

DataConnect V4 Ultra

User Guide

©2025 Morningstar. All Rights Reserved.

DataConnect Product Version: V4.0

Document Version: 71

Document Issue Date: June 27, 2025

Technical Support: (866) 856-4951

Web: byallaccounts.morningstar.com

Table of Contents

Using the Manual	1
Audience	1
Format.....	1
DataConnect Introduction.....	2
Product Overview	2
Product Features	2
Product Description	4
Terminology	5
Related Documents	5
DataConnect Data Specification.....	6
Data Model Overview.....	8
General Object Information.....	11
Data Types.....	11
Object Definitions.....	13
Security.....	13
FI (Financial Institution or Financial Service)	27
Person.....	31
Portfolio.....	32
Login	32
Financial Profile.....	32
Profile Access.....	33
Account Credential	34
Security Question and Answer (SQA).....	37
Account.....	37
Holding.....	46
Transaction.....	50
Investment Option.....	59
Holding Lot	60
DataConnect Operations	63

Table of Contents

Overview.....	63
DataConnect URL.....	63
Protocol	63
Versioning	63
Compression.....	63
General XML Document Information.....	64
Overall Document Structure	64
Input Request.....	64
Output Response	65
Multipart Requests.....	65
Authentication	65
Authorization.....	66
Asynchronous Operations	66
In-Session Activation Codes	67
Sensitive Data	68
Passwords	68
Aggregates	69
Data Types.....	72
Operation Profile	72
Operation Summary	73
General Operations	76
Login.....	76
Purpose	76
Restrictions.....	76
Request: <LOGINRQ>	76
Response: <LOGINRS>	77
Errors	77
Sample XML	77
MultiPart Request.....	79

Table of Contents

Purpose	79
Restrictions.....	79
Response: <MULTIRQ_A>.....	79
Response: <MULTIRS_A>	79
Sample XML	80
General Error	84
Purpose	84
Restrictions.....	84
Response: <GENERALRS>	84
Errors	84
Sample XML	84
Asynchronous Helper Operations	85
Claim Data.....	85
Purpose	85
Restrictions.....	85
Request: <DATACLAIMRQ>	85
Response: <DATACLAIMRS>.....	85
Errors	86
Sample XML	86
Acknowledge Data Received	88
Purpose	88
Restrictions.....	88
Behavior.....	88
Request: <DATAACKRQ>	88
Response: <DATAACKRS>	88
Errors	88
Sample XML	89
Claim File.....	90
Purpose.....	90

Table of Contents

Restrictions.....	90
Behavior	90
Request: <FILECLAIMRQ>.....	91
Response: ZIP file containing CSV exports, or <FILECLAIMRS>.....	91
Errors	93
Sample XML	93
Retrieval Operations	95
Get Data (Synchronous).....	95
Purpose	95
Restrictions.....	95
Behavior	95
Request: <DATAGETRQ>	95
Response: <DATAGETRS>	104
Errors	106
Sample XML	106
Sample XML	109
Get Data (Asynchronous).....	112
Purpose	112
Restrictions.....	112
Request: <DATAGETRQ_A>.....	112
Response: <DATAGETRS_A>	112
Errors	113
Sample XML	113
Get Financial Service List.....	113
Purpose	113
Restrictions.....	113
Request: <FILISTGETRQ>	113
Response: <FILISTGETRS>	113
Errors	114

Table of Contents

Sample XML	114
Get Financial Service	116
Purpose	116
Restrictions.....	116
Request: <FIGETRQ>.....	116
Response: <FIGETRS>	116
Errors	117
Sample XML	117
Export Data (Asynchronous)	121
Purpose	121
Restrictions.....	121
Behavior.....	121
Request: <DATAEXPORTRQ_A>.....	121
Response: <DATAEXPORTRS_A>	125
Errors	125
Sample XML	126
Import Data (Asynchronous).....	129
Purpose	129
Restrictions.....	129
Behavior	129
Request: <DATAIMPORTRQ_A>.....	129
Response: <DATAIMPORTRS_A>	132
Errors	133
Sample XML	133
List Available Tax Lots.....	136
Purpose	136
Restrictions.....	136
Behavior	136
Request: <TAXLOTFILELISTRQ>.....	136

Table of Contents

Response: <TAXLOTFILELISTRS>	137
Errors	138
Sample XML	138
Modification Operations.....	140
Add User.....	140
Purpose	140
Restrictions.....	140
Behavior.....	140
User Notifications.....	141
Request: <USERADDRQ>.....	141
Response: <USERADDRS>.....	143
Errors	144
Sample XML	145
Unsubscribe User	153
Purpose	153
Restrictions.....	153
Behavior.....	153
User Notifications.....	153
Request: <USERUNSUBSCRIBERQ>	154
Response: <USERUNSUBSCRIBERS>.....	154
Errors	155
Sample XML	156
Modify User.....	157
Purpose	157
Restrictions.....	157
Behavior.....	157
Security.....	157
User Notifications.....	157
Request: <USERMODRQ>	158

Table of Contents

Response: <USERMODRS>.....	160
Errors	161
Sample XML	162
Modify Financial Profile	163
Purpose	163
Restrictions.....	163
Request: <PROFMODRQ>	163
Response: <PROFMODRS>	163
Errors	163
Sample XML	164
Add Portfolio.....	165
Purpose	165
Restrictions.....	165
Request: <PORTADDRQ>	165
Response: <PORTADDRS>	165
Errors	166
Sample XML	167
Modify Portfolio.....	167
Purpose	167
Restrictions.....	168
Request: <PORTMODRQ>	168
Response: <PORTMODRS>.....	169
Errors	169
Sample XML	169
Delete Portfolio	171
Purpose	171
Restrictions.....	171
Request: <PORTDELQR>.....	171
Response: <PORTDELR>	171

Table of Contents

Errors	171
Sample XML	172
Add Account Credential	173
Purpose	173
Restrictions.....	173
Request: <ACCTCREDADDRQ>	174
Response: <ACCTCREDADDRS>.....	177
Errors	177
Sample XML	177
Modify Account Credential	179
Purpose	179
Restrictions.....	179
Behavior	179
Request: <ACCTCREDMODRQ>	180
Response: <ACCTCREDMODRS>	183
Errors	184
Sample XML	185
Delete Account Credential.....	187
Purpose	187
Restrictions.....	187
Request: <ACCTCREDDELREQ>	187
Response: <ACCTCREDDELRS>	187
Errors	188
Sample XML	188
Add Account	190
Purpose	190
Restrictions.....	190
Request: <ACCTADDRQ>	191
Response: <ACCTADDRS>	196

Table of Contents

Errors	196
Sample XML	197
Modify Account	199
Purpose	199
Restrictions.....	199
Behavior	199
Request: <ACCTMODRQ>	200
Response: <ACCTMODRS>	204
Errors	204
Sample XML	205
Delete Account.....	206
Purpose	206
Restrictions.....	206
Request: <ACCTDELREQ>.....	206
Response: <ACCTDELRS>	206
Errors	206
Sample XML	207
Aggregation Operations	208
Test Account Credential	208
Purpose	208
Restrictions.....	209
Behavior	209
Request: <ACCTCREDTESTRQ_A>	209
Response: <ACCTCREDTESTRS_A>	210
Errors	212
Sample XML	213
Sample XML with START_OAUTH	216
Test Account.....	220
Purpose.....	220

Table of Contents

Restrictions.....	220
Behavior	220
Request: <ACCTESTRQ_A>	221
Response: <ACCTESTRS_A>	221
Errors	223
Sample XML	224
Update Account from FI.....	226
Purpose	226
Restrictions.....	226
Behavior.....	226
Request: <ACCTUPDRQ_A>	227
Response: <ACCTUPDRS_A>.....	228
Errors	229
Sample XML	229
Update Account Tax Lots.....	230
Purpose.....	230
Restrictions.....	230
Behavior	230
Request: <ACCTUPDTAXLOTRQ>	231
Response: <ACCTUPDTAXLOTRS>.....	232
Errors	232
Sample XML	233
Discover Accounts.....	234
Purpose	234
Restrictions.....	235
Behavior	235
Request: <ACCTDISCRQ_A>	236
Response: <ACCTDISCRS_A>	236
Errors	237

Table of Contents

Sample XML	238
In-Session Activation Codes (ISAC).....	242
Purpose	243
Request: <USERINPUTRESULTRQ>	243
Response: <USERINPUTRESULTRS>	244
Errors	244
USERINPUT Aggregate.....	245
Sample XML	247
Single Sign-on Operations.....	260
SSO - Session Authenticate Request.....	260
Purpose	260
Restrictions.....	260
Request: <SESSIONAUTHRQ>	261
Response: <SESSIONAUTHRS>.....	261
Errors	262
Sample XML	262
SSO - Session Expire Request	264
Purpose	264
Restrictions.....	264
Request: <SESSIONEXPIRERQ>	264
Response: <SESSIONEXPIRERS>.....	264
Errors	264
Sample XML	265
SAML SSO – SAML Session Authenticate Request.....	267
Purpose	267
Restrictions.....	267
Behavior	267
Request: <SAMLAUTHRQ>	268
Response: <SAMLAUTHRS>.....	268

Table of Contents

Errors	268
Sample XML	269
DataConnect Usage Considerations.....	271
DataConnect Access	271
Compression.....	271
Data Availability.....	271
Appendix A: DataConnect Access and Document Type Definitions	272
Appendix B: DataConnect Error Codes	273
Appendix C: Account Update Status Error Codes	280
Appendix D: Sample Call to the API	283
Appendix E: Sample Error Response Documents	286
Appendix F: Tax Lot Files	294
Tax Lot File Names.....	294
Open Lot Tax File	294
Closed Lot Tax File	297

USING THE MANUAL

Audience

This manual is targeted at the software professional responsible for implementing a DataConnect client. The guide assumes that the reader has basic programming experience with one or more programming languages and has some experience with XML, Document Type Definitions (DTD), https, and ZIP compression format.

DataConnect consists of the following products - DataConnect Lite and DataConnect Ultra. This document describes the capabilities of DataConnect Ultra (hereafter known as DataConnect). Please refer to the [*DataConnect Lite User Guide*](#) for details on DataConnect Lite.

Format

The manual is composed of the following sections:

- [Data Connect Introduction](#): Provides an overview of DataConnect, its features, and its operation. It also provides a list of terms with which users should be familiar and a reference to related documents.
- [Data Connect Data Specification](#): Defines the data available through DataConnect.
- [Data Connect Operations](#): Provides a detailed description of all DataConnect Ultra operations.
- [Data Connect Usage Considerations](#): Describes the access, compression, and data availability considerations with which users should be familiar when using DataConnect.
- [Appendices](#): Consists of a number of appendices that include definition and sample code used in the implementation of DataConnect.

DATACONNECT INTRODUCTION

Product Overview

DataConnect provides an Application Programming Interface (API) to the Morningstar® ByAllAccountsSM Aggregation Service. DataConnect consists of the following products:

- **DataConnect Lite:** Enables the retrieval of the financial information for one or more users.
- **DataConnect Ultra:** Provides the methods necessary for building a custom application that uses the service. Includes all capabilities available in DataConnect Lite.

Product Features

The table below summarizes the features currently available in the two DataConnect products.

Functionality	DataConnect Lite	DataConnect Ultra
Access Level		
Individual	√	√
Firm	√	√
Retrieve Basic Financial Information		
User	√	√
Portfolios	√	√
Accounts	√	√
Holdings	√	√
Transactions	√	√
Financial Services	√	√
Securities	√	√
Holding Lots	√	√
Retrieve Extended Financial Information		
Financial Services: Online Access Instructions		√
User: Personal Information		√
Accounts: Access Credentials		√

Operation Styles		
Synchronous	√	√
Asynchronous	√	√
Create and Maintain Information		
Users		√
Portfolios		√
Accounts		√
Additional Operations		
Test account		√
Test account credentials		√
On-demand Update Account (from Financial Service)		√
In-session Activation Codes		√
Aggregation with tax lots		√

Product Description

DataConnect supports communication over an Internet connection using the industry-standard https protocol and XML documents. Data returned is compressed in ZIP standard compression format. DataConnect does not provide any client-side SDK or other software for use in developing a DataConnect client. Programming is done directly to https in a programming language of your choice.

DataConnect can be used to construct a User Interface to the ByAllAccounts data service. It can also be used in conjunction with other products in the ByAllAccounts offerings such as AccountView, ByAllAccounts Connect components, or the Consumer User Interface 2 (CUI2) application. For example, using one of these products:

1. View and edit data for users created via DataConnect.
2. Enter or edit online access credentials for accounts owned by each user.

DataConnect enables you to enroll, maintain, and unsubscribe ByAllAccounts users. You can also create and maintain portfolios and online accounts for those users. Additional functions provide information about the Financial Services supported by ByAllAccounts and the types of information needed (online credentials) to gather data for an account at those Financial Services. You may use DataConnect operations to verify that online credentials for an account were entered correctly and to request that data be gathered for an online account (from its Financial Service) immediately.

DataConnect can return data for a single user, a select set of users, or for the entire set of users associated with a particular ByAllAccounts client. These two retrieval styles are characterized as on-demand and bulk, respectively. If you wish to retrieve data for a single user for presentation in another application, you use the on-demand style of access to retrieve data from the ByAllAccounts servers at any time of day. If you wish to store and serve ByAllAccounts data from your own servers, you use the bulk retrieval style, typically retrieving data from ByAllAccounts once a day.

DataConnect operations with potentially lengthy processing times are provided as asynchronous operations. Some operations are available in both synchronous and asynchronous form. The general style of the asynchronous operation is for the DataConnect server to provide a receipt to you for the requested operation and a data expiration time. Later, you submit a request with this receipt to the DataConnect server to request your data. Data is retained for you to retrieve up to the expiration time.

The DataConnect caller is authenticated through a specially assigned User ID and Password. By necessity, DataConnect must make some sensitive information available to you (needed to build a User Interface to these capabilities), so you must restrict access to this assigned User ID and Password to prevent unauthorized access to ByAllAccounts users' data.

Terminology

The following are some terms with which to be familiar when using DataConnect:

- **Administrator**
Person who has administrative access to ByAllAccounts
- **Advisor**
Person providing investment management and advice services to Investors
- **Assistant:**
Person who assists the advisor in investment management and client service
- **Consultant**
Person with whom an Investor shares their financial information, usually to obtain advice
- **DataConnect client**
Program written by ByAllAccounts customer that interacts with the DataConnect server
- **End User**
Person using a User Interface to the ByAllAccounts server
- **Firm**
The ByAllAccounts customer
- **Investor**
Person with investments; an account holder
- **Unassigned Investor**
An investor that is automatically created and used for accounts that are not assigned to a “real” investor
- **User**
An end user who is an Investor, Advisor, Assistant, or Consultant
- **ByAllAccounts**
The entire ByAllAccounts service, both front end and back end

Related Documents

The following related documents are available from ByAllAccounts:

- [DataConnect V4 Lite User Guide](#): Provides details on DataConnect Lite Version 4.

DATACONNECT DATA SPECIFICATION

This section defines the data available through DataConnect. The [Data Connect Operations section](#) refers to data defined in this section.

The ByAllAccounts product family provides for the creation and maintenance of many types of objects. However, not all ByAllAccounts Advisor/Investor information can be accessed through DataConnect. The objects available through DataConnect are:

- [Person:](#)

A Person known to the service. A Person has personal information, such as name and email address. The term User refers collectively to a Person, their Login, and their Financial Profile. DataConnect supports the following User types:

- Investor: One who invests
- Advisor: One who manages investments for an Investor
- Assistant: One who assists an advisor
- Consultant: One who advises on investments

The Consultant in DataConnect V4 is a personal friend, relative, or professional known to the Investor and registered in the service with the ability to view that Investor's financial information. Consultants that span Investors and who are known more broadly in the service are not yet supported.

- [Login:](#)

A set of credentials that authenticates a Person for ByAllAccounts access. A Login is required for a person to access the ByAllAccounts service directly. Users who do not access the service do not need a Login.

- [Financial Profile:](#)

A container that groups the investments (Portfolios, Accounts, Holdings, and Transactions) and settings for a single Investor (see Person). A Financial Profile may be accessed by several individuals, including the Investor (holder of the accounts) and the Advisor.

- [Profile Access:](#)

Persons who are allowed to access Financial Profiles.

- [Portfolio:](#)

A collection of investments, usually with an associated investment time horizon, risk tolerance, and target allocation. The Portfolio groups a set of Accounts together for investment planning and analysis. An Account can belong to only one Portfolio.

- [Account Credential:](#)

The credentials used to access one or more Accounts at a Financial Service (e.g., a Login and password to a Financial Institution's web site).

- [Security Question and Answer \(SQA\):](#)

The security questions and answers on a credential.

-
- **Account:**

A single Account held at a Financial Service. An Account may be online (where Account information is available online from the Financial Institution via a web site or data server) or off-line (created and manually maintained by a ByAllAccounts user). Online Accounts have an associated Account Credential object that defines the credentials used to access the Account. ByAllAccounts gathers data for online Accounts from the Financial Service on a nightly basis.
- **[Holding:](#)**

A position in a Security. A Holding is always contained in one Account. A Holding may be related to a marketable Security (see Security) known to ByAllAccounts or it may be a named Holding (i.e., one which has no related Security) identified only by name and not generally marketable.
- **[Transaction:](#)**

A record of activity in a Holding or Account (e.g., a buy or sell of a Security). A Transaction is always contained in one Account and may be related to a Holding.
- **[Security:](#)**

A financial instrument that can be identified by ByAllAccounts. The most common types of Securities are stock, bond, mutual fund, option, and cash. ByAllAccounts maintains a single Security master referred to by Holdings and Transactions.
- **[FI \(Financial Institution or Financial Service\):](#)**

A data access service offered by a Financial Institution through which an Investor can view their Account information. A large Financial Institution may provide several different Financial Services with each service providing information for Accounts in a particular segment of business (e.g., Trust Accounts, General Brokerage Accounts, 401(k) Accounts). ByAllAccounts maintains a single Financial Service master referred to by all online Accounts.

The objects and aspects of information not available through DataConnect are:

- **Alert:** Warning delivered to users on a variety of conditions from Security price movement to Portfolio exposure.
- **Historical Information:** Historical data (other than Transactions) for Portfolios, Accounts, and Holdings, including historical relationships, historical position valuations, and archived Portfolios, Accounts, and Holdings.

In addition, the following objects can be retrieved through DataConnect but cannot be inserted, modified, or deleted through DataConnect:

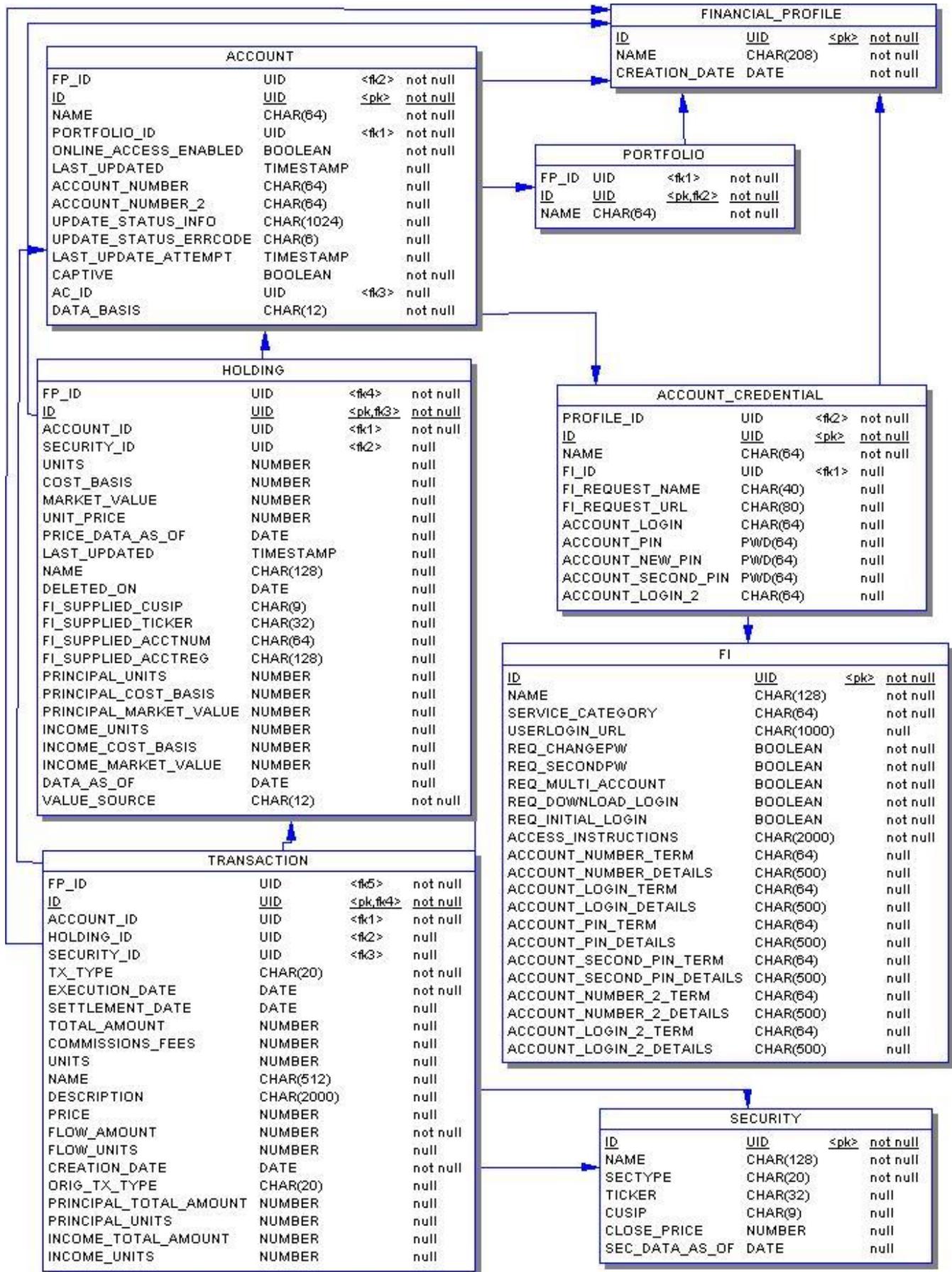
- **Holdings**
- **Transactions**

Data Model Overview

People known to the system are kept in the Person object. A Person has a role that identifies his primary use of the service: Investor, Advisor, Assistant, or Consultant. A Person optionally has one and only one Login that provides access to the service. Some persons may not have Logins, such as Investors who have financial data in the service but their Advisor manages that data. Investor and Advisor users can each have one Financial Profile that holds financial information for their personal accounts. Relationships between people and financial information are stored in the Profile Access object. The Financial Profile is a container of financial information to which different individuals are granted levels of access. An Investor has a relationship to only one Financial Profile - his own. An Advisor can have relationships with multiple Financial Profiles - one being his own and others belonging to Investors for whom he manages investments. An Assistant is assigned to an Advisor and has access to all of the Financial Profiles to which the Advisor has access. This access can be read-only or read-write. One additional type of person is the Consultant who has access to one and only one Financial Profile - that of the Investor to whom he provides advice, either directly or indirectly (through the Advisor).

The Financial Profile contains Portfolios, Accounts, Account Credentials, Holdings, and Transactions. Holdings and Transactions may relate to a well-known Security. Account Credentials may refer to a Financial Service supported by ByAllAccounts.

