	Type	Usage	Description
<BillStatusCode>	Open Enum	Required	Bill Status Code. Defined values are: New, Delivered, Viewed, Retired, Withdrawn, and Undeliverable.
<EffDt>	DateTime	Optional	Effective Date Time. The date and time the Bill Status became effective.
<StatusModBy>	Open Enum	Optional	Status Modified By. If present, indicates who modified the Bill Status Code. Possible values are: Customer, CSP, CSPSR, BSP, BSPSR, CPP, CPPSR, BPP, and BPPSR.
<Memo>	C-255	Optional	Memo Information. To provide additional information about the status.

8.3.3.3 Bill Payment Status Aggregate <BillPmtStatus>

Tag	Type	Usage	Description
<PmtId>	Identifier	Optional	Payment Identifier. The ID of the payment associated with the bill.
<BillPmtStatusCode>	Open Enum	Required	Bill Payment Status Code. Valid values are: None, AutoPay, Scheduled, Processed, PaidOutOfBand, Cancelled, Unpayable, and Posted.
<EffDt>	DateTime	Optional	Effective Date Time. The date and time the Bill Payment Status became effective.
<StatusModBy>	Open Enum	Optional	Status Modified By. If present, indicates who modified the Bill Status Code. Possible values are: Customer, CSP, CSPSR, BSP, BSPSR, CPP, CPPSR, BPP, and BPPSR.
<Memo>	C-255	Optional	Memo Information. To provide additional information, for example, if the payment was rejected, it may contain the reason for the rejection.

8.4 Presentment Service Message Detail

The Presentment Service contains messages to search for billers and to obtain bills.

Typically, the client periodically requests a list of bills from the BSP. The BSP responds with a list of bills; each bill contains summary data such as the due date and amount due. For each bill, the BSP might also return a <URL> to a Web site that contains an HTML-rendered version of the bill.

The server must include the BSP's <SPName> in the <BillerRec> aggregate for each biller sent in the <BillerInqRs>. The client may then use this <SPName> to activate accounts <SvcAcctAddRq>, request bills <BillInqRq>, and change status of bills <BillStatusModRq>. Since the bills for a specific customer may

originate from multiple BSPs, it is the client software's responsibility to ensure that the correct <SPName> is used in each message that it originates.

8.4.1 Biller Inquiry

The Biller Inquiry message enables a client to retrieve a list of all Billers known to the BSP that meets certain selection criteria.

Note that the Biller directory timestamp <BillerDirDt> selection criterion allows the CSP to request all directory entries that have been added or changed since a point in time.

8.4.1.1 Request <BillerInqRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate.
<BillerId>	Aggregate	Optional Repeating	Biller Identification. This field is used as a selection criterion.
<Name>	C-40	Optional Repeating	Biller Name. Assigned by the server at the time the Biller is first added. Cannot be changed by the client. This field is used as a selection criterion.
<PostAddr>	Aggregate	Optional Repeating	Biller Postal Address Aggregate.
<Phone>	Phone Number	Optional Repeating	Customer Service Telephone Number. This field is used as a selection criterion against all customer service phone numbers associated with billers.
<PostalCode>	C-11	Optional Repeating	Postal Code. This is the postal code of the billing account. This field is used as a selection criterion: (1) It may be used to limit the search to billers doing business within a limited geography, or (2) It may be used to identify the correct legal/entity or remittance address such as the cable company for a specific city, e.g. TCI- Sunnyvale CA.
<UpDt>	Timestamp	Optional	Biller Directory update timestamp. This is the time supplied by the server. If present, <BillerInqRs> must include at least those Billers whose information has changed or been added since <UpDt>.
<IndustId>	Aggregate	Optional Repeating	Industry Identifier This field is used as a selection criterion.
<IncBillerContact>	Boolean	Optional	Include Biller Contact Information. If TRUE, the <BillerContact> aggregate is returned for each biller; otherwise, it is not returned.
<IncImages>	Boolean	Optional	Include Images. If TRUE, the client requests that images <Logo>s be returned.

8.4.1.2 Response <BillerInqRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<BillerId>	Identifier	Optional Repeating Echoed	Biller Identification Aggregate.
<Name>	C-40	Optional Repeating Echoed	Biller Name.
<PostAddr>	Aggregate	Optional Repeating Echoed	Biller Postal Address Aggregate.
<Phone>	Phone Number	Optional Repeating Echoed	Customer Service Telephone Number.
<PostalCode>	C-11	Optional Repeating Echoed	Postal Code. This is the postal code of the billing account.
<UpDt>	Timestamp	Optional Echoed	Biller Directory timestamp.
<IndustId>	Aggregate	Optional Repeating Echoed	Industry Identifier
<IncBillerContact>	Boolean	Optional Echoed	Include Biller Contact Information.
<IncImages>	Boolean	Optional Echoed	Include Images.
<IncBillerContact>	Boolean	Optional Echoed	Include Biller Detail.
<IncImages>	Boolean	Optional Echoed	Include Images.
<NewUpDt>	Timestamp	Optional	New Biller Directory Timestamp. This is the response timestamp generated by the server.
<BillerRec>	Aggregate	Optional Repeating	Biller Information Aggregate.

8.4.2 Bill Inquiry

<BillInqRq> retrieves bills or counts of bills from the BSP. The BSP returns a <BillInqRs> that contains a list of zero or more bills, or counts of bills that match specific selection criteria.

The client requests bills from a BSP by using one or more selection criteria, including bill creation date range. To specify the date range, clients use <StartDt> and <EndDt>, which the server compares to <BillDt> within the <BillRec> aggregate.

The BSP returns information sufficient to identify the biller and provide the amount due, due date, and remittance information so that a payment may be made to the biller. The BSP does not provide a viewable form of the bill but may return a URL to an HTML rendering of the bill.

8.4.2.1 Request <BillInqRq>

Tag	Type	Usage	Description
<RqUID>	UUID	Required	Request Identifier. Sent by a client as a universally unique identifier for the message. Used to correlate responses with requests.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<RecCtrlIn>	Aggregate	Optional	Records Control Input Aggregate. It is used in inquiry request messages to allow the client to specify a maximum number of records that the server may return.
<SPName>	Identifier	Required	Service Provider Name.
<SelRangeDt>	Aggregate	Optional	Selection Range Date Aggregate.
<BillId>	UUID	Optional Repeating	Bill Identifier. This field is used as a selection criterion.
<BillerId>	Aggregate	Optional Repeating	Biller Identifier Aggregate. This field is used as a selection criterion.
<BillType>	Open Enum	Optional Repeating	Bill Type. This field is used as a selection criterion.
<BillStatusCode>	Open Enum	Optional Repeating	Bill Status Code. This field is used as a selection criterion.
<BillPmtStatusCode>	Open Enum	Optional Repeating	Bill Payment Status Code. This field is used as a selection criterion.
<NotifyWilling>	Boolean	Optional	Client Willing to Notify. If TRUE, the client is prepared to send notifications of changes to bill status, if desired. If FALSE, the client cannot send notifications of bill status changes.
<IncCounts>	Boolean	Optional	Include Counts of Bills. If TRUE, the client is requesting that the number of bills in each status included in the selection criteria <BillStatusCode>, <BillPmtStatusCode> is returned in the <BillCounts> aggregate in the response.
<IncSummary>	Boolean	Optional	Include Bill Summaries. If TRUE, the client is requesting that the bill summaries for each bill as specified in the request be returned in the response, utilizing the <BillRec> aggregate.

8.4.2.2 Response <BillInqRs>

The <BillInqRs> may contain zero or more bill summaries <BillRec>. Each bill summary corresponds to a (usually monthly) bill. The response may contain either counts of bills or the collection of the bill summaries.

Tag	Type	Usage	Description
<Status>	Aggregate	Required	Response Status Aggregate.
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.