Figure 1 - Data Model Diagram on the following two pages depicts the underlying structure of the data available in DataConnect. Not all restrictions on objects are presented in this diagram. The model is presented in Entity-Relationship notation and includes primary key and foreign key designations to enable you to plan for storage and management of the data. Objects are shown as boxes while relationships are shown as arrows. A relationship arrow points to the parent of the relationship (e.g., Holding is the parent of the Holding-Transaction relationship shown in the Data Model Diagram). The Data Types used in this diagram and in the detailed object definitions that follow are defined in the [Data Types table](#).



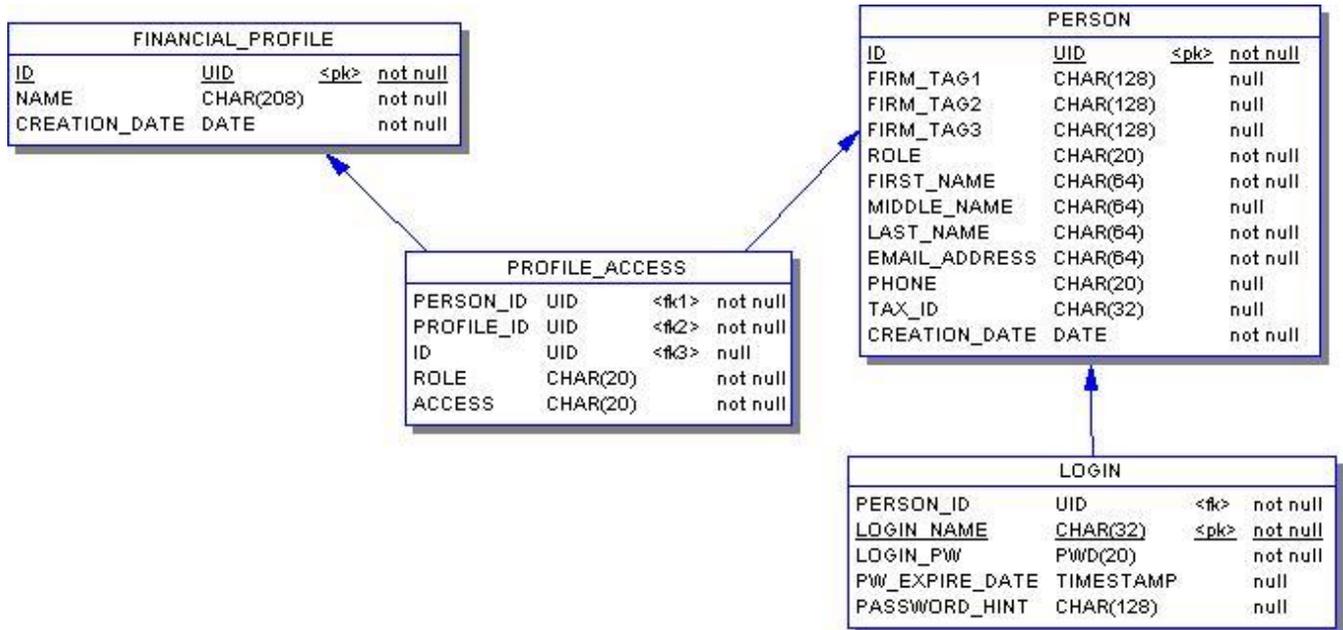


Figure 1 - Data Model Diagram

General Object Information

The following sections define the fields available for each of the data objects available through DataConnect. The following general rules apply to this data:

- Fields designated as required will always have values.
- Fields designated as optional may have values.
- The Data Types defined in the table below are used in the following [Object Definitions](#).

Data Types

The following Data Types are the specifications or type definitions that can be assigned to each element of the data objects:

Data Type	Description
BOOLEAN	Value is either 1 (indicating TRUE) or 0 (indicating FALSE).
CHARn	Alphanumeric string containing up to n characters.
DATE	Date in the form YYYYMMDD , where YYYY is a 4-digit year (e.g., 2003), MM is a 2-digit month code from 01 (January) through 12 (December), and DD is a 2-digit day code from 01 to 31.
NUMBER	Amounts, quantities, and prices use this numeric data type that can contain up to 39 numeric characters including the decimal point. Negative values are indicated by a leading minus sign (-). Values that do not represent whole numbers (e.g., 1.3504) include a decimal point to indicate the start of the fractional amount. No punctuation is used to separate thousands, millions, etc.
PWDn	Password of length n . Password values are WRITE-ONLY, meaning a password can be set but its value can never be retrieved for security reasons. A special mechanism is provided to indicate whether a value exists for a password field without providing the actual password value.
TIMESTAMP	Provides a full time stamp, including time zone of the DataConnect server. Currently used to communicate data expiration times to DataConnect clients. The timestamp uses the form: YYYYMMDDHHmmSS [{gmt-offset}:{tz-name}] where: YYYY is a 4-digit year (e.g., 2003), MM is a 2-digit month code from 01 (January) through 12 (December), DD is a 2-digit day code from 01 to 31, HH is a 2-digit hour code in 24-hour format (00 through 23), mm is a 2-digit minute code (00 through 59), SS is a 2-digit seconds code (00 through 59) gmt-offset is the number of hours that the time zone is offset from GMT; has a leading + or - tz-name is the name of the time zone (e.g., EST). Example: 20030721143522 [-5:EST] is July 21, 2003 2:35:22 PM, Eastern Standard Time

UID	Unique persistent numeric identifier (Unique ID) ranging from 0 to 9999999999999999. Fields of this Data Type are either a primary key or a foreign key and are designated as such in the field Data Type definition. All ID fields have values generated and maintained by ByAllAccounts. An object's ID is persistent across calls to DataConnect (does not change from one day to the next). ID values for a field are monotonically increasing (e.g., ID values assigned to a new object tomorrow are greater than ID values assigned to a new object created today).
------------	---

Object Definitions

The following tables provide detailed descriptions of the fields of the objects in the DataConnect Data Model, including their Data Type and whether they are required or optional.

Note: Required information is indicated with a √.

Security

Field	Required	Data Type	Description
ID	√	UID (PKEY)	Unique numeric ID for this Security
NAME	√	CHAR128	Name for this Security (e.g., "Ford Motor Company")
SECTYPE	√	CHAR20	Type of Security, one of: <ul style="list-style-type: none"> ▪ BOND ▪ CASH ▪ MUTUALFUND ▪ OPTION ▪ OTHER ▪ STOCK
TICKER		CHAR32	Ticker symbol for this Security (e.g., "F"), if available
CUSIP		CHAR9	Committee on Uniform Security Identification Procedures (CUSIP) for this Security (e.g., "345370860"), if available. Will only be present if the firm is licensed for CUSIP data and the firm is enabled to deliver it.
ASSET_CLASS		CHAR64	The asset class of this security, one of: <ul style="list-style-type: none"> ▪ Unclassified ▪ Stocks ▪ Bonds ▪ Cash ▪ Real Estate ▪ Other <p>The INCSECAC option must be used to have this data included in a DATAGET response.</p>
ASSET_SUBCLASS		CHAR64	The asset subclass of this security, one of: <ul style="list-style-type: none"> ▪ Unclassified ▪ Large Cap Growth (US) ▪ Large Cap Core (US) ▪ Large Cap Value (US) ▪ Mid Cap Growth (US) ▪ Mid Cap Core (US) ▪ Mid Cap Value (US)

Security (continued):

Field	Required	Data Type	Description
			<ul style="list-style-type: none"> ▪ Small Cap Growth (US) ▪ Small Cap Core (US) ▪ Small Cap Value (US) ▪ Large Cap Growth (Emerging Foreign) ▪ Large Cap Core (Emerging Foreign) ▪ Large Cap Value (Emerging Foreign) ▪ Mid Cap Growth (Emerging Foreign) ▪ Mid Cap Core (Emerging Foreign) ▪ Mid Cap Value (Emerging Foreign) ▪ Small Cap Growth (Emerging Foreign) ▪ Small Cap Core (Emerging Foreign) ▪ Small Cap Value (Emerging Foreign) ▪ Large Cap Growth (Developed Foreign) ▪ Large Cap Core (Developed Foreign) ▪ Large Cap Value (Developed Foreign) ▪ Mid Cap Growth (Developed Foreign) ▪ Mid Cap Core (Developed Foreign) ▪ Mid Cap Value (Developed Foreign) ▪ Small Cap Growth (Developed Foreign) ▪ Small Cap Core (Developed Foreign) ▪ Small Cap Value (Developed Foreign) ▪ Unclassified Stocks ▪ Invest. Grade Short (US Tax Exempt) ▪ Invest. Grade Intermediate (US Tax Exempt) ▪ Invest. Grade Long (US Tax Exempt) ▪ Medium Grade Short (US Tax Exempt) ▪ Medium Grade Intermediate (US Tax Exempt) ▪ Medium Grade Long (US Tax Exempt) ▪ High Yield Short (US Tax Exempt) ▪ High Yield Intermediate (US Tax Exempt) ▪ High Yield Long (US Tax Exempt) ▪ Invest. Grade Short (US Taxable) ▪ Invest. Grade Intermediate (US Taxable) ▪ Invest. Grade Long (US Taxable) ▪ Medium Grade Short (US Taxable) ▪ Medium Grade Intermediate (US Taxable) ▪ Medium Grade Long (US Taxable) ▪ High Yield Short (US Taxable) ▪ High Yield Intermediate (US Taxable)

Security (continued):

Field	Required	Data Type	Description
			<ul style="list-style-type: none"> ▪ High Yield Long (US Taxable) ▪ Invest. Grade Short (Foreign) ▪ Invest. Grade Intermediate (Foreign) ▪ Invest. Grade Long (Foreign) ▪ Medium Grade Short (Foreign) ▪ Medium Grade Intermediate (Foreign) ▪ Medium Grade Long (Foreign) ▪ High Yield Short (Foreign) ▪ High Yield Intermediate (Foreign) ▪ High Yield Long (Foreign) ▪ Unclassified Bonds ▪ Cash ▪ Real Estate ▪ Hedge Funds ▪ Private Equity Investments ▪ Options & Futures ▪ Precious Metals ▪ Natural Resources ▪ Other Investments ▪ Unclassified Other <p>The INCSECAC option must be used to have this data included in a DATAGET response.</p>
CLOSE_PRICE		NUMBER	The closing price of the security for the date in SEC_DATA_AS_OF. The INCSECDETAIL option must be used to have this data included in a DATAGET response.
SEC_DATA_AS_OF		DATE	The date for which CLOSE_PRICE is the closing price of the security. The INCSECDETAIL option must be used to have this data included in a DATAGET response.
BOND_MATURITY		DATE	The maturity date for a bond.
BOND_COUPON		NUMBER	The coupon rate for the bond expressed as a percentage, e.g. 7.5 or 6.625.
MORNINGSTAR_SECID		CHAR10	The Morningstar investment identifier.
MORNINGSTAR_PERFID		CHAR10	For equities, this field contains the Morningstar performance identifier.
STYLEBOX_CODE		NUMBER	Number representation of Morningstar investment style of stocks and portfolios. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STYLEBOX_NAME		CHAR12	String representation of Morningstar investment style of stocks and portfolios. Only included when licensed for the firm and the INCEXTRASECINFO option is used.

SECTOR_CODE		NUMBER	Number representation of Morningstar sector code. (Industry groups are consolidated into 11 sectors). Only included when licensed for the firm and the INCEXTRASECINFO option is used.
SECTOR_CODE_NAME		CHAR22	String representation of Morningstar Sector code. (Industry groups are consolidated into 11 sectors). Only included when licensed for the firm and the INCEXTRASECINFO option is used.
BUSINESS_COUNTRY_ID		CHAR03	ISO code of the business country of the security. For example: USA. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
BONDSTYLEBOX_LONG		NUMBER	This model is based on the two pillars of fixed-income performance: interest-rate sensitivity and credit quality. The three duration groups are short, intermediate, and long-term, and the three credit quality groups are high, medium, and low quality. These groupings display a portfolio's effective duration and credit quality to provide an overall representation of the fund's risk, given the length and quality of bonds in its portfolio. As with equity funds, nine possible combinations exist, ranging from short duration/high quality for the safest funds to long duration/low quality for the riskiest. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
BONDSTYLEBOX_LONGNAME		CHAR50	This model is based on the two pillars of fixed-income performance: interest-rate sensitivity and credit quality. The three duration groups are short, intermediate, and long-term, and the three credit quality groups are high, medium, and low quality. These groupings display a portfolio's effective duration and credit quality to provide an overall representation of the fund's risk, given the length and quality of bonds in its portfolio. As with equity funds, nine possible combinations exist, ranging from short duration/high quality for the safest funds to long duration/low quality for the riskiest. For example: Limited Sensitivity Low Quality. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
ASSETALLOCATION_USSTOCK		NUMBER	The percentage of the fund's assets in US Stocks (net). This figure is calculated separately for the short and long positions of the portfolio, and the sum of the asset allocation of each will not necessarily equal 100%. The net value is derived by subtracting the short positions from the long. The long and short positions can be rescaled as well. Rescaling ensures that the sum of the asset allocation breakdown will sum to 100%. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
ASSETALLOCATION_NONUSSTOCK		NUMBER	The percentage of the fund's assets in Non US stocks (net). Only included when licensed for the firm and the INCEXTRASECINFO option is used.

ASSETALLOCATION_USBOND		NUMBER	The percentage of the fund's assets in US bonds (net). Only included when licensed for the firm and the INCEXTRASECINFO option is used.
ASSETALLOCATION_NONUSBOND		NUMBER	The percentage of the fund's assets in Non US Bonds (net). Only included when licensed for the firm and the INCEXTRASECINFO option is used.
ASSETALLOCATION_PREFERRED		NUMBER	The percentage of the fund's assets in preferred stocks (net). Only included when licensed for the firm and the INCEXTRASECINFO option is used.
ASSETALLOCATION_CONVERTIBLE		NUMBER	The percentage of the fund's assets in convertibles (net). Only included when licensed for the firm and the INCEXTRASECINFO option is used.
ASSETALLOCATION_CASH		NUMBER	The percentage of the fund's assets in cash (net). Only included when licensed for the firm and the INCEXTRASECINFO option is used.
ASSETALLOCATION_OTHER		NUMBER	The percentage of the fund's assets in other instruments (net). Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_UNITEDSTATES		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in the United States. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_CANADA		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in Canada. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_LATINAMERICA		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in Latin America. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.

REGION_UNITEDKINGDOM		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in the United Kingdom. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_EUROZONE		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in the Eurozone region. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_EUROPEEXEURO		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in the Europe Ex Euro region. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_EUROPEEMERGING		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in the Europe Emerging region. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_AFRICA		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in Africa. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_MIDDLEEAST		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in the Middle East. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.

REGION_JAPAN		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in Japan. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_AUSTRALASIA		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in the Australasia region. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_ASIADEVELOPED		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in the Asia Developed region. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
REGION_ASIAEMERGING		NUMBER	This data set provides a broad breakdown of an investment's geographical exposure in the Asia Emerging region. Each region's exposure is presented as a percentage of non-cash equity assets held by the fund. Regional exposure information summarizes a portfolio's exposure to geopolitical risk. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STOCKSECTOR_BASICMATERIALS		NUMBER	The percentage of the fund's assets that are invested in Basic Materials (rescaled long positions). The Basic Materials sector includes companies that manufacture chemicals, building materials, and paper products. This sector also includes companies engaged in commodities exploration and processing. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STOCKSECTOR_CONSUMERCYCLICAL		NUMBER	The percentage of the fund's assets that are invested in the Consumer Cyclical sector (rescaled long positions). The Consumer Cyclical sector includes retail stores, auto and auto parts manufacturers, companies engaged in residential construction, lodging facilities, restaurants, and entertainment companies. Only included when licensed for the firm and the INCEXTRASECINFO option is used.

STOCKSECTOR_FINANCIAL SERVICES		NUMBER	The percentage of the fund's assets that are invested in the Financial Services sector (rescaled long positions). The Financial Services sector includes companies that provide financial services (banks, savings and loans, asset management companies, credit services, investment brokerage firms, and insurance companies). Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STOCKSECTOR_REALESTATE		NUMBER	The percentage of the fund's assets that are invested in the Real Estate sector (rescaled long positions). The Real Estate sector includes mortgage companies, property management companies, and REITs. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STOCKSECTOR_CONSUMER DEFENSIVE		NUMBER	The percentage of the fund's assets that are invested in the Consumer Defensive sector (rescaled long positions). The Consumer Defensive sector includes companies engaged in the manufacturing of food, beverages, household and personal products, packaging, or tobacco. Also includes companies that provide services such as education and training services. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STOCKSECTOR_HEALTHCARE		NUMBER	The percentage of the fund's assets that are invested in the Healthcare sector (rescaled long positions). The Healthcare sector includes biotechnology, pharmaceuticals, research services, home healthcare, hospitals, long-term care facilities, and medical equipment and supplies. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STOCKSECTOR_UTILITIES		NUMBER	The percentage of the fund's assets that are invested in the Utilities sector (rescaled long positions). The Utilities sector includes electric, gas, and water utilities. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STOCKSECTOR_COMMSERVICES		NUMBER	The percentage of the fund's assets that are invested in the Communication Services sector (rescaled long positions). The Communication Services sector includes companies that provide communication services using fixed-line networks or those that provide wireless access and services. This sector also includes companies that provide internet services such as access, navigation, and internet related software and services. Only included when licensed for the firm and the INCEXTRASECINFO option is used.

STOCKSECTOR_ENERGY		NUMBER	The percentage of the fund's assets that are invested in the Energy sector (rescaled long positions). The Energy sector includes companies that produce or refine oil and gas, oil field services and equipment companies, and pipeline operators. This sector also includes companies engaged in the mining of coal. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STOCKSECTOR_INDUSTRIALS		NUMBER	The percentage of the fund's assets that are invested in the Industrials sector (rescaled long positions). The Industrials sector includes companies that manufacture machinery, hand-held tools, and industrial products. This sector also includes aerospace and defense firms as well as companies engaged in transportations and logistic services. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STOCKSECTOR_TECHNOLOGY		NUMBER	The percentage of the fund's assets that are invested in the Technology sector (rescaled long positions). The Technology sector includes companies engaged in the design, development, and support of computer operating systems and applications. This sector also includes companies that provide computer technology consulting services. Also includes companies engaged in the manufacturing of computer equipment, data storage products, networking products, semiconductors, and components. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
BONDSUPERSECTOR_GOVERNMENT		NUMBER	The government Super Sector includes all conventional debt issued by governments other than those which are included in the Municipal sector, including bonds issued by a Central Bank or Treasury, and bonds issued by local governments, cantons, regions, and provinces. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
BONDSUPERSECTOR_MUNICIPAL		NUMBER	The municipal Super Sector includes taxable and tax-exempt debt obligations issued under the auspices of states, cities, counties, provinces, and other nonfederal government entities. This sector includes issues of private entities which are considered to municipal issues from a regulatory perspective. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
BONDSUPERSECTOR_CORPORATE		NUMBER	The corporate Super Sector includes bank loans, convertible bonds, conventional debt securities issued by corporations, and preferred stock. Only included when licensed for the firm and the INCEXTRASECINFO option is used.

BONDSUPERSECTOR_SECURITIZED		NUMBER	The securitized Super Sector includes all types of mortgage-backed securities, covered bonds, and asset-backed securities. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
BONDSUPERSECTOR_CASH EQUIVALENT		NUMBER	The cash and equivalents Super Sector includes cash in the bank, certificates of deposit, currency, and money market holdings. Cash can also be any fixed-income securities that mature in fewer than 92 days. This Super Sector also includes commercial paper, and any repurchase agreements held by the fund. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
BONDSUPERSECTOR_DERIVATIVE		NUMBER	The derivative Super Sector includes the common types of fixed-income derivative contracts: futures and forwards, options, and swaps. For display purposes, products may elect to identify this sector as Other. Only included when licensed for the firm and the INCEXTRASECINFO option is used.
STYLEBOX_LARGEVALUE		NUMBER	Top row, left corner quadrant of the Morningstar Style Box™. The Morningstar Style Box™ is a nine-square grid – three stock investment styles for each of three size categories: small, mid, and large. Two of the three style categories are “value” and “growth.” However, the third, central column definition differs: for funds, the central column represents “blend” funds, which include value, core, and growth stocks; for stocks, it represents “core” stocks, those for which neither growth characteristics nor value characteristics are dominant. A stock is classified as large, mid, or small based on its position in the cumulative market capitalization of its style zone. Large-cap stocks are those that together account for the top 70% of the capitalization of each style zone; mid-cap stocks represent the next 20%; and small-cap stocks represent the balance. The market caps that correspond to these breakpoints are flexible and may shift from month to month as the market changes. Only included when licensed for the firm and the INCEXTRASECINFO option is used.