Tag	*Type*	*Usage*	*Description*
<RecCtrlOut>	Aggregate	Optional *but see Description*	Records Control Output Aggregate. Required if <RecCtrlIn> was provided in the request and the server supports Records Control.
<SelRangeDt>	Aggregate	Optional Echoed	Selection Range Date Aggregate.
<BillId>	UUID	Optional Repeating Echoed	Bill Identifier.
<BillerId>	Aggregate	Optional Repeating Echoed	Biller Identifier Aggregate.
<BillStatusCode>	Open Enum	Optional Repeating Echoed	Bill Status Code.
<BillPmtStatusCode>	Open Enum	Optional Repeating Echoed	Bill Payment Status Code.
<NotifyWilling>	Boolean	Optional Echoed	Client Willing to Notify.
<IncCounts>	Boolean	Optional Echoed	Include Counts of Bills.
<IncSummary>	Boolean	Optional Echoed	Include Bill Summaries.
<BillRec>	Aggregate	Optional Repeating	Bill Record Aggregate. This is a list of bills that match the selection criteria.
<BillCounts>	Aggregate	Optional Repeating	Bill Counts Aggregate.
<BillStatusCounts>	Aggregate	Optional Repeating	Bill Status Counts. The count(s) of all the bills for that customer, which may be provided by a given status(s). Note, if no selection criteria <BillStatusCode>, <BillPmtStatusCode> are specified when <IncCounts>=TRUE, counts are returned for every status with a non-zero count.
<BillStatusCode>	Open Enum	Required	Bill Status Code.
<Count>	Long	Required	Count of Bills with the given Bill Status Code.
</BillStatusCounts>			
<BillPmtStatusCounts>	Aggregate	Optional Repeating	Bill Payment Status Counts. The count(s) of all the bill payment statuses for that customer, which may be provided by a given status(s). Note selection criteria is ignored when <IncCounts>=TRUE.
<BillPmtStatusCode>	Open Enum	Required	Bill Payment Status Code.
<Count>	Long	Required	Count of Bills with the given Bill Payment Status Code.
</BillPmtStatusCounts>			
</BillCounts>			

8.4.3 Bill Status Modify

The BSP may request the client to send notifications of various state changes for the bill of the associated payment by setting <NotifyReqd> = TRUE in the <BillRec> aggregate (see Section 15.3.2), if the client has

indicated that it is capable of sending notifications; i.e., <NotifyWilling> = TRUE within <BillInqRq>. The following table indicates which entity(s) sets each bill and bill payment status.

Status Code	*Recognized/Set by*
<BillStatusCode> (Status of bill):	
New	BSP/Biller (initially)
Delivered	BSP/Biller
Viewed	Customer or CSP by inference
Retired	Customer
Withdrawn	BSP/Biller
Undeliverable	CSP
<BillPmtStatusCode> (Status of a bill payment)	
None	BSP/Biller
AutoPay	BSP/Biller
Scheduled	CSP/CPP
Processed	CSP/CPP
Posted	BSP/Biller
PaidOutOfBand	Customer or CSP/CPP
Cancelled	Customer or CSP/CPP
Unpayable	CSP/CPP

Setting <BillStatusCode>=Viewed tells the BSP that the CSP has presented the specified bills to the customer. This is a stronger statement than acknowledging that the client has received the bills <BillStatusCode>=Delivered, specifically when the client software implements the pre-fetching or (pull) model.

However, IFX does not define the meaning of "presenting to the customer." In particular, receipt of a <BillStatusCode>=Viewed by the BSP is not intended to have any legal significance. The specification also does not define the maximum elapsed time between the presentation of the bill and sending a notification.

8.4.3.1 Request <BillStatusModRq>

Tag	*Type*	*Usage*	*Description*
<RqUID>	UUID	Required	Request Identifier.
<CustId>	Aggregate	Optional	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<BillId>	UUID	Required	Bill Identifier. Identifies the bill from the given biller.
<BillStatus>	Aggregate	Required OR	Bill Status Aggregate.
<BillPmtStatus>	Aggregate	Required OR	Bill Payment Status Aggregate.

8.4.3.2 Response <BillStatusModRs>

Tag	*Type*	*Usage*	*Description*
<Status>	Aggregate	Required	Response Status Aggregate.

Tag	Type	Usage	Description
<RqUID>	UUID	Required Echoed	Request Identifier.
<CustId>	Aggregate	Optional Echoed	Customer Identification Aggregate. This is the identifier of the user initiating the request.
<BillId>	UUID	Required Echoed	Bill Identifier.
<BillStatus>	Aggregate	Required OR Echoed	Bill Status Aggregate.
<BillPmtStatus>	Aggregate	Required OR Echoed	Bill Payment Status Aggregate.
<CSPRefId>	Identifier	Optional	Financial Institute Reference Identifier.
<SPRefId>	Identifier	Optional	Service Provider Reference Identifier.

8.5 Bill Presentment Service Profile <PresSvcProfInfo>

This section defines the profile aggregate for the Bill Presentment Service. This profile aggregate should be included in the <SvcProfInqRs> response for those servers that support the Bill Presentment Service.

Tag	Type	Usage	Description
<SvcCore>	Aggregate	Required	Service Core Aggregate. Information specified for every service.
<MsgSupt>	Open Enum	Required Repeating	Supported Messages. This is a list of messages that may be supported for Presentment. The convention is to use the name of the message without the Rq or Rs so that each message is only listed once. Valid values: BillerInq, BillInq, BillStatusMod.
<OptSupt>	Open Enum	Optional Repeating	Options Supported. Defined values: RecCtrl
<PrcSched>	Aggregate	Optional	Processing Schedule Aggregate. If omitted, the default processing schedule is assumed.

Index

<AcceptReqd> 5-8
<AcctBal>3-8, **3-12**, 5-32, 6-14, 6-15, 6-16, 6-17, 6-18
<AcctFormat> 8-7
<AcctHelpMsg> 8-7
<AcctId> 3-8, 3-9, 3-10, 8-8
<AcctInqRq> 6-2, **6-15**
<AcctInqRs> 6-2, **6-15**
<AcctKey> 3-9
<AcctMask> **3-1**, 7-10, 8-7
<AcctRestrictMsg> 8-8
<AcctTaxInfo> 6-25
<AcctTaxStatus> 6-23, 6-24
<AcctTaxType> 3-8
<AcctType> 3-9, 3-10, **3-14**, 3-15, 6-23, 6-24
<AcctValidateURL> 8-8
<Addr1> 3-7
<Addr2> 3-7
<Addr3> 3-7
<Addr4> 3-7
<Aging> 6-19
<AllocateAllowed> 8-3, 8-10, 8-12
<Amt> 2-10
<ATMCityState> 6-8
<ATMLocation> 6-7
<ATMOwner> 6-7
<ATMTrnInfo> 6-7
<AuthInfo> 5-16
<AuthSupt> 5-38
<AvailDt> 6-7
<BalInqRq> 6-2, 6-14
<BalInqRs> 6-2, **6-14**
<BalType> 3-12, 3-13
<BankAcctFromProf> 7-57
<BankAcctInfo> 3-7, **3-10**
<BankAcctRec> **3-7**, 5-7, 5-13, 5-32
<BankAcctStatus> 3-7, **3-11**
<BankAcctStatusCode> 3-11, 3-12
<BankAcctTaxInqRq> 6-3, **6-24**
<BankAcctTaxInqRs> 6-3, **6-25**
<BankAcctToProf> 7-57
<BankAcctTrnImgInqRq> **6-22**
<BankAcctTrnImgInqRs> **6-23**
<BankAcctTrnRec> **6-6**, 6-7, 6-8
<BankId> 3-8
<BankInfo> 3-9, 7-12
<BankSvc> **6-1**
<BankSvcProfInfo> **6-49**
<BaseSvc> **5-1**
<BaseSvcProfInfo> 2-21, **5-37**
<BillAmtType> 8-2, 8-9, 8-11
<BillCounts> 8-17, 8-18
<BillDt> 8-2, 8-10, 8-12, 8-16
<BillerAcctIdInfo> 3-1, 8-7
<BillerContact> **3-6**, 7-10, 7-12, 8-6, 8-15
<BillerDirDt> 8-14
<BillerEnrollURL> 8-6
<BillerId>3-3, 3-12, 7-1, 7-9, 7-12, 7-19, 7-32, 7-34, 8-5, **8-6**, 8-8, 8-15, 8-17, 8-18
<BillerInfo> 8-5, **8-6**
<BillerInqRq> 7-1, 7-19, 8-1, 8-5, **8-14**
<BillerInqRs> 8-1, 8-5, 8-14, 8-15
<BillerNum> 3-1, 3-11, 7-9, 8-6
<BillerPayee> 7-11, **7-12**
<BillerPayInfo> **3-6**, 3-21, 7-12, 8-6
<BillerRec>3-21, 5-3, 5-29, 5-30, 7-4, 7-10, **8-5**, 8-7, 8-10, 8-13, 8-14, 8-16
<BillerStatus> 8-5, **8-7**
<BillerStatusCode> 8-7
<BillId> 7-16, 8-9, 8-17, 8-19
<BillInfo> 7-6, 7-17, 8-9, **8-11**
<BillingAcct>3-1, 3-11, 3-12, 5-29, 7-4, 7-5, 7-10, 7-11, 7-17, 7-19, 7-20, 7-32, 7-34, 7-53, 7-54, 8-7
<BillingAddr> 2-13
<BillingName> 3-11
<BillInqRq>8-2, 8-5, 8-11, 8-13, 8-14, 8-16, 8-18
<BillInqRs> 3-11, 8-2, 8-5, 8-16, **8-17**
<BillPmtStatus> 2-16, 8-11, **8-14**, 8-19, 8-20
<BillPmtStatusCode>2-16, 8-14, 8-17, 8-18, 8-19
<BillPmtStatusCounts> 8-18
<BillRec>3-7, 3-21, 7-16, 8-2, 8-5, **8-8**, 8-16, 8-17, 8-18
<BillRefInfo> 7-6, 7-16, 8-5, 8-11, 8-13
<BillRetAddr> 3-6
<BillStatus> 8-11, **8-13**, 8-19
<BillStatusCode> 8-14, 8-17, 8-18, 8-19
<BillStatusCounts> 8-18
<BillStatusModRq>2-16, 5-3, 8-5, 8-10, 8-11, 8-13, 8-14, **8-19**
<BillStatusModRs> 8-5, **8-19**
<BillSummAmt>8-2, 8-3, 8-4, 8-9, 8-10, 8-11, 8-12
<BillSummAmtCode> 8-9, 8-11
<BillSummAmtId>7-6, 7-17, 8-9, 8-10, 8-11, 8-12
<BillSummAmtType> 8-2, 8-4, 8-10, 8-12
<BillSummSubAmt> 8-3, 8-4, 8-10, 8-12
<BillType> 5-14, 8-2, 8-9, 8-11, 8-17
<BinData> 2-5
<BinLength> 2-5
<BranchId> 3-8
<BranchName> 3-9
<Brand> 3-10, 3-22
<BSPReferTo> 8-5, 8-7
<BusName> 7-11
<CanWind> 6-50

<CardAcctId>3-7, **3-9**, 3-12, 5-9, 5-10, 6-1, 6-11, 6-14, 6-15, 6-16, 6-17, 6-18, 6-19, 6-20, 6-21, 6-22, 6-23, 6-25, 6-26, 6-27, 6-28, 6-29, 6-30, 6-31, 6-32
<CardAcctIdFrom>2-14, **3-9**, 6-10, 6-26, 6-36, 6-37, 6-40, 6-44, 6-46, 6-47, 7-3, 7-11, 7-13, 7-16, 7-32, 7-34, 7-53, 7-54
<CardAcctIdTo> **3-9**, 6-7, 6-11, 6-36, 6-37
<CardAcctTrnInqRq> 6-2
<CardAcctTrnInqRs> 6-2
<CardMagData> 3-9, **3-10**, 4-3
<CascadeDel>2-16, 5-19, 5-25, 5-26, 5-31, 6-43, 7-22, 7-23, 7-52
<CaseSen> 5-38
<Category>6-11, 7-11, 7-13, 7-32, 7-34, 7-53, 7-54
<CCAcctRec> **6-4**, 6-16
<CCAcctStmtInqRq> 6-2, 6-17, **6-18**
<CCAcctStmtInqRs> 6-2, **6-18**
<CCAcctStmtRec> 6-18
<CCAcctTrnInqRq> **6-21**
<CCAcctTrnInqRs> **6-21**
<CCAcctTrnRec> **6-8**, 6-19, 6-22
<CCMotoAcct> 3-9
<Certificate> 4-3
<CharType> 5-38
<ChgPswdFirst> 5-38
<ChkBkStyleId> 6-12, 6-47
<ChkClrDt> 7-32, 7-34
<ChkCount> 6-47
<ChkNum>2-13, 6-7, 6-9, 6-27, 7-15, 7-32, 7-34, 7-36
<ChkNumEnd> 6-8
<ChkNumStart> 6-8, 6-12
<ChkOrdAddRq> 6-4, **6-47**
<ChkOrdAddRs> 6-4, **6-48**
<ChkOrdId> 6-11
<ChkOrdInfo> 6-11, 6-48
<ChkOrdRec> 6-48
<ChkOrdStatus> 6-11, **6-12**
<ChkOrdStatusCode> 6-12
<ChkPrint> 6-12
<ChkRange>5-4, **6-8**, 6-19, 6-20, 6-27, 6-28, 6-29, 6-30, 6-31
<City> 3-7, 3-9
<ClientApp> **3-20**, 4-4, 7-13
<ClientDt> 4-3, 4-4
<CloseDt> 8-10, 8-13
<CollateralDesc> 6-5
<CompletedPmtCount> 6-5
<ContactType> 3-6
<ContentType> 2-5
<CorrectAction> 6-7
<Count> 3-15, 6-12, 6-13, 6-48, 6-49, 8-18
<Country> 3-7
<CreatedDt> 7-12
<CryptPswd> 4-1
<CryptSecret> 3-21
<CryptType>3-21, 4-1, 5-16, 5-29, 5-30, 5-38, 8-6
<CSPCustInfoReq> 8-6
<CSPRefId>**3-2**, 3-3, 5-24, 5-25, 5-26, 5-30, 5-31, 5-32, 5-36, 6-7, 6-9, 6-10, 6-20, 6-21, 6-22, 6-23, 6-28, 6-29, 6-33, 6-34, 6-35, 6-36, 6-37, 6-41, 6-42, 6-43, 6-44, 6-45, 6-48, 6-49, 7-13, 7-20, 7-22, 7-23, 7-28, 7-29, 7-30, 7-31, 7-33, 7-35, 7-46, 7-50, 7-51, 7-52, 7-53, 7-54, 8-20
<CurAmt>3-13, 3-14, 6-6, 6-7, 6-8, 6-9, 6-11, 6-17, 6-19, 6-23, 6-24, 6-25, 6-26, 7-6, 7-16, 7-17, 7-18, 7-39, 7-40, 7-47, 8-3, 8-9, 8-10, 8-11, 8-12
<CurCode> 2-10, 3-10, 4-4, 6-26
<CurConvertRule> 2-10, 6-9
<CurRate> 2-10, 6-9
<Cursor> 3-18, 3-19, 3-20
<CustAcctUse> 3-11, 5-7
<CustAddRq> 5-1, **5-15**, 5-38, 5-39
<CustAddRs> 5-1, 5-16, 5-21, 5-22
<CustAudRq> 5-2, **5-21**
<CustAudRs> 5-2, 5-16, **5-21**
<CustContact> **3-5**, 5-7, 6-23
<CustContactPref> 3-5
<CustDelRq> 5-1, **5-18**
<CustDelRs> 5-1, **5-19**, 5-21, 5-23
<CustDiscInqRq> 5-2, **5-36**
<CustDiscInqRs> 5-2, **5-37**
<CustDiscRec> **5-7**, 5-37
<CustDiscStatus> 5-8, 5-37
<CustDiscStatusCode> 5-8, 5-36, 5-37
<CustDiscStatusModRq> 5-2, 5-29, **5-35**
<CustDiscStatusModRs> 5-2, **5-36**
<CustId>2-3, 2-16, 2-22, 3-2, **3-4**, 3-17, 4-2, 4-3, 4-4, 4-5, 5-7, 5-8, 5-11, 5-16, 5-17, 5-18, 5-19, 5-20, 5-21, 5-22, 5-23, 5-24, 5-25, 5-26, 5-27, 5-28, 5-29, 5-30, 5-31, 5-32, 5-33, 5-34, 5-35, 5-36, 5-37, 6-14, 6-15, 6-16, 6-17, 6-18, 6-19, 6-20, 6-21, 6-22, 6-23, 6-24, 6-25, 6-26, 6-27, 6-28, 6-29, 6-30, 6-31, 6-32, 6-33, 6-34, 6-35, 6-36, 6-37, 6-38, 6-39, 6-40, 6-41, 6-42, 6-43, 6-44, 6-45, 6-46, 6-47, 6-48, 6-49, 7-19, 7-20, 7-21, 7-22, 7-23, 7-24, 7-25, 7-26, 7-27, 7-28, 7-29, 7-30, 7-31, 7-33, 7-35, 7-36, 7-37, 7-38, 7-39, 7-40, 7-41, 7-42, 7-43, 7-44, 7-45, 7-46, 7-47, 7-48, 7-49, 7-50, 7-51, 7-52, 7-53, 7-54, 7-55, 7-56, 8-15, 8-16, 8-17, 8-19
<CustIdInqRq> 5-1, **5-20**
<CustIdInqRs> 5-1, **5-20**
<CustInfo> 5-7, 5-16, 5-17, 5-19
<CustInqRq> 5-1, **5-19**
<CustInqRs> 5-1, **5-19**
<CustLangPref> 4-3, 4-4
<CustLoginId> **3-2**, 3-4, 5-20
<CustModRq> 3-2, 5-1, **5-17**
<CustModRs> 5-1, **5-17**, 5-21, 5-22
<CustMsgRec> 5-21, 5-22