<p>STYLEBOX_LARGEBLEND</p>		<p>NUMBER</p>	<p>Top row, middle column quadrant of the Morningstar Style Box™. The Morningstar Style Box™ is a nine-square grid – three stock investment styles for each of three size categories: small, mid, and large. Two of the three style categories are “value” and “growth.” However, the third, central column definition differs: for funds, the central column represents “blend” funds, which include value, core, and growth stocks; for stocks, it represents “core” stocks, those for which neither growth characteristics nor value characteristics are dominant. A stock is classified as large, mid, or small based on its position in the cumulative market capitalization of its style zone. Large-cap stocks are those that together account for the top 70% of the capitalization of each style zone; mid-cap stocks represent the next 20%; and small-cap stocks represent the balance. The market caps that correspond to these breakpoints are flexible and may shift from month to month as the market changes. Only included when licensed for the firm and the INCEXTRASECINFO option is used.</p>
<p>STYLEBOX_LARGE GROWTH</p>		<p>NUMBER</p>	<p>Top row, right corner quadrant of the Morningstar Style Box™. The Morningstar Style Box™ is a nine-square grid – three stock investment styles for each of three size categories: small, mid, and large. Two of the three style categories are “value” and “growth.” However, the third, central column definition differs: for funds, the central column represents “blend” funds, which include value, core, and growth stocks; for stocks, it represents “core” stocks, those for which neither growth characteristics nor value characteristics are dominant. A stock is classified as large, mid, or small based on its position in the cumulative market capitalization of its style zone. Large-cap stocks are those that together account for the top 70% of the capitalization of each style zone; mid-cap stocks represent the next 20%; and small-cap stocks represent the balance. The market caps that correspond to these breakpoints are flexible and may shift from month to month as the market changes. Only included when licensed for the firm and the INCEXTRASECINFO option is used.</p>

<p>STYLEBOX_MIDVALUE</p>		<p>NUMBER</p>	<p>Middle row, left column quadrant of the Morningstar Style Box™. The Morningstar Style Box™ is a nine-square grid – three stock investment styles for each of three size categories: small, mid, and large. Two of the three style categories are “value” and “growth.” However, the third, central column definition differs: for funds, the central column represents “blend” funds, which include value, core, and growth stocks; for stocks, it represents “core” stocks, those for which neither growth characteristics nor value characteristics are dominant. A stock is classified as large, mid, or small based on its position in the cumulative market capitalization of its style zone. Large-cap stocks are those that together account for the top 70% of the capitalization of each style zone; mid-cap stocks represent the next 20%; and small-cap stocks represent the balance. The market caps that correspond to these breakpoints are flexible and may shift from month to month as the market changes. Only included when licensed for the firm and the INEXTRASECINFO option is used.</p>
<p>STYLEBOX_MIDBLEND</p>		<p>NUMBER</p>	<p>Middle row, middle column quadrant of the Morningstar Style Box™. The Morningstar Style Box™ is a nine-square grid – three stock investment styles for each of three size categories: small, mid, and large. Two of the three style categories are “value” and “growth.” However, the third, central column definition differs: for funds, the central column represents “blend” funds, which include value, core, and growth stocks; for stocks, it represents “core” stocks, those for which neither growth characteristics nor value characteristics are dominant. A stock is classified as large, mid, or small based on its position in the cumulative market capitalization of its style zone. Large-cap stocks are those that together account for the top 70% of the capitalization of each style zone; mid-cap stocks represent the next 20%; and small-cap stocks represent the balance. The market caps that correspond to these breakpoints are flexible and may shift from month to month as the market changes. Only included when licensed for the firm and the INEXTRASECINFO option is used.</p>

<p>STYLEBOX_MIDGROWTH</p>		<p>NUMBER</p>	<p>Middle row, right column quadrant of the Morningstar Style Box™. The Morningstar Style Box™ is a nine-square grid – three stock investment styles for each of three size categories: small, mid, and large. Two of the three style categories are “value” and “growth.” However, the third, central column definition differs: for funds, the central column represents “blend” funds, which include value, core, and growth stocks; for stocks, it represents “core” stocks, those for which neither growth characteristics nor value characteristics are dominant. A stock is classified as large, mid, or small based on its position in the cumulative market capitalization of its style zone. Large-cap stocks are those that together account for the top 70% of the capitalization of each style zone; mid-cap stocks represent the next 20%; and small-cap stocks represent the balance. The market caps that correspond to these breakpoints are flexible and may shift from month to month as the market changes. Only included when licensed for the firm and the INEXTRASECINFO option is used.</p>
<p>STYLEBOX_SMALLVALUE</p>		<p>NUMBER</p>	<p>Bottom row, left column quadrant of the Morningstar Style Box™. The Morningstar Style Box™ is a nine-square grid – three stock investment styles for each of three size categories: small, mid, and large. Two of the three style categories are “value” and “growth.” However, the third, central column definition differs: for funds, the central column represents “blend” funds, which include value, core, and growth stocks; for stocks, it represents “core” stocks, those for which neither growth characteristics nor value characteristics are dominant. A stock is classified as large, mid, or small based on its position in the cumulative market capitalization of its style zone. Large-cap stocks are those that together account for the top 70% of the capitalization of each style zone; mid-cap stocks represent the next 20%; and small-cap stocks represent the balance. The market caps that correspond to these breakpoints are flexible and may shift from month to month as the market changes. Only included when licensed for the firm and the INEXTRASECINFO option is used.</p>

<p>STYLEBOX_SMALLBLEND</p>		<p>NUMBER</p>	<p>Bottom row, middle column quadrant of the Morningstar Style Box™. The Morningstar Style Box™ is a nine-square grid – three stock investment styles for each of three size categories: small, mid, and large. Two of the three style categories are “value” and “growth.” However, the third, central column definition differs: for funds, the central column represents “blend” funds, which include value, core, and growth stocks; for stocks, it represents “core” stocks, those for which neither growth characteristics nor value characteristics are dominant. A stock is classified as large, mid, or small based on its position in the cumulative market capitalization of its style zone. Large-cap stocks are those that together account for the top 70% of the capitalization of each style zone; mid-cap stocks represent the next 20%; and small-cap stocks represent the balance. The market caps that correspond to these breakpoints are flexible and may shift from month to month as the market changes. Only included when licensed for the firm and the INEXTRASECINFO option is used.</p>
<p>STYLEBOX_SMALLGROWTH</p>		<p>NUMBER</p>	<p>Bottom row, right column quadrant of the Morningstar Style Box™. The Morningstar Style Box™ is a nine-square grid – three stock investment styles for each of three size categories: small, mid, and large. Two of the three style categories are “value” and “growth.” However, the third, central column definition differs: for funds, the central column represents “blend” funds, which include value, core, and growth stocks; for stocks, it represents “core” stocks, those for which neither growth characteristics nor value characteristics are dominant. A stock is classified as large, mid, or small based on its position in the cumulative market capitalization of its style zone. Large-cap stocks are those that together account for the top 70% of the capitalization of each style zone; mid-cap stocks represent the next 20%; and small-cap stocks represent the balance. The market caps that correspond to these breakpoints are flexible and may shift from month to month as the market changes. Only included when licensed for the firm and the INEXTRASECINFO option is used.</p>

FI (Financial Institution or Financial Service)

Field	Required	Data Type	Description
ID	√	UID (PKEY)	Unique numeric ID for this Financial Service.
NAME	√	CHAR128	Name of Financial Service. The ID for a Financial Service will not change, but Name might change should Financial Institution rename its service.
SERVICE_CATEGORY	√	CHAR64	Category for Financial Service. The category is used primarily as an aid to help users of the service distinguish between different services offered by same Financial Institution. Field may contain one of the following values (list is subject to change) and may change at any time for a particular Financial Service: Banking Brokerage Credit Card Health Insurance Investment Loan Other
USERLOGIN_URL		CHAR1000	The URL for the Financial Service where the User can log in and access their Accounts.
REQ_CHANGEPW	*	BOOLEAN	If 1 (true), the service is an OFX Direct data service and requires that the User's password be changed on first login.
REQ_SECONDPW	*	BOOLEAN	If 1 (true), the Financial Service requires a second password for access.
REQ_MULTI_ACCT	*	BOOLEAN	If 1 (true), all Accounts available under a single set of credentials at a Financial Service must be created in ByAllAccounts prior to gathering data for this service.
REQ_DOWNLOAD_LOGIN	*	BOOLEAN	If 1 (true), the Financial Service requires a Login or password for data download that is different from the Login or password used by the User to log into the Financial Service web site.

FI (Financial Institution or Financial Service) (continued):

Field	Required	Data Type	Description
REQ_INITIAL_LOGIN	*	BOOLEAN	If 1 (true), the User must log into the Financial Service web site prior to asking the service to gather data from that site. Typically, the User must enter information, perform preference selection, or choose a new password before the access is fully enabled at the Financial Service.
REQ_SQA	*	BOOLEAN	If 1 (true), the Financial Service requires answers to security questions to gain access.
ACCESS_INSTRUCTIONS	√	CHAR2000	Provided in HTML format (encoded in a CDATA Section in Response XML files), these instructions further identify the Financial Service, provide contact information at the Financial Institution for that service, and explain or provide links to how to enable online access to the service.
ACCOUNT_NUMBER_TERM		CHAR64	The term used by the Financial Service to refer to the ACCOUNT_NUMBER field for the account.
ACCOUNT_NUMBER_DETAILS		CHAR500	Provides instructions for entering the ACCOUNT_NUMBER .
ACCOUNT_LOGIN_TERM		CHAR64	The term used by the Financial Service to refer to the ACCOUNT_LOGIN field for the Account.
ACCOUNT_LOGIN_DETAILS		CHAR500	Provides instructions for entering the ACCOUNT_LOGIN .
ACCOUNT_PIN_TERM		CHAR64	The term used by the Financial Service to refer to the ACCOUNT_PIN field for the Account.
ACCOUNT_PIN_DETAILS		CHAR500	Provides instructions for entering the ACCOUNT_PIN .
ACCOUNT_SECOND_PIN_TERM		CHAR64	The term used by the Financial Service to refer to the ACCOUNT_SECOND_PIN field for the Account.
ACCOUNT_SECOND_PIN_DETAILS		CHAR500	Provides instructions for entering the ACCOUNT_SECOND_PIN .
ACCOUNT_NUMBER_2_TERM		CHAR64	The term used by the Financial Service to refer to the ACCOUNT_NUMBER_2 field for the Account.
ACCOUNT_NUMBER_2_DETAILS		CHAR500	Provides instructions for entering the ACCOUNT_NUMBER_2 .
ACCOUNT_LOGIN_2_TERM		CHAR64	The term used by the Financial Service to refer to the ACCOUNT_LOGIN_2 field for the Account.
ACCOUNT_LOGIN_2_DETAILS		CHAR500	Provides instructions for entering the ACCOUNT_LOGIN_2 .
SQA_QUESTION_TERM		CHAR64	The term used by the Financial Service to refer to the security question(s).

FI (Financial Institution or Financial Service) (continued):

Fields	Required	Data Type	Description
SQA_ANSWER_TERM		CHAR64	The term used by the Financial Service to refer to the answers for the security question(s).
SQA_INSTRUCTIONS		CHAR500	Provides instructions for entering the security questions and answers (SQA).
SUP_SETTLEMENT **	√	CHAR20	Provides information on the support for settlement-based data retrieval at this Financial Institution. One of: UNKNOWN, UNAVAILABLE, UNSUPPORTED, SUPPORTED
SUP_TRADE **	√	CHAR20	Provides information on the support for trade-based data retrieval at this Financial Institution. One of: UNKNOWN, UNAVAILABLE, UNSUPPORTED, SUPPORTED
SUP_ACCOUNT_DISCOVERY **	√	CHAR20	Provides information on the support for the Discover Account capability at this Financial Institution. One of: UNKNOWN, UNAVAILABLE, UNSUPPORTED, SUPPORTED
SUP_TRANSACTIONS **	√	CHAR20	Provides information on the support for Transaction data at this Financial Institution. One of: UNKNOWN, UNAVAILABLE, UNSUPPORTED, SUPPORTED
SUP_TAX_LOTS **	√	CHAR20	Provides information on the support for Tax Lot data at this Financial Institution. One of: UNKNOWN, UNAVAILABLE, UNSUPPORTED, SUPPORTED
SUP_TEST_CREDENTIAL **	√	CHAR20	Provides information on the support for Test Credential at this Financial Institution. One of: UNKNOWN, UNAVAILABLE, UNSUPPORTED, SUPPORTED
POPULAR	*	BOOLEAN	If 1 (true) then the financial institution is popular.
SUP_AAM **	√	CHAR20	Provides information on the support for Automatic Account Management (AAM) at this Financial Institution. One of: UNKNOWN, UNAVAILABLE, UNSUPPORTED, SUPPORTED
REQ_IN_SESSION_ACTIVATION	√	BOOLEAN	If 1 (true), then the institution may require the completion of an in-session activation workflow to enable the aggregation service to access your accounts at that institution. If 0 (false), then the institution is not known to require in-session activation at the present time.
SETUP_INSTRUCTIONS		CHAR3000	When present, these instructions identify setup instructions for the Financial Service. Provided in HTML format (encoded in a CDATA Section in Response XML files).

FI (Financial Institution or Financial Service) (continued):

Fields	Required	Data Type	Description
SUP_OAUTH**	√	CHAR20	Defines whether this FI uses OAuth authentication. The value is one of: UNKNOWN, UNAVAILABLE, UNSUPPORTED, or SUPPORTED. If the value is SUPPORTED then the FI requires OAuth type authentication, otherwise the FI requires LOGIN type authentication.
IS_FEED	√	BOOLEAN	A 1 (true) indicates that institution provides data as a feed, which is typically a file set available over SFTP (versus a web site). If 0 (false), then it does not.
KNOWN_ISSUE_LONGTERM		CHAR500	This field provides information about known long-term issues, if any, associated with the FI.
HEALTH_STATUS		CHAR20	This field indicates the health status of the FI. If this field is not included, the health status for this financial institution is not calculated yet. INSUFFICIENTUSAGE - there has not been enough usage to confidently determine a likely aggregation outcome. GREEN - users should be able to successfully link their account(s). YELLOW - there is a connectivity issue that may impact users' ability to successfully link their account(s). RED - unable to connect to this FI.
KNOWN_ISSUE_ADVANCED		CHAR2000	This field provides information about known connectivity issues, if any, at the financial institution. This information is usually very detailed and suitable only for advanced users, and generally should not be given to investors (see KNOWN_ISSUE_BASIC). The information is in HTML format (encoded in CDATA in XML response files). Not returned when the <ALL/> tag is used on <FIGETRQ>.
KNOWN_ISSUE_BASIC		CHAR2000	This field provides information about known connectivity issues, if any, at the financial institution. The information in this field (versus KNOWN_ISSUE_ADVANCED) is simplified and appropriate for delivery to investors. The information is in HTML format (encoded in CDATA in XML response files). Not returned when the <ALL/> tag is used on <FIGETRQ>.

Notes:

* Fields marked “**” are delivered in the response XML in the following way:

- If the value of the field is **1** (TRUE), the empty tag is used as an indicator.
- If the value of the field is **0** (FALSE), the absence of the tag is used as an indicator.
For example,

- **<REQ_CHANGEPW/>** indicates the **REQ_CHANGEPW** field's value is **1** (TRUE).
- Absence of **<REQ_CHANGEPW/>** indicates the field value of **REQ_CHANGEPW** is **0** (FALSE).

** The following are valid values for the financial institution support fields:

- UNKNOWN: It has not been investigated whether the financial institution supports this feature.
- UNAVAILABLE: The institution does not provide the feature.
- UNSUPPORTED: It is possible to support the feature, but not with the currently available technology.
- SUPPORTED: The feature is supported by the institution.

Person

Field	Required	Data Type	Description
ID	√	UID (PKEY)	Unique numeric ID for this Person.
FIRM_TAG1		CHAR128	A string you assign that enables you to identify a ByAllAccounts User and correlate it with data in your systems. You can use this field in some operations to identify a Financial Profile. If you plan to do this, you should ensure that this field is unique across all of your Persons.
FIRM_TAG2		CHAR128	A string used to classify the User.
FIRM_TAG3		CHAR128	A string used to classify the User.
ROLE	√	CHAR20	One of: INVESTOR, ADVISOR, ASSISTANT, or CONSULTANT.
FIRST_NAME	√	CHAR64	Person's first name.
MIDDLE_NAME		CHAR64	Person's middle name.
LAST_NAME	√	CHAR64	Person's last name.
EMAIL_ADDRESS		CHAR64	Person's email address. Required only if a login name is provided.
PHONE		CHAR20	Person's phone number.
TAX_ID		CHAR32	Person's SSN or TIN.
CREATION_DATE	√	DATE	Date when this Person was created.
IS_SSO	√	BOOLEAN	Indicates whether the person has single sign on (SSO) access. Either 1 to indicate true, or 0 to indicate false.

Portfolio

Field	Required	Data Type	Description
FP_ID	√	UID (FKEY: FINANCIAL_PROFILE.ID)	Unique numeric ID for the Financial Profile that contains this Portfolio.
ID	√	UID (PKEY)	Unique numeric ID for this Portfolio.
NAME	√	CHAR64	User-assigned name for the Portfolio. Values are case-sensitive and unique across all Portfolios for a User.

Login

Field	Required	Data Type	Description
PERSON_ID	√	UID (FKEY: PERSON.ID)	Unique numeric ID for the Person identified by this Login.
LOGIN_NAME	√	CHAR32 (PKEY)	Login name used by this Person to log into ByAllAccounts services.
LOGIN_PW	√	PW20	Login password used by this person - goes with LOGIN_NAME . This field is WRITE-ONLY.
PW_EXPIRE_DATE		TIMESTAMP	Date/time on which the password for this Login expires.
PASSWORD_HINT		CHAR128	Text to help the Person remember own password.
LAST_LOGIN_DATE		DATE	When the last login date is available, it is included in this field.

Financial Profile

Field	Required	Data Type	Description
ID	√	UID (PKEY)	Unique numeric ID for this Financial Profile.
NAME	√	CHAR208	Name of the profile. Value is the '{Last Name}, {First Name} {Middle Name}' of the Investor who owns this profile.
CREATION_DATE	√	DATE	Date when the profile was created.

Profile Access

Field	Required	Data Type	Description
PERSON_ID	√	UID (FKEY: PERSON.ID)	Unique numeric ID for the Person to whom access is granted.
PROFILE_ID	√	UID (FKEY: FINANCIAL_PROFILE.ID)	Unique numeric ID for the Financial Profile to which access is granted.
ROLE	√	CHAR20	The role the Person has in relation to the profile. One of: INVESTOR, ADVISOR, ASSISTANT, or CONSULTANT.
ACCESS	√	CHAR20	The level of access granted to the Profile. One of: NONE READ READLIMITEDWRITE READWRITE

Account Credential

Fields preceded by the symbol **Ø** are WRITE-ONLY. Once written, the value can never be retrieved; the response will indicate that there is a value present.

Field	Required	Data Type	Description
PROFILE_ID	√	UID (FKEY: FINANCIAL_PROFILE.ID)	ID for the profile that contains the Account Credential.
ID	√	UID (PKEY)	Unique numeric ID for this Account Credential.
NAME	√	CHAR64	User-assigned name for the Account Credential.
FI_ID	*	UID (FKEY: FI.ID)	ID for the Financial Service that the Account Credential accesses.
FI_REQUEST_NAME	*	CHAR1999 (**)	Used to specify the name of a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this service. See related field FI_REQUEST_URL .
FI_REQUEST_URL	*	CHAR1999 (**)	Used to specify the URL of the login page for a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this service. See related field FI_REQUEST_NAME .
ACCOUNT_LOGIN		CHAR64	Login name used to access the Financial Institution service.
Ø ACCOUNT_PIN		PWD64	Password (goes with ACCOUNT_LOGIN) used to access the Financial Institution service.
Ø ACCOUNT_NEW_PIN		PWD64	Some services require the User to provide a new password on first login. This field specifies the new password to use.
Ø ACCOUNT_SECOND_PIN		PWD64	Some services require a second password to access the Account at that Financial Institution service.
ACCOUNT_LOGIN_2		CHAR64	Some services require more than a single piece of account login information to access an Account. This field stores this second piece of account login information.

Account Credential (continued)

Field	Required	Data Type	Description
CREDENTIAL_COMPLETE	√	BOOLEAN	If 1 (true), this account credential contains the components required to attempt authentication at the associated Financial Institution service; otherwise 0 (false) in all other cases. This field is READ-ONLY.
LAST_AUTHENTICATION_ATTEMPT		TIMESTAMP	Date/time of the last Test Credential, Discover Account, Test Account, or Update Account using this Account Credential or one of its related accounts. This field is READ-ONLY.
AUTHENTICATION_STATUSES_INFO		CHAR1024	The status (textual description) of the last attempt to access this Account Credential to authenticate to the financial institution. This field is READ-ONLY.
AUTHENTICATION_STATUSES_ERRCODE		CHAR6	Error code for the result of the last attempt to access this Account Credential (or one of its accounts) via Test Credential, Test Account, Update Account, or Discover Account. This field is READ-ONLY. See Appendix C: Account Update Status Error Codes for valid values.
AUTO_MANAGE	√	BOOLEAN	If 1 (true) accounts for this credential will be automatically maintained by the nightly aggregation process. Note: automatic management of accounts (AAM) must be enabled for the firm and the financial institution must support it.
DATA_BASIS		CHAR12	Whether TRADE-based ("TRADE") or SETTLEMENT-based ("SETTLEMENT") data should be gathered for the accounts linked to this credential. The value of this field is used only if AUTO_MANAGE set to 1 (true) and the firm allows a choice of Data Basis. When 1 (true), the setting applies to all accounts linked to this Account Credential.
GATHER_LOTS	√	BOOLEAN	If 1 (true) tax lot data should be gathered for accounts linked to this credential. The value of this field is used only if AUTO_MANAGE set to 1 (true) and the lot gathering is enabled for the firm. When 1 (true), the setting applies to all accounts linked to this Account Credential.
AUTH_TYPE	√	CHAR12	This field defines the authentication type of the associated financial institution. Valid values are LOGIN and OAUTH. This field is READ-ONLY.

Account Credential (continued)

Field	Required	Data Type	Description
OAUTH_TOKEN_ID		UID	For ACCOUNT_CREDENTIAL objects of type OAUTH, the aggregation system will set the OAUTH_TOKEN_ID once the end user has successfully completed OAUTH set up for the ACCOUNT_CREDENTIAL at the financial institution. Note that the presence of OAUTH_TOKEN_ID does not guarantee that the accounts associated with the credential can be aggregated.
SQA		SQA	Can have more than one. Security Question and Answer (SQA) object.
FEEDRQ_NAME		CHAR128	Feed request name. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_EMAIL		CHAR64	Feed request email. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_FIRM		CHAR128	Feed request firm. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_LOGIN		CHAR64	Feed request login. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_SUBMITTED_DATE		TIMESTAMP	Read-only timestamp, calculated by the system when all of the FEEDRQ_NAME, FEEDRQ_EMAIL, FEEDRQ_FIRM, and FEEDRQ_LOGIN field values have been provided and the service has sent an email to BAA Custodial Operations (CustOps).

Notes:

* = Only one of **FI_ID** or **FI_REQUEST_NAME** can be provided. If **FI_REQUEST_NAME** is given, **FI_REQUEST_URL** can optionally be provided.

** = **FI_REQUEST_NAME** and **FI_REQUEST_URL** together cannot exceed CHAR1999.

Security Question and Answer (SQA)

Field	Required	Data Type	Description
ID	√	UID (PKEY)	Unique numeric ID for this Security Question and Answer.
AC_ID	√	UID (FKEY: ACCOUNT_CREDENTIAL.ID)	ID for the Account Credential that contains the Security Question and Answer.
QUESTION		CHAR128	Security question that must be answered to access the Financial Institution service.
ANSWER		CHAR128	Answer to security question that is required to access the Financial Institution service.

Account

Field	Required	Data Type	Description
FP_ID	√	UID (FKEY: FINANCIAL_PROFILE.ID)	Unique numeric ID for the Financial Profile that contains this Account.
ID	√	UID (PKEY)	Unique numeric ID for this Account.
NAME	√	CHAR64	User-assigned name for the Account. Values are case-sensitive and unique across all Accounts for a User.
PORTFOLIO_ID	√	UID (FKEY: PORTFOLIO.ID)	ID for the Portfolio to which this Account belongs.
ACCOUNT_TYPE		CHAR32	Account type as determined by the system from information available from the custodian. Possible types are listed in Notes on Account , page 41.
ONLINE_ACCESS_ENABLED	√	BOOLEAN	Is 1 (true) if this account has sufficient online access credentials for the service to gather data from the Financial Institution identified for this Account; in all other cases 0 (false). This field is READ-ONLY.

Account (continued):

Field	Required	Data Type	Description
LAST_UPDATED		TIMESTAMP	Date/time on which data for this Account was (successfully) last retrieved from the associated Financial Institution. May not be available for the off-line accounts. This field is READ-ONLY.
ACCOUNT_NUMBER		CHAR128	Alphanumeric string used to identify this Account at the Financial Service where this Account is held.
ACCOUNT_NUMBER_2		CHAR128	Some services require more than a single piece of account number information to access an Account. This field stores this second piece of account number information.
UPDATE_STATUS_INFO		CHAR1024	The status (textual description) of the last attempt to download this Account from the Financial Institution. This field is READ-ONLY.
UPDATE_STATUS_ERRCODE		CHAR6	Error code for the result of the last attempt to download this Account from the Financial Institution. This field is READ-ONLY. See Appendix C: Account Update Status Error Codes for valid values.
LAST_UPDATE_ATTEMPT		TIMESTAMP	Date/time of the last Test Account or Update Account from FI attempt for this account. Field is READ-ONLY.
CAPTIVE	√	BOOLEAN	If 1 (true), the Advisor managing this Account for the Investor considers this Account to contain captive assets (assets under management).
AC_ID		UID (FKEY: ACCOUNT_CREDENTIAL.ID)	ID for the Account Credential object that provides the credentials for accessing the Account at its Financial Service.
DATA_BASIS	√	CHAR12	States whether the data in the account is SETTLEMENT (value "SETTLEMENT") basis or TRADE basis (value "TRADE").
CREATION_DATE	√	TIMESTAMP	Date/time when the account was created.

Field	Required	Data Type	Description
GATHER_LOTS	√	BOOLEAN	If 1 (true) tax lot data should be gathered for this account. Tax lot data is not gathered by the nightly aggregation process but must be requested through a special Update with Tax Lots operation (ACCTUPDTAXLOTRQ). Note: lot gathering must be enabled for the firm.
FI_SUPPLIED_REP_ID		CHAR32	The FI-supplied Advisor / REP identifier for this account. Only available for accounts at select FIs.
FI_SUPPLIED_FIRM_ID		CHAR32	FI-supplied Firm ID (Advisor's firm) for the account.
FI_SUPPLIED_ACCOUNT_TAXID		CHAR32	FI-supplied TAX ID (SSN or TIN) associated with the account.
FI_SUPPLIED_ACCOUNT_NAME		CHAR128	FI-supplied name for the account.
FI_SUPPLIED_ACCOUNT_TITLE		CHAR128	FI-supplied title for the account.
FI_SUPPLIED_CLIENT_FIRST		CHAR64	FI-supplied first name for the account holder for the account.
FI_SUPPLIED_CLIENT_MIDDLE		CHAR64	FI-supplied middle name for the account holder for the account.
FI_SUPPLIED_CLIENT_LAST		CHAR64	FI-supplied last name for the account holder for the account.
FI_SUPPLIED_ADDRESS_STREET		CHAR64	Street address of the client who owns this account.
FI_SUPPLIED_ADDRESS_LINE2		CHAR64	Additional street address of the client who owns this account.
FI_SUPPLIED_ADDRESS_LINE3		CHAR64	Additional street address of the client who owns this account.
FI_SUPPLIED_ADDRESS_LINE4		CHAR64	Additional street address of the client who owns this account.
FI_SUPPLIED_ADDRESS_LINE5		CHAR64	Additional street address of the client who owns this account.
FI_SUPPLIED_ADDRESS_LINE6		CHAR64	Additional street address of the client who owns this account.
FI_SUPPLIED_ADDRESS_CITY		CHAR32	City portion of the address of the client who owns this account.

Field	Required	Data Type	Description
FI_SUPPLIED_ADDRESS_STATE		CHAR32	State portion of the address of the client who owns this account.
FI_SUPPLIED_ADDRESS_ZIP_CODE		CHAR16	Zip code portion of the address of the client who owns this account.
FI_SUPPLIED_CLIENT_PHONE		CHAR32	Phone number of the client who owns this account. Will stay in the format provided by the custodian.
FI_SUPPLIED_CLIENT_DOB		CHAR12	Date of birth of the client who owns this account in standard date format YYYYMMDD.
FI_SUPPLIED_CLIENT_EMAIL		CHAR64	Email address of the client who owns this account.
FI_SUPPLIED_ACCOUNT_TYPE		CHAR64	Custodian reported account type for this account.
FI_SUPPLIED_CUSTODIAN_CODE		CHAR32	Custodian-reported custodian code.
FI_SUPPLIED_CUSTODIAN_NAME		CHAR128	Custodian-reported custodian name.
PLAN_NAME		CHAR128	Name of the plan.
MARKET_VALUE		NUMBER	The sum of the position values for the account as of the LAST_UPDATED date.
EXTERNAL_SERVICE_LEVEL	√	CHAR20	Specifies how an external application plans to provide service for the account, either POSITIONAL basis (value "POSITIONAL") or TRANSACTIONAL basis (value "TRANSACTIONAL"). This field does not control how the aggregation system handles the account.
END_DATE_LAST_TX		DATE	The date used for determining the starting period for retrieval of transactions on the next aggregation. This field cannot be modified.
MARGIN_TRADING_APPROVED		BOOLEAN	Value of 0 (false) or 1 (true). Indicates whether margin trading is approved for the account.

Field	Required	Data Type	Description
ACCOUNTING_METHOD		CHAR20	<p>Method of accounting. One of:</p> <ul style="list-style-type: none"> ▪ FIFO - First In, First Out ▪ INFI - Intraday First in First out ▪ LIFO - Last In, First Out ▪ LOFO - Lowest Cost First Out ▪ LCST - Lowest Cost Short Term ▪ LCLT - Lowest Cost Long Term ▪ HIFO - Highest Cost First Out ▪ HCST - Highest Cost Short Term ▪ HCLT - Highest Cost Long Term ▪ AVG - Average price method ▪ TXSN - Tax Sensitive ▪ STXSN - Short-Term Tax Sensitive ▪ MGML - Minimize gains and maximize losses ▪ MLMG - Minimize losses and maximize gains ▪ OTHER - The value supplied by the financial institution cannot be mapped <p>Note: Some financial institutions provide different accounting methods by security type. In those cases, this field is populated with the method for equities.</p>

Notes on Account

- Valid values for **ACCOUNT_TYPE** are:

Account Types	
Type	Description
INVESTMENT_CASHMANAGEMENT	Cash management account that provides access to short-term investments such as money market mutual funds and CDs.
BANKING_CD	Certificate of Deposit (CD) is a time deposit financial product commonly sold by banks, thrift institutions, and credit unions.
BANKING_CHECKING	A checking account offers access to money for daily transactional needs via a debit card or checks.
INVESTMENT_COVERDELL	A Coverdell Education Savings Account is a trust or custodial account created or organized in the United States only for the purpose of paying the qualified education expenses of the designated beneficiary of the account.

Account Types	
Type	Description
BANKING_CREDITLINE	A line of credit may take several forms, such as overdraft protection, demand loan, special purpose, export packing credit, term loan, discounting, purchase of commercial bills, traditional revolving credit card account, etc. It is effectively a source of funds that can readily be tapped at the borrower's discretion.
BANKING_DEPOSIT	Interest earning account at a bank or other depository institution, the withdrawals from which are limited to the amount of the account's credit balance.
BANKING_MONEYMARKET	A Money Market Account is a type of savings account that pays interest based on current interest rates in the money markets. The minimum balance for this account is often considerably higher than the minimum balance of a basic savings account.
BANKING_OTHER	Banking account; more specific type is not known.
BANKING_SAVINGS	A savings account is an interest-bearing deposit account held at a bank or another financial institution and which provides a modest interest rate.
CREDITCARD	<p>The issuer of the card (usually a bank) creates a revolving account and grants a line of credit to the cardholder, from which the cardholder can borrow money for payment to a merchant or as a cash advance. Credit cards charge interest and are primarily used for short-term financing.</p> <p>A credit card is different from a charge card, which requires the balance to be repaid in full each month.</p>
HEALTHACCOUNT_FSA	Health Savings account of the type Flexible Spending Account (FSA).
HEALTHACCOUNT_HRA	Health Savings account with Health Reimbursement Arrangement (HRA).
HEALTHACCOUNT_HSA	Health Savings Account (HSA) is a savings account used in conjunction with a high-deductible health insurance policy. The HSA enables users to save money tax-free against medical expenses.
HEALTHACCOUNT_OTHER	Health Savings account; more specific type is not known, may be HSA, FSA, or HRA.
INSURANCE_ANNUITY	A contractual financial product sold by financial institutions that is designed to accept and grow a lump sum from an individual and then pay out a stream of regular disbursements to the individual at a later point in time
INSURANCE_LIFEINSURANCE	A life insurance policy is a contract with an insurance company. In exchange for premium payments, the insurance company provides a lump-sum payment, known as a death benefit, to beneficiaries upon the insured's death.
INSURANCE_OTHER	Insurance account; more specific type is not known.

Account Types	
Type	Description
INVESTMENT_401A	A 401(a) plan is a retirement savings plan in which employees cannot choose or change the amount contributed to the plan. It is also called a “money purchase plan”.
INVESTMENT_401K	A 401(k) is a retirement savings plan sponsored by an employer. It lets workers save and invest a piece of their paycheck before taxes are taken out. Taxes are not paid until the money is withdrawn from the account.
INVESTMENT_403B	The 403b plan is an employer-sponsored supplemental retirement savings plan that, similar to a 401k plan, allows employees to contribute on a pre-tax or (if permitted by the 403b plan) Roth after-tax basis. A 403b plan can only be sponsored by a public school or a 501(c)(3) tax-exempt organization.
INVESTMENT_457B	A 457b plan is a supplemental retirement plan for employees who meet eligibility criteria. Typically, if employer is a governmental entity, state or local law will determine who is eligible to participate.
INVESTMENT_529	A plan operated by a state or educational institution, with tax advantages and potentially other incentives to make it easier to save for college and other post-secondary training for a designated beneficiary, such as a child or grandchild.
INVESTMENT_BROKERAGE	A brokerage account is an arrangement between an investor and a licensed brokerage firm that allows the investor to deposit funds with the firm and buy and sell stocks, bonds, mutual funds, exchange-traded funds and other types of investments through the brokerage.
INVESTMENT_IRA	An Individual Retirement Account (IRA) is a type of savings account that is designed to help individuals save for retirement and offers many tax advantages. There are two different types of IRAs: Traditional and Roth. See also INVESTMENT_ROTHIRA.
INVESTMENT_KEOGH	A Keogh plan is a tax-deferred retirement savings plan for people who are self-employed, and is much like an individual retirement account (IRA). The main difference between a Keogh and an IRA is the contribution limit, with Keogh plans allowing significantly more contributions than IRAs.
INVESTMENT_MUTUALFUND	Mutual Fund Account is an investment program funded by shareholders that trades in diversified holdings and is professionally managed.
INVESTMENT_OTHER	Investment account; more specific type is not known.
INVESTMENT_PENSION	A pension is a retirement account that an employer maintains to provide employee a fixed payout upon retirement retire. It is a defined benefit plan in which the benefit on retirement is determined by a set formula, rather than depending on investment returns.

Account Types	
Type	Description
INVESTMENT_PREPAIDTUITION	Pre-paid tuition plans generally allow college savers to purchase units or credits at participating colleges and universities for future tuition and, in some cases, room and board. Most prepaid tuition plans are sponsored by state governments and have residency requirements. Many state governments guarantee investments in pre-paid tuition plans that they sponsor.
INVESTMENT_PROFITSHARE	A profit-sharing plan is a defined contribution plan in which the employer has discretion to determine when and how much the company pays into the plan. The amount allocated to each individual account is usually based on the salary level of the participant (employee).
INVESTMENT_RETIREMENT	An account containing investments for retirement; more specific type is not known.
INVESTMENT_ROTHIRA	A Roth IRA is a retirement savings account that allows money to grow tax-free. A Roth IRA is funded with after-tax dollars, meaning taxes were already paid on the money deposited. In return for no up-front tax break, money grows and grows tax free, and when withdrawn at retirement, the investor pays no taxes.
INVESTMENT_SARSEP	A SARSEP is a simplified employee pension (SEP) plan set up before 1997 that includes a salary reduction arrangement. Under a SARSEP, employees can choose to have the employer contribute part of their pay to their Individual Retirement Account or Annuity (IRA) set up under the SARSEP (a SEP-IRA).
INVESTMENT_SAVINGSBOND	An account in which one can purchase and manage U.S. Savings Bonds.
INVESTMENT_SEPIRA	A Simplified Employee Pension Individual Retirement Account (SEP IRA) is a variation of the Individual Retirement Account used in the United States. SEP IRAs are adopted by business owners to provide retirement benefits for themselves and their employees.
INVESTMENT_TREASURYBOND	An account in which one can purchase and manage U.S. Treasury Bonds.
INVESTMENT_TRUST	With a bank trust account, the bank serves as custodian and a trustee keeps legal control of assets in the account. These assets can include cash, savings bonds, stocks, bonds, mutual funds, real estate and other property and/or investments.
INVESTMENT_UGMA	A Uniform Gift to Minors Act (UGMA) account is a type of custodial trust account for a minor. It provides a way for minors to own securities without requiring the services of an attorney to prepare trust documents or the court appointment of a trustee.
INVESTMENT_UTMA	Similar to UGMA accounts, a Uniform Transfers to Minors Act (UTMA) account is a type of custodial trust account that also allows minors to own other types of property, such as real estate, fine art, patents and royalties, and for the transfers to occur through inheritance.

Account Types	
Type	Description
LOAN_AUTO	Account used to manage an automobile loan.
LOAN_BANK	Account used to manage a general bank loan.
LOAN_BOAT	Account used to manage a boat loan.
LOAN_HOMEEQUITY	Account used to manage a home equity loan.
LOAN_MORTGAGE	Account used to manage a mortgage.
LOAN_OTHER	Account used to manage a loan. The specific type of loan is not known.
LOAN_PERSONAL	Account used to manage a personal loan.
LOAN_STUDENT	Account used to manage a student loan.
OTHER	An account with an account type that is not represented in our current type set. FI_SUPPLIED_ACCOUNT_TYPE field in file may contain useful information.
UNKNOWN	Unable to determine the type for this account.

Holding

Field	Required	Data Type	Description
FP_ID	√	UID (FKEY: FINANCIAL_PROFILE.ID)	Unique numeric ID for the Financial Profile that contains this Holding.
ID	√	UID (PKEY)	Unique numeric ID for this Holding.
ACCOUNT_ID	√	UID (FKEY: ACCOUNT.ID)	ID for the Account that contains this Holding.
SECURITY_ID		UID (FKEY: SECURITY.ID)	ID for Security owned by this Holding.
UNITS		NUMBER	Units of the Security held. Returns at most six decimal places.
COST_BASIS		NUMBER	Financial Service-supplied cost basis for the Holding. If the cost basis is not available, this field may be populated with a calculated value, derived by multiplying the average Financial Service-supplied per-share cost by the total units of the Holding.
MARKET_VALUE		NUMBER	Total market value of this Holding as retrieved from the Financial Institution or entered by the User. Returns at most four decimal places.
UNIT_PRICE		NUMBER	Price of the Security.
PRICE_DATA_AS_OF		DATE	Date for which UNIT_PRICE is valid.
LAST_UPDATED		TIMESTAMP	Date/time on which the data for this Holding was last updated with information from the Financial Service. Absent for Holdings maintained manually by the User.
NAME		CHAR128	The holding's name (typically identifies the security) as provided by Financial Service or entered by the user (offline accounts).
ASSET_CLASS		CHAR64	The asset class of this holding, one of: Unclassified Stocks Bonds Cash Real Estate Other The INCHOLDAC option must be used to have this data included in a DATAGET response.
ASSET_SUBCLASS		CHAR64	The asset class of this holding. Possible values can be found in the Security object definition table. The INCHOLDAC option must be used to have this data included in a DATAGET response.

Holding (continued):

Field	Required	Data Type	Description
DELETED_ON		DATE	If present, then this holding is a <i>sold-off</i> holding (no longer active in the account). This is the date on which the holding was removed from the account.
FI_SUPPLIED_CUSIP		CHAR20	If the Financial Service supplied a CUSIP for this Holding, it is given here. Will only be present if the firm is licensed for CUSIP data and the firm is enabled to deliver it.
FI_SUPPLIED_TICKER		CHAR32	Any ticker or candidate ticker that the Financial Service supplied for this Holding.
PRINCIPAL_UNITS		NUMBER	Principal units as reported by the Financial Institution. Returns at most six decimal places.
PRINCIPAL_COST_BASIS		NUMBER	Principal cost basis as reported by the Financial Institution.
PRINCIPAL_MARKET_VALUE		NUMBER	Principal market value as reported by the Financial Institution (e.g. PRINCIPAL CASH market value). Returns at most four decimal places.
INCOME_UNITS		NUMBER	Income units as reported by the Financial Institution. Returns at most six decimal places.
INCOME_COST_BASIS		NUMBER	Income cost basis as reported by the Financial Institution.
INCOME_MARKET_VALUE		NUMBER	Income market value as reported by the Financial Institution. Returns at most four decimal places.
DATA_AS_OF		DATE	The date which the holding data is 'as of' as reported by the Financial Institution. If the Financial Institution does not report a date, then this field is not provided.
VALUE_SOURCE	√	CHAR12	The source of the market value for this holding. May be one of the following: <ul style="list-style-type: none"> • FI - the market value was collected from the Financial Institution. • WPAPPROX - the value was approximated by ByAllAccounts using the units reported by the Financial Institution and a closing security price obtained from a third party. USER - the value was edited by the user.
ACCRUED_INCOME		NUMBER	Value of the income that has accrued to the holding but has not yet been distributed.
CURRENCY_CODE		CHAR3	ISO 4217 currency code for MARKET_VALUE and UNIT_PRICE
FI_SUPPLIED_CURRENCY		CHAR64	Currency identifier for MARKET_VALUE and UNIT_PRICE as provided by the Financial Institution.

Holding (continued)

Field	Required	Data Type	Description
FI_SUPPLIED_SEDOL		CHAR7	Security's SEDOL as provided by the Financial Institution. Only included when enabled for the firm.
FI_SUPPLIED_ISIN		CHAR12	Security's ISIN as provided by the Financial Institution. Will only be present if the firm is licensed for CUSIP data and the firm is enabled to deliver it.
COUPON_RATE		NUMBER	The interest payment rate of a debt instrument.
MATURITY_DATE		DATE	The date a debt instrument becomes due and pays in full.
ORIGINAL_FACE		NUMBER	The original face or par value for a security that amortizes or accretes (e.g., a mortgage). For such securities the "current face" is available in the UNITS field.
DETERMINED_SEC_TYPE		CHAR20	Possible values for this field are one of the following, however, as of 3/30/16 this field is only valued as BOND or null. One of: BOND CASH MUTUALFUND OPTION OTHER STOCK
FI_SUPPLIED_SEC_TYPE		CHAR64	Any security type provided by the financial institution. Not commonly valued.
PAYDOWN_FACTOR		NUMBER	The paydown factor value for bond if provided by the financial institution.
ASSET_LIABILITY_INDICATOR	√	CHAR9	Valid values for this field are: <ul style="list-style-type: none">• Asset• Liability
ACCRUED_INCOME_BASE		NUMBER	Value of the income in the base currency that has accrued to the holding but has not yet been distributed.

Holding (continued)

Field	Required	Data Type	Description
ACCRUED_INCOME_LOCAL		NUMBER	Value of the income in the local currency that has accrued to the holding but has not yet been distributed.
CURRENCY_CODE_BASE		CHAR3	ISO 4217 currency code of the base currency of the account as determined from the FI_SUPPLIED_CURRENCY_BASE field.
CURRENCY_CODE_LOCAL		CHAR3	ISO 4217 currency code of the local currency of the holding as determined from the FI_SUPPLIED_CURRENCY_LOCAL field.
FI_SUPPLIED_CURRENCY_BASE		CHAR64	The base currency of the account as provided by the Financial Institution.
FI_SUPPLIED_CURRENCY_LOCAL		CHAR64	The local currency of the holding as provided by the Financial Institution.
EXCHANGE_RATE_LOCAL_TO_BASE		NUMBER	The local-to-base exchange rate as provided by the Financial Institution.
EXCHANGE_RATE_LOCAL_TO_USD		NUMBER	The local-to-USD exchange rate as provided by the Financial Institution.
MARKET_VALUE_BASE		NUMBER	The market value of the holding converted to the base currency of the account.
MARKET_VALUE_LOCAL		NUMBER	The market value of the holding in the local currency of the holding.
PRINCIPAL_MARKET_VALUE_BASE		NUMBER	The principal portion of the market value of the holding in the base currency of the account.
PRINCIPAL_MARKET_VALUE_LOCAL		NUMBER	Principal market value as reported by the Financial Institution (e.g. PRINCIPAL CASH market value).
INCOME_MARKET_VALUE_BASE		NUMBER	Income market value as reported by the Financial Institution (e.g. INCOME CASH market value).
INCOME_MARKET_VALUE_LOCAL		NUMBER	Income market value as reported by the Financial Institution (e.g. INCOME CASH market value).
UNIT_PRICE_BASE		NUMBER	The price of the holding converted to the base currency of the account.
UNIT_PRICE_LOCAL		NUMBER	The price of the holding in the local currency of the holding.
MORNINGSTAR_SECID		CHAR10	Morningstar SECURITY identifier.

Transaction

Field	Required	Data Type	Description
FP_ID	√	UID (FKEY: FINANCIAL_PROFILE.ID)	Unique numeric ID for the Financial Profile that contains this Transaction.
ID	√	UID (PKEY)	Unique numeric ID for this Transaction.
ACCOUNT_ID	√	UID (FKEY: ACCOUNT.ID)	ID of the Account on which this Transaction represents activity.
HOLDING_ID		UID (FKEY: HOLDING.ID)	ID for the Holding on which this Transaction represents activity.
SECURITY_ID		UID (FKEY: SECURITY.ID)	ID for the Security for which this Transaction represents activity.
TX_TYPE	√	CHAR20	Type of Transaction. See the list after this table for valid values.
EXECUTION_DATE	√	DATE	Date on which this Transaction was executed.
SETTLEMENT_DATE		DATE	Date on which this Transaction was settled.
TOTAL_AMOUNT		NUMBER	Total \$ value associated with this Transaction (may be negative).
COMMISSIONS_FEES		NUMBER	Commission and/or fees associated with this Transaction.
UNITS		NUMBER	Number of units (of security) involved in this Transaction.
NAME		CHAR512	Either the name of the Security or a short description of the Transaction.
DESCRIPTION		CHAR2000	Either the name of the Security and/or a longer description of the Transaction.
PRICE		NUMBER	Per share price of the Security for purposes of this Transaction.
FLOW_AMOUNT	√	NUMBER	Total amount of the cash flow for this transaction relative to the cash balance of the account.
FLOW_UNITS		NUMBER	Normalized units for the transaction. Whereas UNITS contains the units value directly from the Financial Institution and may have great variations in sign within a given transaction type, FLOW_UNITS contains this same units value but normalized by transaction type (e.g. Withdrawal will always have negative FLOW_UNITS).