<CustName> **3-4**, 3-5, 3-11, 5-7, 5-39, 6-12
<CustNameAddr> **3-5**, 5-29, 5-30
<CustNameReqd> 5-39
<CustPayeeAddRq> 7-3, 7-7, **7-20**
<CustPayeeAddRs> 7-7, **7-20**, 7-26, 7-27
<CustPayeeAudRq> 7-2, 7-7, **7-25**
<CustPayeeAudRs> 7-7, **7-25**
<CustPayeeDelRq> 2-14, 7-2, 7-7, 7-14, **7-23**
<CustPayeeDelRs> 7-7, **7-23**, 7-26, 7-27
<CustPayeeId>3-11, **7-2**, 7-7, 7-10, 7-16, 7-17, 7-18, 7-21, 7-22, 7-23, 7-24, 7-25, 7-26, 7-27, 7-28, 7-29, 7-32, 7-34, 7-51, 7-53, 7-54
<CustPayeeInfo>6-6, 6-7, 7-10, 7-16, 7-20, 7-21, 7-27, 7-28, 7-29
<CustPayeeInqRq> 7-2, 7-7, **7-23**
<CustPayeeInqRs> 7-7, **7-24**
<CustPayeeModRq>7-2, 7-7, **7-21**, 7-27, 7-29, 7-58
<CustPayeeModRs> 7-7, **7-21**, 7-26, 7-27
<CustPayeeMsgRec> 7-26, 7-27
<CustPayeeRec> **7-10**, 7-20, 7-21, 7-25
<CustPayeeSyncRq> 7-2, 7-7, **7-26**
<CustPayeeSyncRs> 7-7, **7-26**
<CustPayeeTypeModRq> 7-7, **7-22**
<CustPayeeTypeModRs> 7-7, **7-22**, 7-26, 7-27
<CustPayeMsgRec> 7-27
<CustPermId> **3-2**, 3-4, 3-11, 5-20
<CustPswd> **4-1**, 4-2, 4-3, 5-16, 5-17
<CustPswdModRq> 5-1, **5-17**
<CustPswdModRs> 5-1, **5-17**
<CustRec> **5-6**, 5-16, 5-17, 5-18, 5-19, 5-20
<CustStatus> 5-7, 5-18
<CustStatusCode> 5-7
<CustStatusModRq> 5-1, **5-18**
<CustStatusModRs> 5-1, **5-18**
<CustSvcAddRq> 5-2, **5-23**, 8-1
<CustSvcAddRs> 5-2, **5-23**, 5-27, 5-28
<CustSvcAudRq> **5-26**
<CustSvcAudRs> **5-26**
<CustSvcDelRq> 5-2, **5-25**
<CustSvcDelRs> 5-2, **5-26**, 5-27, 5-28
<CustSvcId> 5-8, **5-9**, 5-23, 5-24, 5-25, 5-26
<CustSvcInfo> 5-8, **5-9**, 5-23, 5-24
<CustSvcModRq> 5-2, 5-24
<CustSvcModRs> 5-2, **5-24**, 5-27, 5-28
<CustSvcMsgRec> 5-27, 5-28
<CustSvcRec>**5-8**, 5-15, 5-23, 5-24, 5-25, 5-26, 5-32, 5-33
<CustSvcStatus> 5-8, **5-9**, 5-25
<CustSvcStatusCode> 5-9
<CustSvcStatusModRq> 5-24
<CustSvcStatusModRs> **5-25**
<CustSvcSyncRq> **5-27**
<CustSvcSyncRs> **5-28**
<CustSyncRq> **5-22**
<CustSyncRs> 5-16, **5-22**
<CustType> 5-7
<CutoffTm> 5-6
<Day> 2-7, 2-8, 2-9
<DayFax> 3-5
<DayPhone> 3-5
<DaysCall> 3-15
<DaysToEPost> 3-7, 7-4
<DaysToPay> 7-4, 7-10, 7-57
<DaysWith> 6-50, 7-57
<DebBkOrdRec> **6-12**
<DelinqAging> 6-19
<DeliveryMethod>6-12, 6-13, 6-22, 6-23, 6-48, 6-49, 7-24, 7-33, 7-35, 7-58
<DepAcctId>3-7, **3-8**, 3-12, 3-22, 5-9, 5-10, 6-1, 6-11, 6-13, 6-14, 6-15, 6-16, 6-17, 6-19, 6-20, 6-22, 6-23, 6-25, 6-26, 6-27, 6-28, 6-29, 6-30, 6-31, 6-32, 6-48, 6-49
<DepAcctIdFrom>2-14, **3-8**, 6-10, 6-26, 6-36, 6-37, 6-40, 6-44, 6-46, 6-47, 7-3, 7-11, 7-13, 7-16, 7-32, 7-34, 7-53, 7-54
<DepAcctIdTo> **3-8**, 6-7, 6-11, 6-36, 6-37, 7-12
<DepAcctRec> **6-4**, 6-16
<DepAcctStmtInqRq> 6-2, 6-16
<DepAcctStmtInqRs> 6-2, **6-16**
<DepAcctStmtRec> 6-17
<DepAcctTrnInqRq> 6-2, **6-19**
<DepAcctTrnInqRs> 6-2, **6-20**
<DepAcctTrnRec> **6-7**, 6-17, 6-21
<DepBkOrdAddRq> **6-48**
<DepBkOrdAddRs> **6-49**
<DepBkOrdId> 6-12
<DepBkOrdInfo> 6-12
<DepBkOrdStatus> 6-12, **6-13**
<DepBkOrdStatusCode> 6-13
<DepBkStyleId> 6-13, 6-48, 6-49
<DepBookOrderAddRq> 6-4
<DepBookOrderAddRs> 6-4
<DependentType>5-19, 5-26, 5-32, 6-43, 7-23, 7-52
<DepMatureDt> 6-4
<DepStmtTrn> 6-16
<Desc>3-6, 3-10, 3-13, 3-14, 3-15, 6-12, 6-24, 7-18, 8-3, 8-9, 8-10, 8-12
<DfltDaysToPay> 6-50, 7-4, 7-57
<DfltPmtInfo> 2-14, 7-11
<DfltXferDaysToPay> 7-4, 7-57
<DiscDt> 8-7
<DiscId> 5-8, 5-36, 5-37
<DiscInfo> 5-8
<Discount> 7-18
<DiscRec> 5-23, 5-24, 5-30, 5-31
<DiscReqd> 8-7
<DiscURL> 5-8
<DomXferFeeCurAmt> 6-50
<DueDt> 6-4, 6-11, 7-3, 7-13, 7-57