Transaction (continued):

Field	Required	Data Type	Description
CREATION_DATE	√	DATE	Date when this Transaction object was created in the system.
ORIG_TX_TYPE		CHAR20	Transactions reported as a reversal or cancellation by Financial Institution will have the original transaction type code for in this field. Please see the list that follows this table for valid values.
FI_SUPPLIED_CUSIP		CHAR20	Security's CUSIP as provided by the Financial Institution. Will only be present if the firm is licensed for CUSIP data and the firm is enabled to deliver it.
FI_SUPPLIED_TICKER		CHAR32	Security's Ticker as provided by the Financial Institution.
PRINCIPAL_TOTAL_AMOUNT		NUMBER	Principal amount of the transaction as reported by the Financial Institution.
PRINCIPAL_UNITS		NUMBER	Principal units of the transaction as reported by the Financial Institution.
INCOME_TOTAL_AMOUNT		NUMBER	Income amount of the transaction as reported by the Financial Institution.
INCOME_UNITS		NUMBER	Income units of the transaction as reported by the Financial Institution.
ACCRUED_INCOME		NUMBER	Accrued Income associated with the transaction. May be valued for bond purchases and sales or for transactions specifically purchasing or selling accrued interest for a bond.
CONTRACTUAL_SETTLEMENT_DATE		DATE	The date by which the transaction must contractually settle as provided by the Financial Institution.
POST_DATE		DATE	The date that the transaction posted as provided by the Financial Institution.
CURRENCY_CODE		CHAR3	ISO 4217 currency code for the TOTAL_AMOUNT.
FI_SUPPLIED_CURRENCY		CHAR64	Currency identifier supplied by the Financial Institution for TOTAL_AMOUNT.
TOTAL_AMOUNT_LOCAL		NUMBER	The amount of the transaction in the local currency

Transaction (continued)

Field	Required	Data Type	Description
FI_SUPPLIED_ISIN		CHAR12	Security's ISIN as provided by the Financial Institution. Will only be present if the firm is licensed for CUSIP data and the firm is enabled to deliver it.
FI_SUPPLIED_SEDOL		CHAR7	Security's SEDOL as provided by the Financial Institution. Only included when enabled for the firm.
FI_SUPPLIED_TX_TYPE		CHAR64	Text provided by the Financial Institution that describes the type of activity that this transaction represents.
FI_SUPPLIED_TX_TYPE2		CHAR64	Secondary text provided by the Financial Institution that describes the type of activity that this transaction represents.
FI_SUPPLIED_TX_TYPE_CODE		CHAR16	Transaction type code or abbreviation provided by the Financial Institution.
ORIGINAL_FACE		NUMBER	The original face or par value for a security that amortizes or accretes (e.g., a mortgage). For such securities the "current face" is available in the UNITS field.
LOT_ID *		CHAR32	Lot identifier supplied by the Financial Institution.
SUB_LOT_ID *		CHAR32	Identifier supplied by the Financial Institution for a sub-lot (split from an original lot).
OPEN_FAIR_MARKET_VALUE *		NUMBER	Market value of the shares when the lot was opened or transferred into the account
PURCHASE_DATE *		DATE	The date the lot was originally purchased. Used to determine the holding period of the lot. Not necessarily the same as acquisition date. Lot may be purchased on one day then transferred to another account. Purchase date may be listed as 'unknown' or 'various' on the Financial Institution; if so, this field will be left empty.
TRANSFER_DATE *		DATE	Date the lot was transferred in or gifted to the account.
TAXES_WITHHELD *		NUMBER	The fees or commissions paid on the sale of a closed lot.
LOSS_DISALLOWED *		NUMBER	The amount of loss that is disallowed because of a related wash sale.
HOLDING_PERIOD *	√	CHAR20	Can be: SHORT_TERM or LONG_TERM

Transaction (continued)

Field	Required	Data Type	Description
ACQUISITION_METHOD *		CHAR20	The method the lot was acquired. May be one of: <ul style="list-style-type: none"> ▪ GIFT ▪ INHERITANCE ▪ OTHER ▪ PURCHASE ▪ SPLIT ▪ TRANSFER
CLOSE_ACCOUNTING_METHOD *		CHAR26	Method to be used when closing the lot, as provided by the FI. One of: <ul style="list-style-type: none"> ▪ AVG - Average cost, single category ▪ DAVG - Average cost, double category ▪ AVGR - Average cost, single category, whole shares sold ▪ FIFO - First in, first out ▪ LIFO - Last in, first out ▪ MAX - Maximize gain. Sell lowest cost shares first ▪ MIN - Minimize gain; sell highest cost shares first UNKNOWN ▪ VSP - Versus; sell a designated quantity from specific lots
COVERED *		CHAR20	Whether the security is covered by new cost basis regulations. One of: NOT_COVERED or COVERED.
ADJUSTMENT_INDICATOR *		CHAR64	Indication supplied by the Financial Institution of a corporate action, wash or other adjustment on the lot.
COST_SOURCE *		CHAR64	Indication supplied by the Financial Institution of the source of the tax lot information.
DETERMINED_SEC_TYPE		CHAR20	Possible values for this field are one of the following, however, as of 3/30/16 this field is only valued as BOND or null. One of: <ul style="list-style-type: none"> ▪ BOND ▪ CASH ▪ MUTUALFUND ▪ OPTION ▪ OTHER ▪ STOCK

Transaction (continued)

Field	Required	Data Type	Description
COMMISSIONS		NUMBER	Commission associated with this transaction.
FEES		NUMBER	Non-commission fees associated with this transaction.
SPENDING_CATEGORY		CHAR32	The consumer spending category attributed to this transaction. Possible values include but are not limited to: <ul style="list-style-type: none"> ▪ Housing ▪ Utilities ▪ Insurance ▪ Bills ▪ Groceries ▪ Transportation ▪ Clothing ▪ Medical ▪ Childcare ▪ Loans ▪ Entertainment & Restaurants ▪ Travel ▪ Personal Care ▪ Memberships ▪ Luxury ▪ Other Expenses ▪ Paycheck ▪ Deposits ▪ Other Income
SPENDING_CATEGORY_INFO		CHAR7	For consumer spending transactions, specifies if the transaction was an outgoing payment (expense) or incoming receipt (income).
ACCRUED_INCOME_BASE		NUMBER	Accrued Income in base currency.
ACCRUED_INCOME_LOCAL		NUMBER	Accrued Income in local currency.
CURRENCY_CODE_BASE		CHAR3	ISO 4217 currency code for the account's base currency.
CURRENCY_CODE_LOCAL		CHAR3	ISO 4217 currency code for TOTAL_AMOUNT_LOCAL and UNIT_PRICE_LOCAL.
FI_SUPPLIED_CURRENCY_BASE		CHAR64	Currency identifier supplied by the Financial Institution for TOTAL_AMOUNT_BASE.
FI_SUPPLIED_CURRENCY_LOCAL		CHAR64	Local currency of the transaction as provided by the institution.

Transaction (continued)

Field	Required	Data Type	Description
EXCHANGE_RATE_LOCAL_TO_BASE		NUMBER	The local-to-base exchange rate as provided by Financial Institution.
EXCHANGE_RATE_LOCAL_TO_USD		NUMBER	The local-to-USD exchange rate as provided by the Financial Institution.
PRICE_BASE		NUMBER	The price of the holding converted to the base currency of the account.
PRICE_LOCAL		NUMBER	The share price of the security involved in the transaction in the local currency
TOTAL_AMOUNT_BASE		NUMBER	Total amount of the transaction in the Base currency for the account.
PRINCIPAL_TOTAL_AMOUNT_BASE		NUMBER	The principal portion of the amount of the transaction converted to the base currency of the account
PRINCIPAL_TOTAL_AMOUNT_LOCAL		NUMBER	The principal portion of the amount of the transaction in the local currency
INCOME_TOTAL_AMOUNT_BASE		NUMBER	The income portion of the amount of the transaction converted to the base currency of the account
INCOME_TOTAL_AMOUNT_LOCAL		NUMBER	The income portion of the amount of the transaction in the local currency
FI_SUPPLIED_DESCRIPTION		CHAR2000	Transaction description.

Notes on Transaction:

- * indicates transaction fields that are part of the tax lot feature. These fields are primarily related to tax lot capabilities and are not commonly provided by financial institutions.
- Sold-off Holdings are optionally delivered in a response document. If sold-off holdings are not included in the response, Transactions against sold-off Holdings are still delivered. These Transactions include an **ACCOUNT_ID** but no **HOLDING_ID**. Use the INCHOLDINGSO option of the INCHOLDING aggregate to include sold-off holdings in a DATAGET response.

Notes on Transaction (continued):

- Valid values for **TX_TYPE** and **ORIG_TX_TYPE** are:

TX_TYPE Value	Description
ATM	ATM debit or credit (depends on signage of amount)
Buy	Buy a Security
Check	Check written
Closure	Close a holding for an option
Credit	Generic credit
Debit	Generic debit
Deposit	Deposit
Direct debit	Merchant initiated debit
Direct deposit	Direct deposit
Dividend	Dividend paid
Fee	Financial Institution fee
Income	Investment income is realized as cash into the investment Account
Interest	Interest earned or paid (depends on signage of amount)
Expense	Miscellaneous investment expense that is associated with a specific Security
Journal	Journal cash or Securities between Sub-Accounts within the same investment Account
Margin interest	Margin interest expense
Other	Other
Payment	Electronic payment
Point of sale	Point of sale debit or credit (depends on signage of amount)
Reinvestment	Reinvestment of income
Repeat payment	Repeating payment/standing order
Return of capital	Return of capital
Sell	Sell a Security
Service charge	Service charge
Split	Stock or Mutual Fund split
Transfer	Transfer cash or Holdings in or out (depends on signage of amount)
Withdrawal	Withdraw funds from Account

Notes on Transaction (continued):

- The following table defines the sign used for the FLOW_AMOUNT and FLOW_UNITS field in transactions. The sign is based on the transaction type. Signs available are:
 - Positive
 - Negative
 - Neutral - used only by FLOW_AMOUNT, this is a flow of 0
 - As is - the sign in the original data from the Financial Institution. This is usually done to preserve the full meaning of the transaction (e.g. transfer in vs. transfer out are not distinguished by type alone, but by type plus unit sign).

TX_TYPE Value	FLOW_AMOUNT Sign	FLOW_UNITS Sign
ATM	As is	As is
Buy	Negative	Positive
Check	Negative	Negative
Closure	Neutral	As is
Credit	Positive	Positive
Debit	Negative	Negative
Deposit	Positive	Positive
Direct debit	Negative	Negative
Direct deposit	Positive	Positive
Dividend	Positive	Positive
Fee	Negative	Negative
Income	Positive	Positive
Interest	As is	As is
Expense	Negative	Negative
Journal	As is	As is
Margin interest	As is	As is
Other	Neutral	As is
Payment	Negative	Negative
Point of sale	Negative	Negative

Reinvestment	Neutral	Positive
Repeat payment	Negative	Negative
Return of capital	Positive	Positive
Sell	Positive	Negative
Service charge	Negative	Negative
Split	Neutral	As is
Transfer	As is	As is
Withdrawal	Negative	Negative

Investment Option

Field	Required	Data Type	Description
FP_ID	√	UID (FKEY: FINANCIAL_PROFILE.ID)	Unique numeric ID for the Financial Profile that contains this Investment Option.
ID	√	UID (PKEY)	Unique numeric ID for this Investment Option.
ACCOUNT_ID	√	UID (FKEY: ACCOUNT.ID)	ID of the Account in which this Investment Option is available.
SECURITY_ID		UID (FKEY: SECURITY.ID)	ID for the Security this Investment Option represents.
UNIT_PRICE		NUMBER	Price of the Investment Option.
PRICE_DATA_AS_OF		DATE	Date for which UNIT_PRICE is valid.
LAST_UPDATED		TIMESTAMP	Date/time on which the data for this Investment Option was last updated with information from the Financial Service.
NAME	√	CHAR128	The investment option's name (typically identifies the security) as provided by Financial Service.
FI_SUPPLIED_CUSIP		CHAR20	The CUSIP that the Financial Service supplied for this Investment Option. Will only be present if the firm is licensed for CUSIP data and the firm is enabled to deliver it.
FI_SUPPLIED_TICKER		CHAR32	The ticker or other candidate symbol that the Financial Service supplied for this Investment Option.
DATA_AS_OF		DATE	The date which the Investment Option data is 'as of' as reported by the Financial Institution. If the Financial Institution does not report a date, then this field is not provided.
CREATION_DATE		DATE	The date on which this Investment Option was created in the system.
MORNINGSTAR_FUND_ID		CHAR10	This is the Morningstar ID for the investment, which is sometimes called the distinct portfolio level. This field will be populated when the investment option has an associated security that has type MUTUALFUND. It will also be populated for investment options that do not have a security and which are identifiable as a particular mutual fund but with share class unknown.

Holding Lot

Holding Lots are also known as Tax Lots or Position Lots. The aggregation service gathers only open lots from data sources and does so only when the feature is enabled for the firm.

Field	Required	Data Type	Description
ID	√	UID (PKEY)	Unique numeric ID for this Lot.
FP_ID	√	UID (FKEY: FINANCIAL_PROFILE.ID)	ID for the Financial Profile that contains this Holding Lot.
ACCOUNT_ID	√	UID (FKEY: ACCOUNT.ID)	ID for the Account that contains this Holding Lot.
HOLDING_ID		UID (FKEY: HOLDING.ID)	ID for the Holding corresponding to this Holding Lot.
SECURITY_ID		UID (FKEY: SECURITY.ID)	ID for the Security held for this Holding Lot.
NAME		CHAR128	The name of the security of the Holding, a name provided by the Financial Institution.
STATE	√	CHAR12	Will always contain the value: OPEN_LOT.
FI_SUPPLIED_CUSIP		CHAR20	The CUSIP for the security corresponding to this lot as provided by the Financial Institution. CUSIP will only be present if the firm is licensed for CUSIP data and the firm is enabled to deliver it.
FI_SUPPLIED_TICKER		CHAR32	The ticker symbol for the security corresponding to this lot as provided by the Financial Institution.
FI_SUPPLIED_SEDOL		CHAR7	The SEDOL for the security corresponding to this lot as provided by the Financial Institution. Only included when enabled for the firm.
FI_SUPPLIED_ISIN		CHAR12	The ISIN for the security corresponding to this lot as provided by the Financial Institution. Will only be present if the firm is licensed for CUSIP data and the firm is enabled to deliver it.
UNITS	√	NUMBER	Number of units in this lot.
LOT_ID		CHAR32	Lot identifier supplied by the Financial Institution.
SUB_LOT_ID		CHAR32	Identifier supplied by the Financial Institution for a sub-lot (split from an original lot).
COST_BASIS		NUMBER	Total adjusted cost of the units. May be null if reported as 'unknown' at the site.

Holding Lot (Continued)

Field	Required	Data Type	Description
ORIGINAL_COST_BASIS		NUMBER	Total original cost of the units.
COST_PER_SHARE		NUMBER	Adjusted cost per unit.
OPEN_FAIR_MARKET_VALUE		NUMBER	Market value of the shares when the lot was opened or transferred into the account
UNIT_PRICE		NUMBER	Current market price for open lots.
TOTAL_AMOUNT		NUMBER	Current market value for open lots.
PURCHASE_DATE		DATE	The date the lot was originally purchased. Used to determine the holding period of the lot. Not necessarily the same as acquisition date. Lot may be purchased on one day then transferred to another account. Purchase date may be listed as 'unknown' or 'various' on the Financial Institution; if so, this field will be left empty.
TRANSFER_DATE		DATE	Date the lot was transferred in or gifted to the account.
CLOSE_DATE		DATE	Date the lot was closed. Not used because the service aggregates only open lots.
COMMISSIONS_FEES		NUMBER	The fees or commissions paid on the sale of a closed lot
TAXES_WITHHELD		NUMBER	The fees or commissions paid on the sale of a closed lot.
LOSS_DISALLOWED		NUMBER	The amount of loss that is disallowed because of a related wash sale.
HOLDING_PERIOD	√	CHAR20	Can be: SHORT_TERM or LONG_TERM

Holding Lot (Continued)

Field	Required	Data Type	Description
ACQUISITION_METHOD		CHAR20	The method the lot was acquired. May be one of: <ul style="list-style-type: none"> ▪ GIFT ▪ INHERITANCE ▪ OTHER ▪ PURCHASE ▪ SPLIT ▪ TRANSFER
CLOSE_ACCOUNTING_METHOD		CHAR20	Method to be used when closing the lot. One of: <ul style="list-style-type: none"> ▪ AVG - Average cost, single category ▪ DAVG - Average cost, double category ▪ AVGR - Average cost, single category, whole shares sold ▪ FIFO - First in, first out ▪ LIFO - Last in, first out ▪ MAX - Maximize gain. Sell lowest cost shares first ▪ MIN - Minimize gain; sell highest cost shares first ▪ UNKNOWN VSP - Versus; sell a designated quantity from specific lots
COVERED		CHAR20	Whether the security is covered by new cost basis regulations. One of: NOT_COVERED or COVERED.
ADJUSTMENT_INDICATOR		CHAR64	Indication supplied by the Financial Institution of a corporate action, wash or other adjustment on the lot.
COST_SOURCE		CHAR64	Indication supplied by the Financial Institution of the source of the tax lot information.
LAST_UPDATED	√	TIMESTAMP	Date and time when the information in this Holding Lot was last updated from the Financial Institution.

DATACONNECT OPERATIONS

Overview

This section provides a detailed description of all DataConnect Ultra operations.

DataConnect URL

The DataConnect API for V4 consists of a single URL invocation point:

<https://www.byallaccounts.net/dataconnect/WPServlet?RequestType=DataConnectV4>

This URL must be specified exactly (including case). Your request is routed to the appropriate component on ByAllAccounts' servers. You must:

- Use https (http requests are rejected).
- Use a request method type of **POST** (**GET** type requests are rejected).

The data you provide in the **POST**:

- Is the Input Request Document.
- Contains your Login request (with credentials that authenticate you as a valid user of DataConnect).
- Is your data retrieval or data update request.

The DataConnect web server returns to you a DataConnect response file that is compressed using the ZIP compression format.

Protocol

DataConnect complies with the HTTP/V1.1 protocol and requires use of SSL (https). All input request documents must be sent via **POST** over https.

Versioning

This release of DataConnect uses the following version:

<VERSION>Version4.0**</VERSION>**

Compression

DataConnect Response documents are always compressed in ZIP format.

General XML Document Information

All DataConnect Request and Response documents must adhere to the following:

- All XML element names must be in upper case. For example, the element **<USER_GROUP>** is correct, while the element **<user_group>** is incorrect.
- Values of elements are not case-sensitive unless specifically noted.
- Each document must include a DTD reference to a publicly available DataConnect DTD. The DTD to use is defined in subsequent sections. DataConnect validates your request document against the DTD referenced in your document. If the document cannot be validated, an error is returned to you.
- UTF-8, a compressed version of Unicode that uses only a single byte for most common characters, is the character set used for all documents.

Overall Document Structure

Input Request

- The input request contains the following:
 - **DOCTYPE**
 - **DataConnect Version Specification**
 - **Login Request** providing credentials to authenticate the caller
 - **Operation Request.** The operation request can be:
 - A single request
 - A multipart request. A multipart request has mixed synchronous/asynchronous behavior. See [Multipart Requests](#) for more information.

An input request document has the following overall structure:

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE DATACONNECTRQ PUBLIC  
'-//DataConnect DTD//DataConnect//EN'  
'http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/  
DataConnectRQ.dtd'>  
  
<DATACONNECTRQ>  
  <VERSION>Version4.0</VERSION>  
  { One login request }  
  { One operation request }  
</DATACONNECTRQ>
```

Output Response

A response document has the following overall structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DATACONNECTRS PUBLIC
'-//DataConnect DTD//DataConnect//EN'
'http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/
DataConnectRS.dtd'>
<DATACONNECTRS>
  <VERSION>Version4.0</VERSION>
  { One login response }
  { One operation response }
</DATACONNECTRS>
```

Multipart Requests

DataConnect provides a limited capability for the submission of multiple requests in a single input request document. A multipart request is itself a request and wholly contains other requests. The multipart request has mixed synchronous/asynchronous behavior. A running time is allocated to the multipart request. If the request does not complete within that time, a receipt is returned to the caller and the request continues to run. The caller gets the response data either synchronously or as a response to a **Claim Data** operation.

A sample multipart request is as follows:

```
<LOGINRQ>...</LOGINRQ>
<MULTIRQ_A>
  <ACCTADDRQ>
  </ACCTADDRQ>

  <DATAGETRQ>
  </DATAGETRQ>
</MULTIRQ_A>
```

Authentication

The DataConnect user must provide a username and password that is used for authentication. If a user requests an operation for which he or she is not authorized, an error is returned. Please see the [Authorization](#) section for a description of the privileges required to access DataConnect.

A login request must be included as the first request in each DataConnect input request document. Its purpose is to submit credentials for authentication.

Authorization

The type of Login (administrative or non-administrative) and the API privilege level of the Login are used to determine which operations are available to that Login.

Administrative users of the DataConnect API are given access to all users related to the administrative user's firm. Advisors are given access to their own data and to the data of their clients. They are also given the ability to create new clients (Investors) or Assistants (for themselves). If the firm's usage model is Investor-managed, Investors are given full access to their own data. If the firm's usage model is Advisor-managed, the Advisor determines the level of access (if any) that the Investors have to their financial information.

Asynchronous Operations

By nature, some DataConnect operations are lengthy. For these operations, an http connection cannot be maintained for the duration of the operation because it consumes resources unnecessarily and is subject to interruption by a number of factors, including various client and server timeout settings. The following operations are potentially lengthy:

- **Discover Accounts:** Enables the user to retrieve a list of accounts that are available to a particular login at a financial institution.
- **Export Data:** Retrieves accounts, holdings, and transactions in CSV-delimited format.
- **Get Data:** Primarily used to retrieve a large amount of data (e.g., many users, many Transactions, etc.).
- **Test Account:** Used to test the online access credentials of one or more Accounts. Performance is determined by the responsiveness of the Financial Service that ByAllAccounts service is contacting.
- **Test Account Credential:** Used to test whether an Account Credential authenticates successfully. Can only be executed on a single Account Credential.
- **Update Account from FI:** Used to update data for multiple Accounts from the corresponding Financial Service. Performance is determined by the responsiveness of the Financial Service that the ByAllAccounts service is contacting and the quantity of data being downloaded.

The above operations are provided as asynchronous operations. Asynchronous operations are distinguished by an **_A** at the end of the request and response tags (e.g., **<DATAGETRQ>** is synchronous, **<DATAGETRQ_A>** is asynchronous). While only asynchronous forms are available for **Test Account** and **Update Account from FI**, both synchronous and asynchronous forms are available for **Get Data**. The DataConnect client determines which form to use.

For asynchronous operations, the simple request/response communication sequence used by synchronous operations is replaced with the following steps:

1. DataConnect client sends initial request (**<DATAGETRQ_A>**, etc.).
2. DataConnect server reads and parses initial request. If the request is not valid, an error is returned. If the request is valid, an **Operation Started** status along with a receipt (to retrieve the data later) and an expiration timestamp (point after which the data is no longer retained) are returned. A suggested wait time in milliseconds before issuing the **Claim Data** request is also included in the response.
3. DataConnect server begins processing the request.
4. DataConnect client waits for the number of milliseconds specified in the **<CLAIM_WAIT>** and then sends the subsequent request (**<DATACLAIMRQ>**), including the receipt, to determine if the operation is completed. DataConnect server responds with one of the following:

- Invalid/expired receipt
 - Operation in progress
 - Status and data from the operation
5. Once the client receives data, he should call back the DataConnect server with an Acknowledge Data Receipt request. Upon receipt of this acknowledgement, the DataConnect server removes the data from the temporary data store.

DataConnect clients must accommodate the return of the **Operation Started** status and code to the above scheme in those cases.

Once the operation completes, the data resulting from the operation is retained until the expiration time of the receipt. Expiration time is determined on a per operation basis.

In-Session Activation Codes

A financial institution may require an In-Session Activation Code (ISAC) to complete an authentication sequence. ByAllAccounts aggregation service logs into the financial institution on behalf of the user and therefore must replicate the login process. An In-Session Activation Code is a one-time, temporary identification code that is sent to the user through a known contact mechanism such as the user's mobile phone or email address. The user must then enter that code to successfully complete the login process and gain access to his/her accounts.

DataConnect provides support for ISAC interactions, optionally at request of the calling application on the Test Account Credential and Test Account operations. The ISAC interaction protocol is complex and consists of a variable set of steps that are determined by the financial institution. DataConnect uses the USERINPUT aggregate to provide structure for the flow of request for information from the institution to the user and for the providing of user responses to those requests back to the financial institution. The USERINPUT aggregate contains many sub-elements to define the form.

The user is prompted with the USERINPUT aggregate in a response, and they formulate a corresponding USERINPUTRESULTRQ which also uses the USERINPUT aggregate. The User Input Result option <USERINPUTRESULTRQ> allows the caller to provide user input from a request previously issued as part of an "in-session activation code" sequence.

The ISAC protocol is documented in detail with full examples in [In-Session Activation Codes \(ISAC\)](#), page 242.

Sensitive Data

DataConnect provides only the sensitive data needed to implement a User Interface to DataConnect services. Only Administrative users with API Full privilege can retrieve sensitive data for users other than themselves. Passwords (fields with the Date Type **PWDn**) are WRITE-ONLY and are never returned to the DataConnect client.

The following fields are treated as sensitive:

- **Account_Credential.ACCOUNT_LOGIN**
- **Account_Credential.ACCOUNT_PIN** (WRITE-ONLY)
- **Account_Credential.ACCOUNT_NEW_PIN** (WRITE-ONLY)
- **Account_Credential.ACCOUNT_SECOND_PIN** (WRITE-ONLY)
- **Account_Credential.ACCOUNT_LOGIN_2**
- **Account.ACCOUNT_NUMBER**
- **Account.ACCOUNT_NUMBER_2**
- **Financial_Profile.NAME**
- **Holding.FI_SUPPLIED_ACCTNUM**
- **Holding.FI_SUPPLIED_ACCTREG**
- **Login.LOGIN_NAME**
- **Login.LOGIN_PW** (WRITE-ONLY)
- **Login.PASSWORD_HINT**
- **Login.PW_EXPIRE_DATE**
- **Person.FIRST_NAME**
- **Person.MIDDLE_NAME**
- **Person.LAST_NAME**
- **Person.EMAIL_ADDRESS**
- **Person.TAX_ID**
- **Person.PHONE**
- **Person.FIRM_TAG1**

Account.ACCOUNT_NUMBER is sensitive. However, this field is always provided through DataConnect (all facilities) to allow the DataConnect client to correlate the Account with data in back-office systems.

Passwords

Password fields (fields with the Data Type **PWDn**) cannot be retrieved through DataConnect. However, components that want to present a password field to an end user must know if a password value is present or not. Some password fields, such as **Login.LOGIN_PW**, are required, while others, such as **Account_Credential.ACCOUNT_PIN**, are optional.

When setting a password, the password value must be provided in clear text just as any other field. For example, to provide the value for the **Account_Credential.ACCOUNT_PIN** field when creating or modifying an account credential, something similar to the following can be provided:

```
<ACCOUNT_PIN>asdfjkl2</ACCOUNT_PIN>
```

When retrieving passwords, remember that the actual value of the password is never provided. Instead, if there is no password, the entire element is omitted from the result data. Clearly, this is the case only for optional password

fields where no value is present. In all cases where the password has a value, the `<VALUE_PRESENT/>` empty tag is used to indicate this situation:

```
<LOGIN_PW> <VALUE_PRESENT/> </LOGIN_PW>
```

Aggregates

The following aggregates are used in DataConnect:

- **<STATUS>**

The **<STATUS>** aggregate is used to communicate the result status of a requested operation. **<STATUS>** is always returned as part of a response message and contains:

- **<ERRCODE>**: A numeric code that uniquely identifies the error
- **<ERRMSG>**: The text corresponding to **ERRCODE** (Omitted if **<ERRCODE>** indicates the operation was successful)

One or more errors (**ERRCODE+ERRMSG**) can be included in a **<STATUS>** aggregate. Typically, multiple error messages are provided if there are multiple errors in the input request. In most other cases, a single error is provided. In the case where multiple errors are provided, it is safe to treat the first error in the list as the primary error.

- **<ACCOUNT_STATUS>**

The **<ACCOUNT_STATUS>** aggregate is used to provide status information for a **Test** or **Update Account from FI** operation for a single Account. The **<ACCOUNT_STATUS>** aggregate can contain the following:

- **<ID>**: ID of the Account
- **<LAST_UPDATE_ATTEMPT>**: Timestamp of the last attempt to test or update the Account
- **<LAST_UPDATED>**: Timestamp of the last successful **Test** or **Update Account from FI** operation
- **<UPDATE_STATUS_ERRCODE>**: Numeric error code for the last **Test** or **Update Account from FI** operation
- **<UPDATE_STATUS_INFO>**: Textual information describing the result of the last **Test** or **Update Account from FI** operation
- **<FAILEDSQLIST>**: If the operation failed due to a missing or incorrect Security Question and Answer (SQA), then a **FAILEDSQLIST** tag is included. When present, **FAILEDSQLIST** will contain at least one SQA. The SQA will always contain an ID. If the SQA is a new question for which no answer exists, the SQA will have been added to the database as part of the operation.

- **<USER_IDENT>**

The **<USER_IDENT>** aggregate is used to provide identifying information for a user by specifying one identifier field value and its type:

- **<PERSON_ID>**
- **<PERSON_FIRM_TAG1>**
- **<PERSON_LOGIN_NAME>**

The **<USER_IDENT>** identifies a Person, and is also used to indirectly identify the Financial Profile for that Person. The Financial Profile for the Person is the profile to which he has a Profile Access relationship with role **INVESTOR**.

- **<FILIST>**

The **<FILIST>** aggregate is used to group one or more **<FI>** elements.

- **<SECURITYLIST>**

The **<SECURITYLIST>** aggregate is used to group one or more **<SECURITY>** elements.

- **<DISCOVEREDACCOUNTLIST>**

The **<DISCOVEREDACCOUNTLIST>** aggregate is used to group zero or more **<DISCOVEREDACCOUNT>** objects.

- **<DISCOVEREDACCOUNTINFO>**

The **<DISCOVEREDACCOUNTINFO>** aggregate is used inside of a **<DISCOVEREDACCOUNT>** object and always consists of exactly two elements:

- **<LABEL>** indicates the name of the field being identified
- **<VALUE>** indicates the value of the field identified by the label

- **<DISCOVEREDACCOUNT_STATUS>**

The **<DISCOVEREDACCOUNT_STATUS>** aggregate is used to provide status information for a **Discover Accounts** operation for a single Account Credential.

- **<ID>** Account Credential ID specified in the request
- **<DISCOVEREDACCOUNT_STATUS_TIMESTAMP>** Timestamp of the completion of the **Discover Accounts** operation
- **<DISCOVEREDACCOUNT_STATUS_INFO>** Textual information describing the result of the **Discover Accounts** operation.
- **<DISCOVEREDACCOUNT_STATUS_ERRCODE>** Numeric error code for the **Discover Accounts** operation. These are the same as the possible values for **<UPDATE_STATUS_ERRCODE>** (part of

the **<ACCOUNT_STATUS>** aggregate returned by the **Update Account** and **Test Account** operations) listed in APPENDIX D

- **<FAILEDSQLIST>**: If the operation failed due to missing or incorrect the security questions and answers (SQA), then a FAILEDSQLIST tag is included. When present, FAILEDSQLIST will contain at least one SQA. The SQA will always contain an ID. If the SQA is a new question for which no answer exists, the SQA will have been added to the database as part of the operation.

▪ **<ACCOUNT_CREDENTIAL_STATUS>**

The **<ACCOUNT_CREDENTIAL_STATUS>** aggregate is used to provide status information for a **Test Credential** operation for a single account credential.

<ACCOUNT_CREDENTIAL_STATUS> aggregate can contain the following:

- **<ID>**: ID of the Account Credential
- **<LAST_AUTHENTICATION_ATTEMPT>**: Timestamp of the last attempt to test or update the Account Timestamp of the last successful operation
- **<AUTHENTICATION_STATUS_ERRCODE>**: Numeric error code for the last relevant operation
- **<AUTHENTICATION_STATUS_INFO>**: Textual information describing the result of the last relevant operation
- **<FAILEDSQLIST>**: If the operation failed due to missing or incorrect the security questions and answers (SQA), then a FAILEDSQLIST tag is included. When present, FAILEDSQLIST will contain at least one SQA. The SQA will always contain an ID. If the SQA is a new question for which no answer exists, the SQA will have been added to the database as part of the operation.

▪ **<USERINPUT>**

The **<USERINPUT>** aggregate is used to define the two-way form used for handling in-session activation codes (ISAC). The aggregate is entirely made up of a list of one or more **INPUT** aggregates which define the form. ISAC interactions may be included in a **Test Account Credential (ACCTCREDTESTRS_A)** and **Test Account (ACCTTESTR)** operations which use the **USERINPUT** aggregate to define the inputs.

After receiving an **ACCTCREDTESTRS_A** or **ACCTTESTR** with a **USERINPUT** element in the response, an **USERINPUTRESULTRQ** request must be submitted to continue the operation. The **USERINPUTRESULTRQ** echoes the information in the **USERINPUT** block the user has already received, with modifications to reflect the selected values. Refer to [In-Session Activation Codes \(ISAC\)](#), page 242 for details and code examples, as well as information about the **<USERINPUT>** aggregate.

Data Types

In addition to the [Data Types](#) defined in the General Object Information section, the following Data Type is used in values of request or response elements:

Data Type	Description
RECEIPTnn	A receipt for an asynchronous operation. The receipt is alphanumeric and can contain up to nn characters.

Operation Profile

In general, each operation consists of a request/response message pair. The request is what is sent to DataConnect to request that the operation be performed. The response is what DataConnect returns to indicate whether the operation was successfully or not, as well as to return any data related to the request.

Operation Summary

The following tables contain all DataConnect operations. They include the purpose of each operation and its associated Request/Response pair.

General Operations		
Login	Authenticates the caller as a user with sufficient privilege to perform DataConnect functions.	Request: <LOGINRQ> Response: <LOGINRS>
Multipart Request	A container of other requests.	Request: <MULTIRQ_A> Response: <MULTIRS_A>
General Error	General Error is a response-only message that is returned when DataConnect is unable to process the input request document because it is empty, malformed, or otherwise fundamentally incorrect.	Request: N/A Response: <GENERALRS>
Asynchronous Helper Operations		
Claim Data	Retrieves data resulting from a previously requested asynchronous operation.	Request: <DATACLAIMRQ> Response: <DATACLAIMRS>
Acknowledge Data Received	Informs the DataConnect server that the data was successfully retrieved and that DataConnect may release the data.	Request: <DATAACKRQ> Response: <DATAACKRS>
Claim File	Used in conjunction with operations whose final response is a file. For example: Export Data (Asynchronous) operation or Import Data (Asynchronous) to download a ZIP file.	Request: <FILECLAIMRQ> Response: ZIP file containing CSV exports or <FILECLAIMRS>
Retrieval Operations		
Get Data (synchronous)	Retrieves personal profiles and financial information for one or more users.	Request: <DATAGETRQ> Response: <DATAGETR>
Get Data (asynchronous)	Provides the same data retrieval capability as the synchronous Get Data function.	Request: <DATAGETRQ_A> Response: <DATAGETR_A>
Get Financial Service List	Retrieves the list of financial services supported by ByAllAccounts.	Request: <FILISTGETRQ> Response: <FILISTGETRS>
Get Financial Service	Retrieves detailed information for one or more financial services.	Request: <FIGETRQ> Response: <FIGETR>
Export Data (asynchronous)	Retrieves accounts, holdings, transactions, and clients in CSV-delimited format.	Request: <DATAEXPORTRQ_A> Response: <DATAEXPORTRS_A>
Import Data (asynchronous)	Import Investors in CSV-delimited format.	Request: <DATAIMPORTRQ_A> Response: <DATAIMPORTRS_A>
Export Tax Lot Files	Request a list of available tax lot files in user's scope. User can be Advisor, Assistant, or a DC Ultra user.	Request: <TAXLOTFILELISTRQ> Response: <TAXLOTFILELISTRS>

Operation Summary (continued):

Modification Operations		
Add User	Creates a new user.	Request: <USERADDRQ> Response: <USERADDRS>
Modify User	Modifies information for a user.	Request: <USEMODRQ> Response: <USERMODRS>
Unsubscribe User	Unsubscribes a user from the service.	Request: <USERUNSUBSCRIBERQ> Response: <USERUNSUBSCRIBERS>
Modify Financial Profile	Changes information for a Financial Profile.	Request: <PROFMODRQ> Response: <PROFMODRS>
Add Portfolio	Adds a new portfolio for a user.	Request: <PORTADDRQ> Response: <PORTADDRS>
Modify Portfolio	Changes the name of a portfolio.	Request: <PORTMODRQ> Response: <PORTMODRS>
Delete Portfolio	Permanently deletes a portfolio and all other objects (accounts, holdings, and transactions) contained in that portfolio.	Request: <PORTDELQ> Response: <PORTDELRS>
Add Account Credential	Adds a new account credential to a profile.	Request: <ACCTCREDADDRQ> Response: <ACCTCREDADDRS>
Modify Account Credential	Changes information for an account credential.	Request: <ACCTCREDMODRQ> Response: <ACCTCREDMODRS>
Delete Account Credential	Permanently deletes an account credential. Related accounts are left without associated credentials.	Request: <ACCTCREDDELQ> Response: <ACCTCREDDELRS>
Add Account	Adds a new account for a user.	Request: <ACCTADDRQ> Response: <ACCTADDRS>
Modify Account	Changes information for an account.	Request: <ACCTMODRQ> Response: <ACCTMODRS>
Delete Account	Permanently deletes an account and all other objects (holdings, transactions) contained in the account.	Request: <ACCTDELQ> Response: <ACCTDELRS>

Operation Summary (continued):

Aggregation Operations		
Test Account Credential (asynchronous)	Tests whether an Account Credential will authentication successfully.	Request: <ACCTCREDTESTRQ_A> Response: <ACCTCREDTESTRS_A>
Test Account (asynchronous)	Attempts to login in to an account's financial service to verify that the online access credentials stored with the account are correct.	Request: <ACCTTESTRQ_A> Response: <ACCTTESTRS_A>
Update Account from FI (asynchronous)	Gathers data for all accounts for the specified financial services (for a particular user).	Request: <ACCTUPDRQ_A> Response: <ACCTUPDRS_A>
Update Account Tax Lots	Requests that tax lot data be aggregated for a set of accounts.	Request: <ACCTUPDTAXLOTRQ> Response: <ACCTUPDTAXLOTRS>
Discover Accounts (asynchronous)	Retrieves a list of accounts that are available to a particular login at a financial institution.	Request: <ACCTDISCRQ_A> Response: <ACCTDISCRS_A>
User Input Result	Allows the caller to provide user input from a request previously issued as part of an "in-session activation code" sequence.	Request: USERINPUTRESULTRQ Response: USERINPUTRESULTRS
Single Sign-on Operations		
ByAllAccounts Application SSO - Session Authentication Request	Authenticates a user to enable subsequent access to a ByAllAccounts application or component by programmatic means.	Request: SESSIONAUTHRQ Response: SESSIONAUTHRS
ByAllAccounts Application SSO - Session Expire Request	Terminates a session created by SESSIONAUTHRQ.	Request: SESSIONEXPIRERQ Response: SESSIONEXPIRERS
ByAllAccounts Application SAML SSO - Session Authentication Request	Authenticates a user using SAML to enable subsequent access to a ByAllAccounts component by programmatic means.	Request: SAMLAUTHRQ Response: SAMLAUTHRS

General Operations

This section describes the operations of general utility, including authentication and general error response.

The following items are defined for each DataConnect operation (**Note:** Sections identified as optional are omitted if there is no relevant information for that operation.):

- **Purpose:** Why or for what the operation is used
- **Restrictions:** Limitations or notes on the use of the operation
- **Behavior (optional):** For complex operations, explains in detail the behavior of the operation under different circumstances
- **User Notifications (optional):** User notifications resulting from the function (if any)
- **Request:** Form of the function request
- **Response:** Form of the function response
- **Errors:** Status codes that may be returned in the response
- **Sample XML:** Examples of the operation request and response **Note:** Ellipses (...) may be used in Sample XML to indicate the presence of additional information not shown.

Login

Purpose

Authenticates the caller as a user with sufficient privilege to perform DataConnect API functions.

Restrictions

The following restriction applies to the use of this function:

There must be only one **LOGINRQ** in an input request document and it must be the first request in the document.

Request: <LOGINRQ>

The <LOGINRQ> request can contain the following:

Tag	Required	Field	Description
<LOGIN_NAME>	√	LOGIN_NAME	Login name for a user with DataConnect API privileges.
<LOGIN_PW>	√	LOGIN_PW	Password that goes with LOGIN_NAME
<NEW_LOGIN_PW>		NEW_LOGIN_PW	Changes password for LOGIN_NAME to NEW_LOGIN_PW. The current LOGIN_PW must be provided in LOGIN_PW. This option is typically used when LOGIN_PW has expired and must be reset to gain access to DataConnect. If the password change does not succeed, then the LOGINRQ fails and any subsequent operations contained in the request are not attempted.

Response: <LOGINRS>

The <LOGINRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
56003	Error	The login name or password is incorrect.
56011	Error	Caller's account is disabled. Contact Technical Support for assistance.
56019	Error	Caller has been unsubscribed and can no longer log in.
56027	Error	The password has expired.
56803	Error	The new password does not meet minimum length requirements.
56811	Error	The new password cannot be a single repeated character.
56819	Error	The new password cannot be all letters or all numbers.
56827	Error	The new password must be different than your old password.
56835	Error	The password has too many sequential or repeating characters (e.g. AAAA or 1234).
56843	Error	The new password cannot be the same as the login.
56851	Error	The new password must have at least one letter.
67739	Error	Access was not performed from an allowed IP address.

Sample XML

The Login Request is always the first request in an input request document. An additional request to retrieve or update data follows the Login Request.

The following is a sample Login Request:

```
<DATACONNECTRQ>  
  <VERSION>Version4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>joesmith</LOGIN_NAME>  
    <LOGIN_PW>xyz123</LOGIN_PW>  
  </LOGINRQ>
```

...

</DATACONNECTRQ>

The following is a sample Login Response:

<DATACONNECTRS>

<VERSION>VERSION4.0</VERSION>

<LOGINRS>

<STATUS>

<ERRCODE>0</ERRCODE>

<ERRMSG>Success</ERRMSG>

</STATUS>

</LOGINRS>

...

</DATACONNECTRS>

MultiPart Request

Purpose

The MultiPart Request is a container of other requests. It supports only a small set of requests.

Restrictions

The MultiPart Request has the following restrictions:

It may contain only two requests. The first request must be an **Add Account** or **Modify Account** request and the second request must be a **Data Get** request. The **Add Account** and **Modify Account** operations may specify the **DO_UPDATE** option. If they do, the **Data Get** waits for this operation to complete before beginning. See [general information on multipart requests](#).

Response: <MULTIRQ_A>

The <MULTIRQ_A> can contain the following:

Tag	Required	Data Type	Description
<ACCTADDRQ> or <ACCTMODRQ>	√		Add Account or Modify Account operation.
<DATAGETRQ>	√		Data Get operation. If an Account was just added in the first part of the request, the Data Get may include an <ACCOUNT_QUERY> aggregate with an ID of -1 to reference the just-added account.

Response: <MULTIRS_A>

The <MULTIRS_A> response can contain either the combined responses of the submitted requests (if the operations completed) or the following (if the operations are still in progress):

Tag	Required	Data Type	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS> .
<RECEIPT>	√	RECEIPT64	Receipt for the client to inquire at a later time about the result of this operation. Only issued if <STATUS> is successful.
<RECEIPT_EXP>		TIMESTAMP	Expiration time of <RECEIPT>
<CLAIM_WAIT>		NUMBER	Number of milliseconds to wait before attempting to retrieve the results of the MultiPart request via the Claim Data request. Present only if <RECEIPT> is present.

If the operation is complete, the combined responses of the submitted requests are returned:

Tag	Required	Data Type	Description
<STATUS>			See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS> .
<ACCTADDRS> or <ACCTMODRS>			Result of Account Add or Account Modify operation.
<DATAGETRS>			Result of Data Get operation.

Sample XML

The Login Request is always the first request in an input request document. An additional request to retrieve or update data follows the Login Request.

The following is a sample MultiPart Request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>joesmith</LOGIN_NAME>  
    <LOGIN_PW>xyz123</LOGIN_PW>  
  </LOGINRQ>  
  <MULTIRQ_A>  
    <ACCTADDRQ>  
      <DO_UPDATE/>  
      <FP_ID>49221</FP_ID>  
      <AC_ID>123</AC_ID>  
      <NAME>Account 1</NAME>  
      <ACCOUNT_NUMBER>A123J789</ACCOUNT_NUMBER>  
    </ACCTADDRQ>  
    <DATAGETRQ>  
      <GET_DATA_QUERY>
```

```
<ACCOUNT_QUERY>
  <ID>-1</ID>
</ACCOUNT_QUERY>
</GET_DATA_QUERY>
<INCSECURITY/>
<INCAccount/>
<INCHOLDING/>
</DATAGETRQ>
</MULTIRQ_A>
</DATACONNECTRQ>
```

The following is a sample MultiPart Response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <MULTIRS_A>
  <ACCTADDRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <FP_ID>49221</FP_ID>
    <ID>56544</ID>
  </ACCTADDRS>
```

Sample MultiPart Response (continued):

```
<DATAGETRS>
<SECURITYLIST>
  <SECURITY>
    <ID>81</ID>
    <NAME>VANGUARD WINDSOR</NAME>
    <SECTYPE>MUTUALFUND</SECTYPE>
    <TICKER>VWNDX</TICKER>
    <CUSIP>922018106</CUSIP>
  </SECURITY>
</SECURITYLIST>
<FP_DATA>
  <ACCOUNT>
    <FP_ID>49221</FP_ID>
    <ID>56544</ID>
    <NAME>Account 1 </NAME>
    <PORTFOLIO_ID>140541</PORTFOLIO_ID>
    <AC_ID>123</AC_ID>
    <ACCOUNT_NUMBER>A123J789</ACCOUNT_NUMBER>
    <ONLINE_ACCESS_ENABLED>1</ONLINE_ACCESS_ENABLED>
    <LAST_UPDATED>20030621113200</LAST_UPDATED>
    <UPDATE_STATUS_INFO>Update successful on Jun 21, 2003 11:31:38 AM
    </UPDATE_STATUS_INFO>
    <UPDATE_STATUS_ERRCODE>0</UPDATE_STATUS_ERRCODE>
    <CAPTIVE>1</CAPTIVE>
    <DATA_BASIS>TRADE</DATA_BASIS>
    <CREATION_DATE>20191030141719 [-5:EDT]</CREATION_DATE>
    <GATHER_LOTS>0</GATHER_LOTS>
    <EXTERNAL_SERVICE_LEVEL>Transactional</EXTERNAL_SERVICE_LEVEL>
```

</ACCOUNT>

<HOLDING>

<FP_ID>49221</FP_ID>

<ACCOUNT_ID>56544</ACCOUNT_ID>

<ID>430813</ID>

<SECURITY_ID>81</SECURITY_ID>

<UNITS>2358.4900</UNITS>

<COST_BASIS>50000</COST_BASIS>

<MARKET_VALUE>26721.69</MARKET_VALUE>

<UNIT_PRICE>11.33</UNIT_PRICE>

<PRICE_DATA_AS_OF>20030414</PRICE_DATA_AS_OF>

<LAST_UPDATED>20191030141719 [-5:EDT]</LAST_UPDATED>

</HOLDING>

</FP_DATA>

</DATAGETRS>

</MULTIRS_A>

</DATACONNECTRS>

General Error

Purpose

General Error is a response-only message that is returned when DataConnect is unable to process the input request document because it is empty, malformed, or otherwise fundamentally incorrect.

Restrictions

None

Response: <GENERALRS>

The <GENERALRS> can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.