<DupChkOverride>6-33, 6-34, 6-41, 7-28, 7-29, 7-43, 7-44, 7-45, 7-50, 7-51
<EchoAggregate> 2-4
<EffDt>2-18, 3-11, 3-12, 3-13, 3-14, 3-16, 5-7, 5-8, 5-9, 5-10, 5-29, 6-11, 6-12, 6-13, 7-4, 7-14, 7-16, 7-17, 7-18, 7-43, 7-44, 7-45, 8-5, 8-7, 8-14
<Element1> 2-4
<Element2> 2-4
<EmailAddr> 3-5, 3-6
<EndDt> 3-15, 3-16, 6-17, 6-19, 8-16
<EnrollDesc> 5-38
<EnrollProf> 5-15, 5-38
<EU.Cur> 2-20, 4-4
<EveFax> 3-5
<EvePhone> 3-5
<ExpDt>3-10, 3-13, 3-14, 4-4, 5-16, 5-18, 6-5, 6-6, 6-9, 8-13
<ExtAcctBal> **3-13**, 6-14, 6-15, 6-16, 6-18
<ExtBalType> 3-13
<Fax> 3-6
<FeeCurAmt> 6-9, 6-50
<FICreditTrcNum> 7-13, 7-31, 7-32, 7-34
<FIDebitTrcNum> 7-12, 7-31, 7-32, 7-34
<FinalCurAmt> 3-17, 5-4
<FinalDueDt> 3-16, 3-17
<FinalPrcDt> 3-16, 3-17
<FirstName> 3-4
<ForExCommit> 5-3
<ForExInfo> 6-11
<ForExRateId> 6-26
<ForExRateInfo> **6-9**, 6-25, 6-26
<ForExRateInqRq> 6-3, **6-26**
<ForExRateInqRs> 6-3, **6-26**
<ForExRateRec> 6-26
<ForExRateType> 6-26
<Fraction> 2-8, 2-9
<Freq> 3-16, 6-5, 6-13, 7-58
<FSPayee> 7-7, 7-11
<FullName> 3-5
<GenSessKey> 4-3, 4-4
<Heading> 8-8
<HighCurAmt> 3-16, 6-24
<HistRetentionDays> 7-12, 7-57, 8-6
<HoldCoId> 5-5
<HoldCount> 6-4
<HolDt> 5-12
<HolInfo> 5-12
<HolInqRq> 5-1, **5-12**
<HolInqRs> 5-1, **5-12**, 7-4
<Hour> 2-8, 2-9
<ImageURL> 8-2, 8-13
<ImmediatePmt> 7-13
<ImmediateXfer> 5-4, 6-11
<IncBal> 5-32, 6-15
<IncBillerContact> 8-15, 8-16
<IncCounts> 8-17, 8-18
<IncDesc> 5-37
<IncDetail> 6-16, 6-17, 6-18, 6-19
<IncDisc> 5-37
<IncExtBal> 6-14, 6-15
<IncHistory> 7-33, 7-35
<IncImages> 8-15, 8-16
<IncSummary> 8-17, 8-18
<IncToken>5-19, 5-20, 5-32, 5-33, 6-29, 6-30, 6-37, 6-38, 6-44, 6-45, 7-24, 7-25, 7-33, 7-35, 7-53, 7-54
<IndustId>**3-20**, 6-6, 7-10, 7-19, 7-20, 8-6, 8-15, 8-16
<IndustNum> 3-20
<InitialCurAmt> 3-17, 5-4
<InsPaidCurAmt> 6-6
<InsPaidDt> 6-6
<IntAPY> 6-24
<InterXferProf> 6-50
<IntlXferFeeCurAmt> 6-50
<IntRateInfo> 6-24
<IntRateInqRq> 6-2, **6-23**
<IntRateInqRs> 6-2, **6-24**
<InvoiceAdj> 7-18
<InvoiceAdjNum> 7-18
<InvoiceInfo> 5-4, 7-6, 7-17, 7-29
<InvoiceLineItem> 7-18
<InvoiceLineItemNum> 7-18
<InvoiceNum> 7-18
<InvoicePremium> 7-18
<InvoiceVouchNum> 7-18
<Language> 4-5, 5-6
<LastDepCurAmt> 6-4
<LastDepDt> 6-4
<LastName> 3-4
<LastNRec> 7-33, 7-35
<LastPmtCurAmt> 6-5
<LastPmtDt> 6-5, 6-6
<LastStmtDt> 6-16
<LastTrnDt> 6-15
<LastYrInsPaidCurAmt> 6-6
<LastYrInsPaidDt> 6-6
<LastYrTaxPaidCurAmt> 6-6
<LastYrTaxPaidDt> 6-6
<LateFeeCurAmt> 7-18
<LegalName> 3-5, 3-6, 5-5
<LineItem> 5-4
<LoanAcctId>3-7, **3-10**, 3-12, 5-10, 6-1, 6-12, 6-14, 6-15, 6-22, 6-23, 6-25, 6-27, 6-28, 6-29, 6-30, 6-31, 6-32
<LoanAcctIdFrom>**3-10**, 6-10, 6-36, 6-37, 6-40, 6-44, 6-47
<LoanAcctIdTo> **3-10**, 6-7, 6-11, 6-36, 6-37
<LoanAcctRec> **6-5**, 6-16
<LoanInfo> 6-5
<LoanInfoCommon> **6-5**, 6-6
<LOCAcctRec> **6-5**, 6-16