Errors

<GENERALRS> may contain the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
65547	Error	An Internal error occurred
65699	Error	>** Parsing error, line 11 Element "XXXX" does not allow "xxxx" here
65747	Error	The DataConnect version is unsupported or invalid
65819	Error	The request was empty

Sample XML

In response to a malformed request, a response similar to the following may be received:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <GENERALRS>  
    <STATUS>  
      <ERRCODE>65819</ERRCODE>  
      <ERRMSG>The request was empty</ERRMSG>  
    </STATUS>  
  </GENERALRS>  
</DATACONNECTRS>
```

Asynchronous Helper Operations

Used in conjunction with asynchronous operations, the operations presented in this section are used to complete the operation. These helper operations provide generic messages for retrieving data resulting from an asynchronous operation and for releasing that data once the DataConnect client receives it.

Claim Data

Purpose

Retrieves data resulting from a previously requested asynchronous operation.

Restrictions

The following restrictions apply to this operation:

The credentials provided in the **<LOGINRQ>** accompanying this operation and the credential used to originally submit the operation request must be identical. If they are not, an error is returned.

Request: **<DATACLAIMRQ>**

The **<DATACLAIMRQ>** can contain the following:

Tag	Required	Data Type	Description
<RECEIPT>	√	RECEIPT64	Receipt previously issued to the DataConnect client for an asynchronous operation.

Response: **<DATACLAIMRS>**

The following is a response to a **<DATACLAIMRQ>** request.

If the asynchronous operation is not completed, a **<DATACLAIMRS>** response, containing the following, is returned:

Tag	Required	Data Type	Description
<STATUS>	√		See <STATUS> aggregate description.
<RECEIPT>	√	RECEIPT64	Receipt provided in request.
<CLAIM_WAIT>			Provided only if <RECEIPT> is valid. Value is the number of milliseconds to wait before reattempting the Claim Data request.

The status typically indicates that the operation is in progress, or it could report that the receipt expired.

If the asynchronous operation is complete, the data returned is the full response for that operation and is one of the following, and the **<DATACLAIMRS>** response tags are not used:

- **<DATAGETRS_A>**
- **<ACCTTESTRS_A>**
- **<ACCTUPDRS_A>**

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
0	Success	Success
0	Success	Operation started
0	Success	Operation in progress
65547	Error	An Internal error occurred
65570	Warning	The operation did not complete in its allotted time
65579	Error	The receipt has expired

Sample XML

The following is a sample request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  
  <DATACLAIMRQ>  
    <RECEIPT>4456858471129290880</RECEIPT>  
  </DATACLAIMRQ>  
  
</DATACONNECTRQ>
```

The following is a sample response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>

  <DATACLAIMRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Operation in progress </ERRMSG>
    </STATUS>
    <RECEIPT>4456858471129290880</RECEIPT>
    <CLAIM_WAIT>5000</CLAIM_WAIT>
  </DATACLAIMRS>

</DATACONNECTRS>
```

Acknowledge Data Received

Purpose

Informs the DataConnect server that the data was successfully retrieved and that DataConnect may release the data. Upon receiving this request, the DataConnect server removes the data from storage and the receipt is expired regardless of any prior expiration date issued with the receipt.

Restrictions

None.

Behavior

This operation exhibits the following behaviors:

1. If the **<DATAACKRQ>** is submitted while the operation corresponding to the receipt is still in progress, an error is returned. The operation must complete prior to retrieving the data and acknowledging data receipt.
2. If the credentials provided in the **<LOGINRQ>** accompanying this operation and these credential differ from those originally used to submit the operation request, an error is returned.

Request: <DATAACKRQ>

The **<DATAACKRQ>** can contain the following:

Tag	Required	Data Type	Description
<RECEIPT>	√	RECEIPT64	Receipt previously issued to the DataConnect client for this asynchronous operation.

Response: <DATAACKRS>

The **<DATAACKRS>** can contain the following:

Tag	Required	Data Type	Description
<STATUS>	√		See <STATUS> aggregate description.
<RECEIPT>	√	RECEIPT64	Receipt provided in request.

Errors

This operation may return any of the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
0	Success	Success
65547	Error	An Internal error occurred
65579	Error	The receipt has expired

Sample XML

The following is a sample request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>

  <DATAACKRQ>
    <RECEIPT>4456858471129290880</RECEIPT>
  </DATAACKRQ>

</DATACONNECTRQ>
```

The following is a sample response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>

  <DATAACKRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <RECEIPT>4456858471129290880</RECEIPT>
  </DATAACKRS>

</DATACONNECTRS>
```

Claim File

Purpose

The Claim File operation is used in conjunction with the:

- [Export Data \(Asynchronous\)](#) operation
- [Import Data \(Asynchronous\)](#) operation
- [List Available Tax Lots](#) operation

When the operation has completed and indicated that a requested export file is ready to download, a Claim File operation should be submitted to download the ZIP file.

Restrictions

Available to all user types. The Claim File must be performed by the same user who submitted the original Export Data request.

Behavior

This operation exhibits the behaviors described here. This description uses Export Data (Asynchronous) as an example. The resulting ZIP for the Import Data operation is described in [Import Data \(Asynchronous\)](#).

The resulting ZIP files for Export Data (Asynchronous) and Exporting Tax Lots are described in [Response: ZIP file containing CSV exports, or <FILECLAIMRS>](#).

- 1) In normal successful usage a ZIP file of CSVs is produced containing a README.txt and CSV files (listed in the response table on page 91). Whether a particular CSV file is included depends on two factors: if it was requested and if there is data for it.

For example, export_accounts.csv is included if and only INCACCOUNTFILE was included in the request AND at least one account was found. The same is true for export_positions.csv (INCHOLDINGFILE), and for export_transactions.csv (INCTXFILE). It is possible to have no CSVs and just the README.txt file. The README.txt file will always be present, and will include error messages if any unexpected errors occurred when generating any of the files.

Here are examples of README.txt file files. There is always one ZIP file present. In this one, the user requested four CSV files and got four CSV files:

```
2014-02-25 15:52 EST,  Accounts exported to export_accounts.csv
2014-02-25 15:52 EST,  Positions exported to export_positions.csv
2014-02-25 15:52 EST,  Transactions exported to export_transactions.csv
2014-02-25 15:52 EST,  Clients exported to export_clients.csv
```

Here the user requested only accounts (via including INCACCOUNTFILE), but there were none, so the user received no CSV files:

2014-02-25 15:52 EST, There were no Accounts to export

- 2) If no file is found (due to an unexpected error condition), then a ZIP file containing a standard DataConnect XML response with a FILECLAIMRS (or possibly a GENERALRS or LOGINRS) indicating the error condition.

Request: <FILECLAIMRQ>

The <FILECLAIMRQ> request can be formed one of two ways, depending on which of the following was used to generate the files:

- DATAEXPORTRS_A or DATAIMPORTRS_A
- TAXLOTFILELISTRQ

For DATAEXPORTRS_A or DATAIMPORTRS_A - If the file was created using a DATAEXPORTRS_A or DATAIMPORTRS_A operation, the request consists of the RECEIPT associated with the operation which has already been established to be completed.

Tag	Required	Field	Description
<RECEIPT>	√	RECEIPT	The receipt provided during the original DATAEXPORTRS_A or DATAIMPORTRS_A operation.

For TAXLOTFILELISTRQ - If the file was created using a TAXLOTFILELISTRQ operation, use these tags.

Tag	Required	Field	Description
<ID>	√	ID	The unique ID of the file to be claimed.
<FILETYPE>	√	TAXLOT	Claim the tax lot files.

Response: ZIP file containing CSV exports, or <FILECLAIMRS>

The normal response is different depending on which of the following was used to generate the files:

- DATAEXPORTRS_A or DATAIMPORTRS_A
- TAXLOTFILELISTRQ

For DATAEXPORTRS_A or DATAIMPORTRS_A, the normal response to a FILECLAIMRS will be a ZIP file. For Export Data (Asynchronous), the ZIP file may contain the contents show below. The resulting ZIP for Import Data (Asynchronous) is described in [Import Data \(Asynchronous\)](#).

Filename	Required	Description
README.txt	√	A brief log file
export_accounts.csv		A list of accounts, included only if 1) INCACCOUNTFILE tag was included in original request, and 2) at least one account to export was found
export_positions.csv		A list of holdings, included only if 1) INCHOLDINGFILE tag was included in original request, and 2) at least one holding to export was found
export_transactions.csv		A list of transactions, included only if 1) INCTXFILE tag was included in original request, and 2) at least one transaction to export was found
export_clients.csv		A list of clients, included only if 1) INCCLIENTFILE tag was included in the original request and 2) at least one client to export was found
export_investmentoptions.csv		A list of the investment options, included only if 1) INCINVOPTFILE tag was included in original request, and 2) at least one investment option to export was found.

For details about these files, refer to the *AccountView and DataConnect Export* guide posted at http://www.byallaccounts.net/manuals/accountview/BAA_Export.PDF.

For TAXLOTFILELISTRQ, the normal response to a FILECLAIMRS will be a ZIP file, with contents as shown below.

Filename	Required	Description
<filename>.CSV		The tax lot file requested

For details about the possible files, refer to [Appendix F: Tax Lot Files](#).

The **<FILECLAIMRS>** XML response will contain the following (error cases only):

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
0	Success	Success
65547	Error	An Internal error occurred
65579	Error	The receipt has expired
65875	Error	The FILETYPE is invalid
65947	Error	Caller is not authorized to make this request
65963	Error	User type not supported for this operation.
67355	Error	Tax lot gathering is not enabled for this firm.
67499	Error	Bulk Export files not generated
67859	Error	Export file could not be found

Sample XML

The following is a sample **<FILECLAIMQ>** request for DATAEXPORTRS_A or DATAIMPORTRS_A

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>JoeSmith1</LOGIN_NAME>  
    <LOGIN_PW>JoeSmith1</LOGIN_PW>  
  </LOGINRQ>  
  <FILECLAIMRQ>  
    <RECEIPT>4456858471129290880</RECEIPT>  
  </FILECLAIMRQ>  
</DATACONNECTRQ>
```

The operation should yield a ZIP file containing a README.txt file and possibly one or more CSV files.

The following is a sample **<FILECLAIMQ>** request when claiming the file for TAXLOTFILELISTRQ.

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>AdvisorChris</LOGIN_NAME>  
    <LOGIN_PW>AdvisorChrisPW</LOGIN_PW>  
  </LOGINRQ>  
  <FILECLAIMRQ>  
    <ID>1</ID>  
    <FILETYPE>TAXLOT</FILETYPE>  
  </FILECLAIMRQ>  
</DATACONNECTRQ>
```

The operation should yield a ZIP file containing a CSV file.

The following is a sample **<FILECLAIMQ>** response showing an error when trying to claim the file for TAXLOTFILELISTRQ.

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>  
    </STATUS>  
  </LOGINRS>  
  <FILECLAIMRS>  
    <STATUS>  
      <ERRCODE>65963</ERRCODE>  
      <ERRMSG>User type not supported for this operation.</ERRMSG>  
    </STATUS>  
  </FILECLAIMRS>  
</DATACONNECTRS>
```

Retrieval Operations

This section defines operations for retrieving data from DataConnect.

Get Data (Synchronous)

Purpose

Retrieves personal profiles and financial information for one or more users.

Restrictions

None.

Behavior

This operation exhibits the following behaviors:

1. Administrative users are not returned in the response data.
2. The response data is structured so that Securities, Financial Services, and Users are delivered in their own separate aggregate at the beginning of the document. Subsequent data for each User refers to those shared Securities and Financial Services. Data returned for each of these object types is determined by which Users are included in the response set. Specifying **<INCHOLDING/>** or **<INCTX>** will not affect the Securities included in the **<SECURITYLIST>**.

Request: **<DATAGETRQ>**

The request consists of the following two primary items:

- Type of information to include in the response. Choose from any combination of the following: FI (Financial Institution or Financial Service), Security, User (Person, Login, and Profile Access), Financial Profile, Portfolio, Account, Account Credential, Holding, Transaction. Designate the type of information to include using the **<INCxxx>** tags.
- Data to include in the response. You can specify data retrieval criteria for a single type of object per request and can choose from one of the following object types: User, Financial Profile, Portfolio, or Account. If Transactions are to be included, you can specify a date range or ID range to constrain the set of Transactions returned.

The type of object you use to constrain the request affects how data for other object types are retrieved. For example, if you retrieve data for a single Account but elect to receive Security and Portfolio information, then only the Portfolio that contains that Account and the Securities for Holdings in that Account are returned to you.

Designate the data to include in the response using the **<GET_DATA_QUERY>** element.

The <DATAGETRQ> request can contain the following:

Tag	Required	Field	Description
<GET_DATA_QUERY>			Note: See <GET_DATA_QUERY> aggregate description. If no GET_DATA_QUERY is provided, all data available to the caller is returned, constrained only by any transaction range specified in <INCTX> .
<GETOPTIONS>			Options for data retrieval. See the <GETOPTIONS> Aggregate description following.
<INCSECURITY>			See the <INCSECURITY> Aggregate description following. Provides an option for including security closing prices. If present, specifies that Security information should be included in the response. If absent, Securities are not included in the response.
<INCFI/>			Empty tag: If present, specifies that Financial Service information should be included in the response. If absent, Financial Services are not included in the response.
<INCUSER/>			Empty tag: If present, specifies that Person, Login, and Profile Access information should be included in the response. If absent, this information is not included in the response.
<INCFP/>			Empty tag: If present, specifies that Financial Profile objects should be included in the response. If absent, Financial Profiles are not included in the response.
<INCPORFOLIO/>			Empty tag: If present, specifies that Portfolios should be included in the response. If absent, Portfolios are not included in the response.
<INCACCTCRED/>			Empty tag: If present, specifies that Account Credentials and their SQAs should be included in the response. If absent, Account Credentials and SQAs are not included in the response.
<INCAACCOUNT/>			Empty tag. If present, specifies that Accounts should be included in the response. If absent, Accounts are not included in the response.
<INCHOLDING>			If present, specifies that holdings should be included in the response. The aggregator provides for inclusion of sold-off holdings. Note: See the <INCHOLDING> Aggregate description.

<INCHOLDINGLOT/>			Empty tag: If present, specifies that holding lots should be included in the response.
<INCTX>			If present, specifies that Transactions should be included in the response. If absent, Transactions are not included in the response. Note: See <INCTX> aggregate description.
<INCINVOPT>			Empty tag: If present, specifies that Investment Options should be included in the response.

<GET_DATA_QUERY> Aggregate

The **<GET_DATA_QUERY>** aggregate identifies the objects to be retrieved. If omitted, all objects available to the caller are assumed. This is combined with the specification of which type of information to include in the response (see **<INCxxx>** elements in the prior table). You must choose which type of object you will use to drive your query. Then, you may specify search criteria for one or more objects of that type. The following object types are available (see corresponding query components in the table below):

- User
- Financial Profile
- Portfolio
- Account

Data included in the response is filtered according to the driving objects. So, if you identify a single Portfolio to be retrieved using **<PORTFOLIO_QUERY>**, and you specify the **<INCACCOUNTS/>** element only, your response data includes details for all the Accounts that are members of the specified Portfolio. Similarly, going up the data 'tree', if you specify **<INCUSER/>** instead of **<INCACCOUNTS/>**, the information for the user who 'owns' the Portfolio (i.e., the Portfolio contained in the Financial Profile for that INVESTOR user) is included in the response. The broadest filter you can specify is by User.

<GET_DATA_QUERY> can contain one of the following:

Tag	Required	Field	Description
<USER_QUERY>			Identifies the Users for which data should be retrieved. See USER_QUERY aggregate .
<FINANCIAL_PROFILE_QUERY>		Financial_Profile.ID	Identifies the Financial Profiles for which data should be retrieved. More than one <FINANCIAL_PROFILE> can be included.
<PORTFOLIO_QUERY>		Portfolio.ID	Identifies the Portfolios for which data should be retrieved. More than one <PORTFOLIO> can be included.
<ACCOUNT_QUERY>		Account.ID	Identifies the Accounts for which data should be retrieved. More than one <ACCOUNT> can be included.

Notes:

DataGet can be included in a multipart request where the first part of the request is a data update operation (e.g., **Add, Modify, Delete**). When **DataGet** is used in such cases and the initial request is an **Add**, the special value -1 can be used as an **ID** in any of the above <XXX_QUERY> elements to refer to the object created by the first part of the request.

<USER_QUERY> Aggregate

Identifies the set of users to return in the result data. This aggregate contains one of the following:

- **<USER_IDENT>**: Identifies a particular Person for whom data should be retrieved. More than one **<USER_IDENT>** can be given in the **<USER_QUERY>**.
- **<PERSON_ROLE>**: Gives a **ROLE** value for a Person: INVESTOR, ADVISOR, ASSISTANT, or CONSULTANT. Use this to retrieve only certain types of users (e.g., Advisors). If not specified, users of all types (INVESTOR, ADVISOR, ASSISTANT, CONSULTANT) are returned. When an INVESTOR type user is requested, the system-created Unassigned Investor(s) may be returned.
- **<HAVING_ADVISOR>**: Contains a **<USER_IDENT>** for an Advisor. Use this to retrieve data for all Persons who are served by this Advisor. Provide an empty **<HAVING_ADVISOR/>** tag to retrieve persons who are not assigned to an Advisor.
- **<USER_SEARCH>**: Allows for a case-insensitive containment-based search using at least one of **FIRST_NAME** and/or **MIDDLE_NAME** and/or **LAST_NAME**. If multiple search fields are used, the results will include only users which match all of the search criteria. Optionally the **USER_SEARCH** may also include one or more **ROLE** values for a Person: INVESTOR, ADVISOR, ASSISTANT, or CONSULTANT; the user search will be restricted to users which match the given **ROLE(s)**.

<FINANCIAL_PROFILE_QUERY>, <PORTFOLIO_QUERY>, <ACCOUNT_QUERY> Aggregates

Each of these three aggregates can contain one or more **<ID>s** that identify objects of that type to include in the result data.

A **FINANCIAL_PROFILE_QUERY** identifies the **ID(s)** of the target profiles:

```
<FINANCIAL_PROFILE_QUERY>  
<ID>145</ID>  
<ID>553</ID>  
</FINANCIAL_PROFILE_QUERY>
```

A **PORTFOLIO_QUERY** identifies a single Financial Profile and one or more Portfolios from that Financial Profile to include. The following example requests that data for the Portfolios with **ID** 123 or 125 be included in the result data:

```
<PORTFOLIO_QUERY>  
<FP_ID>45226</FP_ID>  
<ID>123</ID>  
<ID>125</ID>  
</PORTFOLIO_QUERY>
```

The **ACCOUNT_QUERY** is very similar to the **PORTFOLIO_QUERY**. It identifies a single Financial Profile and one or more Accounts from that Financial Profile that should be included:

```
<ACCOUNT_QUERY>
<FP_ID>45226</FP_ID>
<ID>123</ID>
<ID>125</ID>
</ACCOUNT_QUERY>
```

The query can reference an Account object added in the first part of a multipart request. To do so, use the value -1 in the **ID** field to identify the Account:

```
<ACCOUNT_QUERY>
<FP_ID>45226</FP_ID>
<ID>-1</ID>
</ACCOUNT_QUERY>
```

<GETOPTIONS> Aggregate

The <GETOPTIONS> aggregate allows the user to specify that only data for captive or non-captive counts is to be included in the response. This tag is optional. GETOPTIONS contains one of the following:

- One of the following to control inclusion of captive or non-captive accounts in the response:
 - <CAPTIVEONLY/> - directs that accounts, account credentials, holdings, transactions, and Fls should only be returned in the response if the account (itself or as the related account for account credentials, holdings, transactions, and Fls) is captive (CAPTIVE field is true).
 - <NONCAPTIVEONLY/> - directs that accounts, account credentials, holdings, transactions, and Fls should only be returned in the response if the account (itself or as the related account for account credentials, holdings, transactions, and Fls) is non-captive (CAPTIVE field is false).
- <INCMULTICURR/> - causes multi-currency information to be included for any holdings and transactions in the response

<INCSECURITY> Aggregate

The <INCSECURITY> aggregate is used to request that securities be included in the DATAGET response. INCSECURITY can take one of the following forms:

- Can be an empty tag, <INCSECURITY/>
- Can include a subordinate tag <INCSECDETAIL/> to request that security prices be delivered in the security data:
<INCSECURITY>

<INCSECDETAIL/>

</INCSECURITY>

- Can include a subordinate tag <INCSECAC/> to request that ASSET_CLASS and ASSET_SUBCLASS fields be delivered in the security data:

<INCSECURITY>

<INCSECAC/>

</INCSECURITY>

- Can include a subordinate tag <INCEXTRASECINFO/> to include additional Morningstar security data as applicable if the Firm is licensed through Morningstar ByAllAccounts to receive the data:

<INCSECURITY>

< INCEXTRASECINFO />

</INCSECURITY>

<INCHOLDING> Aggregate

The <INCHOLDING> aggregate is used to request that holdings be included in the DATAGET response. A holding is either *active* or *sold-off*. Active holdings are those that the Financial Institution reports as present in the account. Sold-off holdings are those that were once present in an account but that the Financial Institution no longer reports as present due to sale, transfer, exchange, or otherwise removal of the holdings from the account. Sold-off holdings are identified by the presence of a DELETED_ON tag in the holding element that provides the date on which the holding was removed from the account.

The <INCHOLDING> can have one of the following forms:

- Can be the empty tag <INCHOLDING/>. This causes only active holdings to be included in the response.
- Can also include a sub-tag <INCHOLDINGSO> that controls the inclusion of sold-off holdings in the response. Note that active holdings are always included when the INCHOLDING tag is specified regardless of the use of the subordinate INCHOLDINGSO tag. The INCHOLDINGSO tag can take one of the following forms:

- can be an empty tag, <INCHOLDINGSO/>, would be used as follows:

<INCHOLDING>

<INCHOLDINGSO/>

</INCHOLDING>

This specifies that sold-off holdings should be included in the response. If the request does not specify the INCTX tag then all sold-off holdings are included in the response. Caution should be taken when using this option as this the number of sold-off holdings could be very large. If the request includes the INCTX tag then only sold-off holdings referenced by transactions included in the response are included.

- can include a date restriction:
 <INCHOLDING>
 <INCHOLDINGSO>20040101</INCHOLDINGSO>
 </INCHOLDING>

The date restricts the sold-off holdings included in the response. Only those holdings with a DELETED_ON date on or after the specified date will be included in the response. When this form of INCHOLDINGSO is used no other criteria (such as the presence of an INCTX tag in the DATAGETRQ) are used to restrict the sold-off holdings returned in the response.

- Can also include a subordinate tag <INCHOLDAC/> to request that ASSET_CLASS and ASSET_SUBCLASS fields be delivered in the holding data:
 <INCHOLDING>
 <INCHOLDAC/>
 </INCHOLDING>

<INCTX> Aggregate

The <INCTX> aggregate:

- Can be an empty tag (in which case, **all Transactions** are returned).

OR

- Can contain one of the following to restrict the set of Transactions returned:

Tag	Required	Field	Description
<TX_START_DATE>	√	Transaction.EXECUTION_DATE	See <INCTX> aggregate notes below.
<TX_END_DATE>		Transaction.EXECUTION_DATE	See <INCTX> aggregate notes below.

OR

Tag	Required	Field	Description
<TX_START_ID>	√	Transaction.ID	See <INCTX> aggregate notes below.
<TX_END_ID>		Transaction.ID	See <INCTX> aggregate notes below.

OR

Tag	Required	Field	Description
<TX_SETTLE_START_DATE>	√	Transaction.SETTLEMENT_DATE	See <INCTX> aggregate notes below.
<TX_SETTLE_END_DATE>		Transaction.SETTLEMENT_DATE	See <INCTX> aggregate notes below.