<LocInfo> 6-5
<LocLimit> 6-5
<Logo> 8-6, 8-15
<LogoURL> 8-6
<LongText> 5-8
<LowCurAmt> 3-16, 6-24
<MagData1> 3-10
<MagData2> 3-10
<MagData3> 3-10
<MatchedRec> 3-19, 3-20
<MatDt> 6-5
<Max> 5-38
<MaxCurAmt> 3-8
<MaxRec> 3-18, 3-19, 3-20
<Memo>3-21, 3-22, 5-14, 5-38, 6-6, 6-8, 7-6, 7-11, 7-15, 7-17, 7-32, 7-34, 7-53, 7-54, 8-2, 8-3, 8-7, 8-9, 8-10, 8-11, 8-12, 8-14
<Method>5-26, 5-27, 5-33, 5-34, 6-38, 6-39, 6-45, 6-46, 7-25, 7-26, 7-35, 7-36, 7-41, 7-47, 7-48, 7-55
<MiddleName> 3-5
<Min> 5-38
<MinAmtDue> 6-18
<MinCurAmt> 3-8
<MinPmtCurAmt> 6-5
<Minute> 2-8, 2-9
<MktgInfo> 6-14, 6-17, 6-19, 6-21, 6-22
<ModPending> 6-13, 6-42, 7-21, 7-51, 7-58
<ModPendingType> 6-13, 7-21, 7-51, 7-58
<Month> 2-7, 2-8, 2-9
<MortAcctRec> **6-6**, 6-16
<MsgRecDt>5-21, 5-22, 5-27, 5-28, 5-34, 5-35, 6-31, 6-33, 6-39, 6-40, 6-46, 6-47, 7-26, 7-27, 7-36, 7-37, 7-41, 7-42, 7-48, 7-49, 7-55, 7-56
<MsgSupt> 2-22, **5-3**, 5-38, 6-49, 7-57, 8-20
<MustViewDtlToPay> 8-11, 8-13
<Name>3-6, 3-9, 3-10, 3-20, 5-12, 6-6, 6-7, 6-9, 7-10, 7-11, 7-16, 7-19, 7-20, 7-24, 7-32, 7-34, 8-6, 8-15, 8-16
<NameAddrType> 3-5
<NameSuffix> 3-5
<NewToken>2-17, 3-3, 5-22, 5-28, 5-35, 6-32, 6-40, 6-47, 7-27, 7-37, 7-42, 7-49, 7-56
<NewUpDt> 5-32, 8-16
<NextDt> 6-17, 6-18
<NextPmtCurAmt> 6-5
<Nickname>3-5, 3-17, 5-5, 5-10, 7-2, 7-11, 7-24, 7-25, 7-32, 7-34
<NotifyReqd> 8-11, 8-13, 8-18
<NotifyWilling> 8-11, 8-13, 8-17, 8-18
<NumInsts> 3-16, 3-17
<OpenDt> 8-10, 8-12
<OptionalElement> 2-3
<OptionalRepeating> 2-3
<OptSupt>3-18, 3-19, **5-3**, 5-5, 5-38, 6-10, 6-13, 6-49, 7-57, 8-7, 8-20
<Org> **3-3**, 3-11, 3-20, 6-25
<OrgContact>**3-6**, 3-11, 3-22, 5-5, 7-12, 8-5, 8-7
<OrigCurAmt> 6-6
<OrigDt> 6-5, 6-6, 6-9
<OriginatorName> 6-8
<OrigPmtDueDt> 7-12
<OrigPmtPrcDt> 7-12
<PaidCurAmt> 7-18
<PastDuePmtCount> 6-5
<Payable> 8-2
<PayeeAddRq> 7-2
<PayeeInfo> 7-27
<PayeeType> 7-22, 7-24
<PayInfo> 8-10, 8-12
<PaySummAmt> 8-9, 8-11
<PaySupt> 3-8, 5-13
<PaySvc> **7-1**
<PaySvcProfInfo> **7-57**
<Phone> 3-6, 3-10, 6-12, 8-15, 8-16
<PINBlock> 4-3
<PmtAddRq>2-13, 2-20, 7-2, 7-7, 7-10, **7-27**, 7-28, 7-35
<PmtAddRs> 3-2, 7-7, **7-28**, 7-36, 7-37
<PmtAudRq> 7-7, **7-35**
<PmtAudRs> 7-7, 7-28, 7-29, **7-36**
<PmtAuthAddRq> 7-8, **7-38**
<PmtAuthAddRs> 7-8, **7-38**, 7-41, 7-42
<PmtAuthAudRq> 7-8, **7-40**
<PmtAuthAudRs> 7-8, **7-41**
<PmtAuthCanRq> 7-8, **7-39**
<PmtAuthCanRs> 7-8, **7-39**, 7-41, 7-43
<PmtAuthCount> 7-15
<PmtAuthId> 7-14, 7-15, 7-38, 7-39, 7-40, 7-41
<PmtAuthInfo> 7-15, 7-38, 7-39
<PmtAuthInqRq> 7-8, **7-40**
<PmtAuthInqRs> 7-8, **7-40**
<PmtAuthModRq> 7-8, **7-38**
<PmtAuthModRs> 7-8, **7-39**, 7-41, 7-42
<PmtAuthMsgRec> 7-41, 7-42
<PmtAuthRec> **7-15**, 7-38, 7-39, 7-40
<PmtAuthStatus> 7-15, **7-16**
<PmtAuthStatusCode> 7-16
<PmtAuthSyncRq> 7-8, **7-42**
<PmtAuthSyncRs> 7-8, **7-42**
<PmtCanRq> 2-14, 7-7, 7-14, **7-30**, 7-31, 7-46
<PmtCanRs> 7-7, **7-31**, 7-36, 7-38
<PmtDueDt> 6-5, 6-18, 7-57, 8-2, 8-10, 8-12
<PmtId>3-3, 6-7, 7-12, 7-16, 7-28, 7-29, 7-30, 7-31, 7-32, 7-33, 7-35, 7-36, 7-43, 7-44, 7-45, 8-14
<PmtInfo>7-3, 7-11, 7-12, **7-13**, 7-15, 7-28, 7-29, 7-50, 7-51, 7-57, 8-9, 8-11
<PmtInqRq> 2-16, 7-7, **7-31**, 7-33
<PmtInqRs> 7-7, **7-33**
<PmtInst> 3-7, **3-21**, 7-39, 7-40, 8-10, 8-13
<PmtInstType> 3-22
<PmtMethod> 7-15, 7-32, 7-34

<PmtModel> 7-57
<PmtModRq> 2-13, 7-7, 7-15, **7-28**, 7-29
<PmtModRs> 3-3, 7-7, **7-29**, 7-36, 7-37
<PmtMsgRec> 7-36, 7-37, 7-48, 7-49
<PmtRec> **7-12**, 7-28, 7-29, 7-31, 7-35
<PmtRemitAddr> 7-12
<PmtRemitName> 7-12
<PmtStatus> 7-6, 7-12, **7-13**, 7-30, 7-36, 7-37
<PmtStatusCode> 7-6, 7-14, 7-15, 7-32, 7-34
<PmtStatusModRq> 7-7, **7-30**
<PmtStatusModRs> 7-7, **7-30**, 7-36, 7-37
<PmtSummAmt> 7-6, 7-17
<PmtSyncRq> 2-18, 7-8, **7-37**
<PmtSyncRs> 7-8, 7-28, 7-29, **7-37**
<PmtType> 7-32, 7-33
<PostAddr>2-13, 3-5, 3-6, **3-7**, 3-9, 3-10, 5-7, 5-39, 6-12, 7-10, 7-11, 8-15, 8-16
<PostAddrReqd> 5-39
<PostalCode>3-7, 3-9, 3-10, 7-19, 7-20, 8-8, 8-15, 8-16
<PostedDt> 6-6, 6-8
<PrcDaysOff> 5-6, 7-4
<PrcDt> 7-3, 7-13, 7-57
<PrcDtAdj> 5-6
<PrcSched> **5-6**, 6-13, 6-50, 7-57, 8-20
<PrefetchURL> 8-13
<PrefTimeEnd> 3-5
<PrefTimeStart> 3-5
<PrenoteReqd> 3-7
<PresAcctId>2-13, 3-1, 3-11, 5-10, 8-2, 8-6, 8-9, 8-11
<PresAcctInfo> 3-11, **3-12**, 5-10
<PresAcctRec> **3-11**
<PresAcctStatus> 3-11, **3-12**
<PresAcctStatusCode> 3-12
<PresSvc> **8-1**
<PresSvcProfInfo> **8-20**
<PrevYrCurAmt> 6-25
<PrimaryAcct> 3-11
<ProfileAggregate> 2-4
<Prompt> 3-21
<ProxyClient> 4-4
<Pswd> 4-1
<PswdDelivery> 5-16
<PurposeDesc> 6-5
<Rate> 6-24, 6-25, 7-18
<RecCtrlIn>**3-18**, 3-19, 5-21, 5-22, 5-26, 5-27, 5-28, 5-33, 5-34, 5-35, 6-19, 6-20, 6-21, 6-22, 6-23, 6-24, 6-29, 6-30, 6-31, 6-32, 6-36, 6-37, 6-38, 6-39, 6-40, 6-44, 6-45, 6-46, 6-47, 7-19, 7-24, 7-25, 7-26, 7-27, 7-32, 7-33, 7-35, 7-36, 7-37, 7-41, 7-42, 7-47, 7-49, 7-53, 7-54, 7-55, 7-56, 8-15, 8-17
<RecCtrlOut>3-19, 5-4, 5-21, 5-22, 5-27, 5-28, 5-34, 5-35, 6-20, 6-22, 6-24, 6-30, 6-31, 6-32, 6-37, 6-39, 6-40, 6-44, 6-46, 6-47, 7-19, 7-24, 7-25, 7-27, 7-33, 7-36, 7-37, 7-41, 7-42, 7-48, 7-49, 7-53, 7-55, 7-56, 8-15, 8-17
<RecInterModRq> 6-13
<RecModelInfo>**3-16**, 5-4, 6-10, 6-41, 6-42, 7-15, 7-50, 7-51
<RecPmtAddRq> 7-2, 7-9, 7-10, 7-15, **7-50**
<RecPmtAddRs> 7-9, **7-50**, 7-55, 7-56
<RecPmtAudRq> 7-9, **7-54**
<RecPmtAudRs> 7-9, **7-55**
<RecPmtCanRq> 7-9, **7-52**
<RecPmtCanRs> 7-9, **7-52**, 7-55, 7-57
<RecPmtDelRq> 7-14
<RecPmtId>7-12, 7-15, 7-32, 7-34, 7-51, 7-52, 7-53, 7-55
<RecPmtInqRq> 7-9, **7-52**
<RecPmtInqRs> 7-9, **7-53**
<RecPmtMod> 7-15
<RecPmtModRq> 7-9, 7-15, **7-51**, 7-58
<RecPmtModRs> 7-9, 7-15, **7-51**, 7-55, 7-56
<RecPmtMsgRec> 7-55, 7-56
<RecPmtProf> 7-58
<RecPmtRec> **7-15**, 7-50, 7-51, 7-52, 7-54
<RecPmtSyncRq> 7-9, **7-56**
<RecPmtSyncRs> 7-9, **7-56**
<RecSeriesEnd> 6-10, 7-15
<RecXferAddRq> 6-3, 6-10, **6-41**
<RecXferAddRs> 6-3, **6-41**, 6-46, 6-47
<RecXferAudRq> 6-4, **6-45**
<RecXferAudRs> 6-4, **6-45**
<RecXferCanRq> 6-4, **6-43**
<RecXferCanRs> 6-4, **6-43**, 6-46, 6-47
<RecXferId>6-9, 6-10, 6-36, 6-37, 6-38, 6-39, 6-42, 6-43, 6-44, 6-45, 6-46
<RecXferInqRq> 6-4, **6-43**
<RecXferInqRs> 6-4, **6-44**
<RecXferMod> 6-10
<RecXferModRq> 6-4, 6-10, 6-13, **6-42**
<RecXferModRs> 3-17, 6-4, **6-42**, 6-46, 6-47
<RecXferMsgRec> 6-46, 6-47
<RecXferProf> 6-13
<RecXferRec> **6-10**, 6-41, 6-42, 6-43, 6-45
<RecXferSyncRq> 6-4, **6-46**
<RecXferSyncRs> 6-4, **6-47**
<RegPmtCurAmt> 6-5
<RemainingInsts> 6-10, 7-15
<RemitAddr> 3-6
<RemitAddRq> 7-4, 7-8, **7-43**
<RemitAddRs> **7-43**, 7-48, 7-49
<RemitAudRq> 7-8, **7-47**
<RemitAudRs> 7-8, 7-43, 7-44, **7-48**
<RemitDelRq> 7-8, **7-46**
<RemitDelRs> 7-8, **7-46**, 7-48, 7-49
<RemitId>7-15, 7-16, 7-44, 7-45, 7-46, 7-47, 7-48
<RemitInfo>7-6, 7-13, 7-16, 7-27, 7-43, 7-44, 7-45
<RemitInqRq> 7-8, **7-46**
<RemitInqRs> 7-8, **7-47**

<RemitModRq> 7-8, **7-44**
<RemitModRs> 7-8, **7-44**, 7-48, 7-49
<RemitMsgRec> 7-48, 7-49
<RemitName> 3-6
<RemitRec> **7-16**, 7-44, 7-45, 7-47
<RemitStatus> 7-16, **7-17**, 7-45, 7-46, 7-47
<RemitStatusCode> 7-17
<RemitStatusModRq> 7-8, **7-45**
<RemitStatusModRs> 7-8, **7-45**, 7-48, 7-49
<RemitSyncRq> 7-8, **7-48**
<RemitSyncRs> 7-8, **7-49**
<RemittAddRs> 7-8
<RequiredElement> 2-3
<ResolvePendingDt> 5-7, 5-9, 5-11
<RqUID>2-3, 2-19, 2-22, **3-2**, 3-17, 4-2, 4-5, 5-15, 5-16, 5-17, 5-18, 5-19, 5-20, 5-21, 5-22, 5-23, 5-24, 5-25, 5-26, 5-27, 5-28, 5-29, 5-30, 5-31, 5-32, 5-33, 5-34, 5-35, 5-36, 5-37, 6-14, 6-15, 6-16, 6-17, 6-18, 6-19, 6-20, 6-21, 6-22, 6-23, 6-24, 6-25, 6-26, 6-27, 6-28, 6-29, 6-30, 6-31, 6-32, 6-33, 6-34, 6-35, 6-36, 6-37, 6-38, 6-39, 6-40, 6-41, 6-42, 6-43, 6-44, 6-45, 6-46, 6-47, 6-48, 6-49, 7-19, 7-20, 7-21, 7-22, 7-23, 7-24, 7-25, 7-26, 7-27, 7-28, 7-29, 7-30, 7-31, 7-33, 7-35, 7-36, 7-37, 7-38, 7-39, 7-40, 7-41, 7-42, 7-43, 7-44, 7-45, 7-46, 7-47, 7-48, 7-49, 7-50, 7-51, 7-52, 7-53, 7-54, 7-55, 7-56, 8-15, 8-16, 8-17, 8-19
<SalesSlipRefNum> 6-8
<Second> 2-8, 2-9
<Secret> 3-1, 3-21
<SecretFormat> 3-21
<SecretId> 3-21
<SecretList> 3-21, 5-15, 5-16, 5-29, 5-30
<SecretMask> **3-1**, 3-21
<SecretOptional> 3-21
<SecretPrompt>3-1, **3-20**, 3-21, 5-15, 5-29, 5-30, 5-38, 8-6
<SelRangeCurAmt>**3-16**, 6-19, 6-20, 6-21, 6-22, 6-36, 6-38, 6-44, 7-33, 7-35, 7-53, 7-54
<SelRangeDt>**3-15**, 5-21, 5-26, 5-27, 5-33, 5-34, 6-16, 6-17, 6-18, 6-19, 6-20, 6-21, 6-22, 6-29, 6-30, 6-31, 6-38, 6-39, 6-45, 6-46, 7-25, 7-35, 7-36, 7-40, 7-41, 7-47, 7-48, 7-54, 7-55, 8-17
<SelRangeDueDt> **3-15**, 6-36, 6-38, 7-32, 7-34
<SelRangePrcDt> **3-15**, 7-33, 7-34
<SentRec> 3-19, 3-20
<ServerDt> 4-4
<SessKey> **4-2**, 4-3, 4-4
<SettlementId> 3-22
<SettlementInfo> 3-22
<SettlementMethod> 3-22
<Severity>2-3, 2-18, 3-17, 3-18, 5-26, 5-27, 5-33, 5-34, 7-47
<ShipTo> 6-12
<ShortDesc> 8-3, 8-9, 8-10, 8-11, 8-12
<SignoffRq> 4-5
<SignoffRs> **4-5**
<SignonCert> 4-3
<SignonInfo> 4-1, 5-17, 5-38
<SignonMagPIN> 4-3
<SignonPswd> 4-2
<SignonRole> 4-3, 4-4
<SignonRq> 2-21, 3-4, 4-2, 4-4, 4-5
<SignonRs> 4-2, **4-4**
<SignonTransport> 4-3
<SkipNextN> 3-17, 5-4
<SpaceAlwd> 5-38
<SPName>2-19, **3-3**, 3-4, 5-5, 5-9, 5-10, 5-12, 5-21, 5-26, 5-27, 7-1, 7-9, 8-5, 8-6, 8-7, 8-14, 8-17
<SPRefId>**3-2**, 3-3, 5-16, 5-17, 5-18, 5-19, 5-24, 5-25, 5-26, 5-30, 5-31, 5-32, 5-36, 6-7, 6-9, 6-17, 6-18, 6-20, 6-21, 6-22, 6-23, 6-25, 6-28, 6-29, 6-34, 6-35, 6-36, 6-37, 6-41, 6-42, 6-43, 6-44, 6-45, 6-48, 6-49, 7-13, 7-21, 7-22, 7-23, 7-28, 7-29, 7-30, 7-31, 7-33, 7-35, 7-44, 7-45, 7-46, 7-50, 7-51, 7-52, 7-53, 7-54, 8-20
<SPRefIdCorrect> 6-7
<SrvrStatusCode> 3-18
<StartDt> 3-15, 3-16, 6-17, 6-19, 8-16
<StartTS> 3-16
<StateProv> 3-7, 3-9
<Status>2-3, 2-16, 2-17, 2-18, 2-19, 2-22, **3-17**, 4-4, 4-5, 5-12, 5-16, 5-17, 5-18, 5-19, 5-20, 5-21, 5-22, 5-23, 5-24, 5-25, 5-26, 5-27, 5-28, 5-29, 5-30, 5-31, 5-32, 5-33, 5-34, 5-35, 5-36, 5-37, 6-14, 6-15, 6-17, 6-18, 6-20, 6-21, 6-23, 6-24, 6-25, 6-26, 6-27, 6-28, 6-30, 6-31, 6-32, 6-33, 6-34, 6-35, 6-37, 6-38, 6-40, 6-41, 6-42, 6-43, 6-44, 6-45, 6-47, 6-48, 6-49, 7-19, 7-20, 7-21, 7-22, 7-23, 7-24, 7-25, 7-27, 7-28, 7-29, 7-30, 7-31, 7-33, 7-36, 7-37, 7-38, 7-39, 7-40, 7-41, 7-42, 7-43, 7-44, 7-45, 7-46, 7-47, 7-48, 7-49, 7-50, 7-51, 7-52, 7-53, 7-55, 7-56, 8-8, 8-15, 8-17, 8-19
<StatusCode> 3-18
<StatusDesc>3-11, 3-12, 3-17, 3-18, 5-7, 5-8, 5-9, 5-10, 5-26, 5-27, 5-33, 5-34, 6-8, 8-2
<StatusModBy>3-11, 3-12, 5-7, 5-8, 5-9, 5-10, 6-11, 6-12, 6-13, 7-16, 8-7, 8-14
<StdPayeeId>3-11, 3-12, 7-1, 7-3, 7-7, 7-9, 7-11, 7-19, 7-28, 7-32, 7-34, 8-6
<StdPayeeInfo> 7-9, **7-10**
<StdPayeeInqRq> 7-3, 7-7, **7-19**
<StdPayeeInqRs> 7-7, **7-19**
<StdPayeeRec> 7-4, **7-9**, 7-20
<StmtImage> 8-2, 8-11, 8-13
<StmtSummAmt> 6-17, 6-18
<StmtSummType> 6-17, 6-19
<StopChkAddRq> 6-3, 6-27
<StopChkAddRs> 6-3, **6-27**, 6-31, 6-33
<StopChkAudRq> 6-3, **6-30**

<StopChkAudRs> 6-3, **6-31**
<StopChkCanRq> 6-3, **6-28**
<StopChkCanRs> 6-3, **6-28**, 6-31, 6-33
<StopChkInfo>6-8, **6-9**, 6-23, 6-27, 6-28, 6-29, 6-30
<StopChkInqRq> 6-3, **6-29**
<StopChkInqRs> 6-3, **6-29**
<StopChkMsgRec> 6-31, 6-32, 6-33
<StopChkProf> 6-49
<StopChkRec> **6-8**, 6-28, 6-29, 6-30
<StopChkStatusCode> 6-8
<StopChkSyncRq> 6-3, **6-32**
<StopChkSyncRs> 6-3, **6-32**
<StopCount> 6-4
<SubAmt> 8-3
<SvcAcctAddRq>5-2, **5-28**, 5-29, 7-1, 8-2, 8-8, 8-14
<SvcAcctAddRs> 5-2, 5-29, 5-34, 5-35
<SvcAcctAudRq> 5-2, 5-29, **5-33**
<SvcAcctAudRs> 5-2, **5-33**
<SvcAcctDelRq> 5-2, **5-31**
<SvcAcctDelRs> 5-2, **5-31**, 5-34, 5-35
<SvcAcctId> 5-9, **5-10**, 5-29, 5-30, 5-31
<SvcAcctInfo> 5-9, **5-10**, 5-29, 5-30, 5-31
<SvcAcctInqRq> 5-2, **5-32**, 8-2
<SvcAcctInqRs> 5-2, **5-32**, 8-2
<SvcAcctModRq> 5-2, **5-30**
<SvcAcctModRs> 5-2, **5-30**, 5-34, 5-35
<SvcAcctMsgRec> 5-34, 5-35
<SvcAcctRec> **5-9**, 5-30, 5-31
<SvcAcctStatus> 5-9, **5-10**, 5-33, 5-34, 8-2
<SvcAcctStatusCode> 5-10, 5-11, 5-23, 8-2
<SvcAcctSyncRq> 5-2, 5-29, **5-34**
<SvcAcctSyncRs> 5-2, **5-35**
<SvcCore>2-17, **5-5**, 5-6, 5-12, 5-37, 6-49, 7-57, 8-20
<SvcName>5-9, 5-10, 5-12, 5-23, 5-26, 5-27, 5-30, 5-32, 5-36
<SvcProfInqRq> 5-1, **5-11**, 5-37, 6-49
<SvcProfInqRs> 5-1, 5-6, **5-12**, 7-57, 8-20
<SvcStatus> 5-26, 5-27
<SvcUpDt> 5-33
<TagName> 2-5
<TaxId> 5-7, 6-25
<TaxInqRq> 3-3
<TaxPaidCurAmt> 6-6
<TaxPaidDt> 6-6
<TaxStatus> 5-13
<TaxType> 6-25
<TaxYear> 6-25
<Term> 3-11, **3-15**, 6-5, 6-24
<TermUnits> 3-15
<TitlePrefix> 3-5
<Token>2-17, 2-18, **3-3**, 3-4, 5-19, 5-20, 5-22, 5-28, 5-32, 5-33, 5-34, 5-35, 6-29, 6-30, 6-32, 6-37, 6-38, 6-40, 6-44, 6-45, 6-46, 7-24, 7-25, 7-26, 7-27, 7-33, 7-35, 7-37, 7-42, 7-49, 7-53, 7-54, 7-56
<TotalCurAmt> 7-18
<TrnSrc> 6-6, 6-20, 6-21, 6-22
<TrnType> 6-6, 6-20, 6-21, 6-22
<UpdatePayee> 7-29
<UpDt>5-7, 5-11, 5-12, 5-19, 5-20, 5-32, 5-33, 6-15, 8-15, 8-16
<URL> 3-6, 5-6, 6-17, 6-19, 8-11, 8-13, 8-14
<USA.ACHProf> 6-50
<USA.ACHTrnInfo> 6-8
<USA.RTN> 5-5, 6-8
<UTCOffset> 2-8, 2-9
<Version> 2-17, 3-20, 5-5
<WebEnrollURL> 5-38
<XferAddRq> 6-3, **6-33**, 6-38
<XferAddRs> 6-3, **6-33**, 6-39, 6-41
<XferAudRq> 6-3, **6-38**
<XferAudRs> 6-3, **6-38**
<XferCanRq> 6-3, **6-34**
<XferCanRs> 6-3, **6-35**, 6-39, 6-41
<XferFromSupt> 3-8, 5-13
<XferId>3-3, 6-7, 6-9, 6-34, 6-35, 6-36, 6-37, 6-38, 6-39
<XferInfo> 6-10, 6-33, 6-34, 6-41, 6-42
<XferInqRq> 6-3, **6-36**
<XferInqRs> 6-3, **6-37**
<XferModRq> 6-3, 6-10, **6-34**
<XferModRs> 3-17, 6-3, **6-34**, 6-39, 6-41
<XferMsgRec> 6-39, 6-40
<XferPayee> 7-7, 7-11, **7-12**
<XferProf> 6-10, **6-13**, 6-50
<XferRec> **6-9**, 6-33, 6-34, 6-35, 6-38
<XferStatus> 6-10, **6-11**, 6-35, 6-36
<XferStatusCode> 6-11, 6-36, 6-38
<XferStatusModRq> **6-35**
<XferStatusModRs> **6-35**, 6-39, 6-41
<XferSyncRq> 6-3, **6-39**
<XferSyncRs> 6-3, **6-40**
<XferToSupt> 3-8, 5-13
<Year> 2-7, 2-8, 2-9