Two additional options are available in <INCTX>. The **<INCEXINFO/>** option in the <INCTX> aggregate causes the inclusion of additional fields that contain descriptive information about the transaction from the custodian: FI_SUPPLIED_DESCRIPTION, FI_SUPPLIED_TX_TYPE, FI_SUPPLIED_TX_TYPE2, and FI_SUPPLIED_TX_TYPE_CODE. The <INCHOLDINGLOTINFO> option in the <INCTX> aggregate is an empty tag which if present, specifies that holding lot related information for the transactions should be included in the response.

<INCTX> aggregate notes:

- **<TX_START_DATE>** and (optionally) **<TX_END_DATE>** are used to constrain the set of Transactions returned by **EXECUTION_DATE**:
 - The date range is applied to the Transaction’s **EXECUTION_DATE** field.
 - **YYYYMMDD**: The format used to identify the date, where **YYYY** is a four-digit year such as 2003, **MM** is a two-digit month identifier ranging from 01 (January) to 12 (December), and **DD** is a two-digit day identifier ranging from 01 to 31.
 - If there is a start date but no end date, the end date is set to the day of the API call.
 - The start date must be before or the same as the end date.
 - The date range is inclusive of start date and exclusive of end date. If a start date is specified, then all Transactions with execution dates on or after that date are included. If an end date is also specified, then only Transactions with an execution date before the given end date are included.

- **<TX_START_ID>** and (optionally) **<TX_END_ID>** are used to constrain the set of Transactions returned by **Transaction.ID**:
 - **NNN**: A Transaction ID (see **ID** field in the [Transaction section](#) and see UID Data Type description in DataConnect [Data Types section](#)) - a positive integer from 0 to 9999999999999999 (18 digits). No punctuation (such as comma “,” separator or decimal point) should be used in this ID value.
 - If there is start Transaction number but no end Transaction number, then all Transactions with an ID equal to or greater than the ID specified in **<TX_START_ID>** are returned.
 - The start ID must be less than or equal to the value of the end ID.

- The ID range is inclusive of start ID and exclusive of end ID. If a start ID is specified, then all Transactions with IDs equal to or greater than the start ID included. If an end ID is also specified, then only Transactions with IDs less than the end ID are included.
- **<TX_SETTLE_START_DATE>** and (optionally) **<TX_SETTLE_END_DATE>** are used to constrain the set of Transactions returned by:
 - The date range is applied to the Transaction’s **SETTLEMENT_DATE** field.
 - **YYYYMMDD**: The format used to identify the date, where **YYYY** is a four-digit year such as 2003, **MM** is a two-digit month identifier ranging from 01 (January) to 12 (December), and **DD** is a two-digit day identifier ranging from 01 to 31.
 - If there is a start date but no end date, the end date is set to the day of the API call.
 - The start date must be before or the same as the end date.
 - The date range is inclusive of start date and exclusive of end date. If a start date is specified, then all Transactions with settlement dates on or after that date are included. If an end date is also specified, then only Transactions with a settlement date before the given end date are included.

Response: <DATAGETRS>

The **<DATAGETRS>** response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate .
<SECURITYLIST>		SECURITY.*	See <SECURITYLIST> aggregate . List contains one or more Securities. Includes Securities referenced by Users included in the response.
<FILIST>		Financial_Service.ID, Financial_Service.NAME, Financial_Service.SERVICE_CATEGORY	See <FILIST> aggregate . List contains one or more Financial Services. Includes services referenced by the Users in this response.
<USERLIST>		Person.*, Login.*, Profile_Access.*	See <USER> aggregate . List contains one or more users. Only Profile Access objects related to the target Financial Profiles are included.
<FP_DATA>			See <FP_DATA> aggregate . Financial Profile data for the Users selected.

* = indicates **All** fields

The <FP_DATA> aggregate contains the financial data for a single Investor. It can contain the following:

Tag	Required	Field	Description
<FINANCIAL_PROFILE>	√	Financial_Profile.*	See Financial Profile object data definition.
<PORTFOLIO>		Portfolio.*	Can have more than one. See Portfolio object data definition.
<ACCOUNT_CREDENTIAL>		Account Credential.*	Can have more than one. See Account Credential object data definition.
<ACCOUNT>		Account.*	Can have more than one. See Account object data definition.
<HOLDING>		Holding.*	Can have more than one. See Holding object data definition.
<HOLDING_LOT>		Holding_Lot*	Can have more than one. See Holding Lot object data definition.
<TRANSACTION>		Transaction.*	Can have more than one. See Transaction object data definition.
<INVESTMENT_OPTION>		Investment_Option *	Can have more than one. See Investment Option object data definition.

* = indicates **All** fields

The <USER> aggregate contains the personal and access information for a single user. It can contain the following:

Tag	Required	Field	Description
<PERSON>	√	Person.*	See Person object data definition.
<LOGIN>		Login.*	See Login object data definition.
<PROFILE_ACCESS>		Profile_Access.*	Can have more than one if Person is an Advisor or Assistant.

Errors

This operation may return any of the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
65699	Error	The request is invalid or formatted incorrectly
65707	Error	The date is invalid or formatted incorrectly
65715	Error	The transaction ID range is invalid
65723	Error	The transaction date range is invalid
65731	Error	The transaction start date is invalid
65739	Error	The transaction end date is invalid
65939	Error	The requested user was not found
65947	Error	Caller is not authorized to make this request
66099	Error	The <object name> could not be found
66107	Error	The requested user was not found
66115	Error	The requested user has been unsubscribed. No further operations can be performed on the user
67635	Error	At least one of first name, middle name, or last name must be provided.
67643	Error	The user type must be one of INVESTOR, ADVISOR, ASSISTANT, or CONSULTANT

Sample XML

The following is a sample **<DATAGETRQ>** request that gets Financial Services, Portfolios, Accounts, and user information for the User with the person ID of 8000.

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>.... </LOGINRQ>  
  <DATAGETRQ>  
    <GET_DATA_QUERY>  
      <USER_QUERY>  
        <USER_IDENT>  
          <PERSON_ID>8000</PERSON_ID>  
        </USER_IDENT>  
      </USER_QUERY>
```

```
</GET_DATA_QUERY>
<INCUSER/>
<INCFP/>
<INCACTCRED/>
<INCAccount/>
<INCHOLDING/>
<INCTX/>
</DATAGETRQ>
</DATACONNECTRQ>
```

The following is a sample of a corresponding **<DATAGETRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <DATAGETRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <USERLIST>
      <USER>
        <PERSON>
          <ID>8000</ID>
          <FIRM_TAG1>FirmTag1</FIRM_TAG1>
          <ROLE>INVESTOR</ROLE>
          <FIRST_NAME>Bob</FIRST_NAME>
```

```
<MIDDLE_NAME>Joseph</MIDDLE_NAME>
<LAST_NAME>Smith</LAST_NAME>
<EMAIL_ADDRESS>BAAQA_EMAIL9@YAHOO.COM</EMAIL_ADDRESS>
<CREATION_DATE>20030826</CREATION_DATE>
<IS_SSO>1</IS_SSO>
</PERSON>
<LOGIN>
  <PERSON_ID>8000</PERSON_ID>
  <LOGIN_NAME>BOBSMITH</LOGIN_NAME>
  <LOGIN_PW>
  <VALUE_PRESENT />
</LOGIN_PW>
<PROFILE_ACCESS>
  <PERSON_ID>8000</PERSON_ID>
  <PROFILE_ID>11000</PROFILE_ID>
  <ROLE>INVESTOR</ROLE>
  <ACCESS>READWRITE</ACCESS>
</PROFILE_ACCESS>
</USER>
</USERLIST>
<FP_DATA>
  <FINANCIAL_PROFILE>
    <ID>11000</ID>
    <NAME>Bob Joseph Smith</NAME>
    <CREATION_DATE>20090930</CREATION_DATE>
  </FINANCIAL_PROFILE>
</FP_DATA>
</DATAGETRS>
</DATACONNECTRS>
```

Sample XML

The following is a sample **<DATAGETRQ>** request that gets Financial Services, Portfolios, Accounts, and user information for the Users whose first name contains Bob and whose ROLE is INVESTOR:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ> ... </LOGINRQ>
  <DATAGETRQ>
    <GET_DATA_QUERY>
      <USER_QUERY>
        <USER_SEARCH>
          <FIRST_NAME>Bob</FIRST_NAME>
          <PERSON_ROLE>INVESTOR</PERSON_ROLE>
        </USER_SEARCH>
      </USER_QUERY>
    </GET_DATA_QUERY>
    <INCUSER/>
    <INCFP/>
    <INCACCTCRED/>
    <INCACCOUNT/>
    <INCHOLDING/>
    <INCTX/>
  </DATAGETRQ>
</DATACONNECTRQ>
```

The following is a sample of a corresponding **<DATAGETRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
```

```
<ERRCODE>0</ERRCODE>
  <ERRMSG>Success</ERRMSG>
</STATUS>
</LOGINRS>
<DATAGETRS>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Success</ERRMSG>
  </STATUS>
  <USERLIST>
    <USER>
      <PERSON>
        <ID>8000</ID>
        <FIRM_TAG1>FirmTag1</FIRM_TAG1>
        <ROLE>INVESTOR</ROLE>
        <FIRST_NAME>Bob</FIRST_NAME>
        <MIDDLE_NAME>Joseph</MIDDLE_NAME>
        <LAST_NAME>Smith</LAST_NAME>
        <EMAIL_ADDRESS>BAAQA_EMAIL9@YAHOO.COM</EMAIL_ADDRESS>
        <CREATION_DATE>20030826</CREATION_DATE>
        <IS_SSO>1</IS_SSO>
      </PERSON>
      <LOGIN>
        <PERSON_ID>8000</PERSON_ID>
        <LOGIN_NAME>BOBSMITH</LOGIN_NAME>
        <LOGIN_PW>
          <VALUE_PRESENT />
        </LOGIN_PW>
      </LOGIN>
      <PROFILE_ACCESS>
        <PERSON_ID>8000</PERSON_ID>
        <PROFILE_ID>11000</PROFILE_ID>
```

```
<ROLE>INVESTOR</ROLE>
  <ACCESS>READ</ACCESS>
</PROFILE_ACCESS>
</USER>
</USERLIST>
<FP_DATA>
  <FINANCIAL_PROFILE>
    <ID>8000</ID>
    <NAME>Bob J Smith</NAME>
    <CREATION_DATE>20020613</CREATION_DATE>
  </FINANCIAL_PROFILE>
</FP_DATA>
</DATAGETRS>
</DATACONNECTRS>
```

Get Data (Asynchronous)

Purpose

This asynchronous function provides the same data retrieval capability as the **Synchronous Get Data** function. Please refer to the description of that function for details on data retrieval. Only additional tags relating to the asynchronous nature of the function are presented in this section.

Restrictions

1. This function is asynchronous. Please refer to Asynchronous Operations section for details of using asynchronous functions and to the Asynchronous Helper Operations section for details of related operations.
2. DataConnect clients use the <DATACLAIMRQ> request to retrieve data resulting from an Asynchronous Get Data operation.
3. DataConnect clients should send a <DATAACKRQ> request to tell the DataConnect server that data from a prior asynchronous operation can be released. If this request is not sent, DataConnect retains the data until it expires.

Request: <DATAGETRQ_A>

Identical to <DATAGETRQ> but the operation is processed asynchronously. Please refer to <DATAGETRQ> for details.

Response: <DATAGETRS_A>

The <DATAGETRS_A> contains the following:

Tag	Required	Data Type	Description
<STATUS>	√		See <STATUS> aggregate description.
<RECEIPT>		RECEIPT64	Receipt for the client to inquire later about the result of this operation. Only issued if <STATUS> is successful.
<RECEIPT_EXP>		TIMESTAMP	Expiration time of <RECEIPT>, including time zone. Only present if <RECEIPT> is present.
<CLAIM_WAIT>		NUMBER	Number of milliseconds to wait before attempting to retrieve the results of the Get Data operation via the Claim Data request. Only present if <RECEIPT> is present.

This response provides the receipt needed to claim the results of the **Data Get** operation later. A <DATACLAIMRQ> must be subsequently submitted to check on the status of the **Data Get** operation and to retrieve the final results of the operation. Once the operation is completed, a <DATAGETRS> response containing the requested data is received. Please see the Response section of the **Get Data (Synchronous)** operation (<DATAGETRQ>) for the details of the data returned.

Errors

Please refer to the Errors section of the **Get Data (Synchronous)** operation (<DATAGETRQ>).

Sample XML

See the sample code provided in the **Get Data (Synchronous)** section.

Get Financial Service List

Purpose

Retrieves the list of Financial Services supported by ByAllAccounts. Only identifying information for the Financial Service is returned (**ID**, **NAME**, and **SERVICE_CATEGORY**).

Note: See the **Get Financial Service** operation for more detailed information about the Financial Services supported by ByAllAccounts. This list can change from day to day as support for new services is added. Therefore, it is recommended that the information should not be cached for more than 24 hours.

Restrictions

None.

Request: <FILISTGETRQ>

The <FILISTGETRQ/> is an empty tag (contains no data values or other elements).

Response: <FILISTGETRS>

The <FILISTGETRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<FILIST>			See <FILIST> aggregate description. The following information is returned for each FI: Financial_Service.ID , Financial_Service.NAME , Financial_Service.SERVICE_CATEGORY .

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
66099	Error	The requested data could not be found

Sample XML

The following is a sample **<FILISTGETRQ>** request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  <FILISTGETRQ/>  
</DATACONNECTRQ>
```

The following is a sample corresponding **<FILISTGETRS>** response:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>...</LOGINRS>  
  <FILISTGETRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>  
    </STATUS>  
    <FILIST>  
      <FI>  
        <ID>98</ID>  
        <NAME>TD Waterhouse Investment Services</NAME>  
        <SERVICE_CATEGORY>Investment
```

```
        </SERVICE_CATEGORY>
    </FI>

    <FI>
        <ID>200</ID>
        <NAME>Western State Bank</NAME>
        <SERVICE_CATEGORY>Banking
        </SERVICE_CATEGORY>
    </FI>

</FILIST>
</FILISTGETRS>
</DATACONNECTRS>
```

Get Financial Service

Purpose

Retrieves detailed information for one or more Financial Services. This information includes instructions for enabling online access to the service and terminology used by the service to name pieces of credential information required for access, as well as any special instructions for using this service. This information can change from day to day as financial institution may revise their requirements. Therefore, avoid caching this information for more than 24 hours.

Restrictions

None.

Request: <FIGETRQ>

The <FIGETRQ> can contain the following:

Tag	Required	Field	Description
<ID> or <ALL/>	√	Financial_Service.ID	One or more Financial Service IDs for which to retrieve information or <ALL/> to request that all Financial Services be retrieved.

Response: <FIGETRS>

The <FIGETRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<FILIST>			See <FILIST> aggregate description. This list contains full Financial Service objects (all fields). Note: See the FI (Financial Institution or Financial Service) object definition for details.

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
65555	Error	Either one or more IDs of financial services or ALL must be specified
65779	Error	The <field name> is outside the valid range valid range of 0 to 999999999999999999
66099	Error	The requested data could not be found
66811	Error	Financial service IDs and ALL cannot be specified in the same request
67938	Warning	Response contains partial data set

Sample XML

The following is a sample **<FIGETRQ>** request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  
  <FIGETRQ>  
    <ID>2000</ID>  
    <ID>2001</ID>  
  </FIGETRQ>  
</DATACONNECTRQ>
```

The following is a corresponding sample **<FIGETRS>** response:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>
```

```
<ERRMSG>Success</ERRMSG>
</STATUS>
</LOGINRS>
<FIGETRS>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Success</ERRMSG>
  </STATUS>
  <FILIST>
  <FI>
    <ID>430</ID>
  <NAME>Quick & Reilly, Inc</NAME>
  <SERVICE_CATEGORY>Investment</SERVICE_CATEGORY>
  <USERLOGIN_URL>https://www.quickandreilly.com/cgi-bin/login
  </USERLOGIN_URL>
  <REQ_SQA>0</REQ_SQA>
  <ACCESS_INSTRUCTIONS>
    <![CDATA[Accessing your Quick and Reilly account online requires two steps: registering for online
    access and then "linking" your existing accounts for online access. To register, complete the <A
    HREF="https://www.quickandreilly.com/cgi-bin/UserReg?">online Customer Registration form.</A>
    Upon completion of the form, select <i>Link my existing Quick and Reilly account</i>. Selecting this
    option links your account(s) for online access. For more information on linking accounts, visit <A
    HREF="https://www.quickandreilly.com/out/signup/linkmain.html">Quick & Reilly Link Account
    Introduction</A>.<P>If you have any questions, please call Quick & Reilly at <b>(800) 837-
    7220</b>.]>
  </ACCESS_INSTRUCTIONS>
  <ACCOUNT_NUMBER_TERM>Account Number</ACCOUNT_NUMBER_TERM>
  <ACCOUNT_LOGIN_TERM>User ID</ACCOUNT_LOGIN_TERM>
  <ACCOUNT_PIN_TERM>Password</ACCOUNT_PIN_TERM>
  <SUP_SETTLEMENT>UNKNOWN</SUP_SETTLEMENT>
  <SUP_TRADE>UNKNOWN</SUP_TRADE>
  <SUP_ACCOUNT_DISCOVERY>UNKNOWN</SUP_ACCOUNT_DISCOVERY>
  <SUP_REGISTER_MY_COMPUTER>UNKNOWN
  </SUP_REGISTER_MY_COMPUTER>
```

<SUP_TRANSACTIONS>UNKNOWN</SUP_TRANSACTIONS>
<SUP_TAX_LOTS>UNKNOWN</SUP_TAX_LOTS>
<SUP_TEST_CREDENTIAL>UNKNOWN</SUP_TEST_CREDENTIAL>
<SUP_AAM>UNKNOWN</SUP_AAM>
<REQ_IN_SESSION_ACTIVATION>0</REQ_IN_SESSION_ACTIVATION>
 < SUP_OAUTH>UNKNOWN< /SUP_OAUTH>
 <IS_FEED>0</IS_FEED>
 <HEALTH_STATUS>GREEN</HEALTH_STATUS>
</FI>
<FI>
 <ID>2354</ID>
<NAME>AIM Funds</NAME>
<SERVICE_CATEGORY>Investment</SERVICE_CATEGORY>
<USERLOGIN_URL><http://www.aiminvestments.com/home></USERLOGIN_URL>
<REQ_SQA>0</REQ_SQA>
<ACCESS_INSTRUCTIONS>
 <![CDATA[This service uses AIM Funds' online Account Access to get your financial information. To use their Account Access you need your SSN or Tax ID and a PIN. If you do not have these, complete their online registration. For more information, visit their website.]>
</ACCESS_INSTRUCTIONS>
<ACCOUNT_NUMBER_TERM>Account Number</ACCOUNT_NUMBER_TERM>
<ACCOUNT_NUMBER_DETAILS>The 10-digit number as it appears on the AIM Funds website (no dashes or spaces).</ACCOUNT_NUMBER_DETAILS>
<ACCOUNT_LOGIN_TERM>SSN or Tax ID</ACCOUNT_LOGIN_TERM>
<ACCOUNT_LOGIN_DETAILS>Your Social Security Number or Tax Identification number, without dashes or spaces.</ACCOUNT_LOGIN_DETAILS>
<ACCOUNT_PIN_TERM>PIN</ACCOUNT_PIN_TERM>
<ACCOUNT_PIN_DETAILS>6-8 characters; must contain two numbers and two letters.</ACCOUNT_PIN_DETAILS>
<SUP_SETTLEMENT>UNKNOWN</SUP_SETTLEMENT>
<SUP_TRADE>UNKNOWN</SUP_TRADE>
<SUP_ACCOUNT_DISCOVERY>UNKNOWN</SUP_ACCOUNT_DISCOVERY>

```
<SUP_REGISTER_MY_COMPUTER>UNKNOWN
</SUP_REGISTER_MY_COMPUTER>
<SUP_TRANSACTIONS>UNKNOWN</SUP_TRANSACTIONS>
<SUP_TAX_LOTS>UNKNOWN</SUP_TAX_LOTS>
<SUP_TEST_CREDENTIAL>UNKNOWN</SUP_TEST_CREDENTIAL>
<SUP_AAM>UNKNOWN</SUP_AAM>
<REQ_IN_SESSION_ACTIVATION>0</REQ_IN_SESSION_ACTIVATION>
  < SUP_OAUTH>UNKNOWN< /SUP_OAUTH>
  <IS_FEED>0</IS_FEED>
</FI>
</FILIST>
</FIGETRS>
</DATACONNECTRS>
```

Export Data (Asynchronous)

Purpose

Retrieves accounts, holdings, transactions, and clients in CSV-delimited format.

Restrictions

This is an asynchronous operation. Please see the [Asynchronous Operations](#) section for general information on handling asynchronous operations.

Behavior

This operation exhibits the following behaviors:

1. The request is always asynchronous.
2. The final response of this request, when successful, is a ZIP file containing a README.txt and possibly CSV files. Each CSV file is only produced when at least one data record is found for the requested object type. Possible CSV files are listed in the response table on page [91](#). See [Claim File](#) operation.
3. Because this operation is asynchronous and has a non-standard final response format, you must use the following sequence of calls to obtain the desired data:
 - a. Submit a <DATAEXPORTRQ_A> request. This request will return to you a RECEIPT to be used in subsequent operations and CLAIM_WAIT as a suggested polling interval.
 - b. Submit a <DATACLAIMRQ> request with the receipt. Submit additional <DATACLAIMRQ> requests with the receipt to poll every CLAIM_WAIT duration for job completion until DATACLAIMRS says the operation is no longer in progress (either it completed successfully or took an error).
 - c. Submit a <FILECLAIMRQ> request to obtain the final response that contains the exported data. Do not skip the DATACLAIMRQ step. If you attempt a <FILECLAIMRQ> request without having received confirmation from DATACLAIMRQ that the operation is complete, then FILECLAIMRQ will tell you the receipt is expired.

Request: <DATAEXPORTRQ_A>

The request consists of the following two primary items:

- Type of information to include in the response. Choose from any combination of the following: Accounts, Holdings (positions), Transactions, Clients, and Investment Options. Designate the type of information to include using the <INCxxx> tags.
- Data to include in the response. You can specify data retrieval criteria for a set of Accounts, or opt to leave out the selection and get data for all accounts.

The <DATAEXPORTRQ_A> request can contain the following:

Tag	Required	Field	Description
<EXPORT_DATA_QUERY>		Aggregate	Like the GET_DATA_QUERY used by Get Data, but allows only an inner ACCOUNT_QUERY tag which will accept any number of IDs. If omitted, all data available to the user is returned.
<EXPORT_DATA_AS_OF_DATE>		DATE	Specifies an historical date. If not specified, default is 'current', non-historical data. If specified, it must be a date prior to today. To get the current data omit this tag. To get data as of the close of business yesterday, use yesterday's date. Although this tag primarily affects the holdings table, it has some effect on all three files: Accounts, Holdings (positions), and Transactions. The format will be the same, but many fields will be empty because there is no historical value available or because the field is not relevant from an historical context. For descriptions of the files, refer to the <i>ByAllAccounts Export</i> guide posted at http://www.byallaccounts.net/manuals/accountview/BAA_Export.PDF .
<INACCOUNTFILE/>	*		Empty tag. If present, specifies that Accounts CSV file should be included in the response if any accounts to export are found. If the tag is absent, the Accounts file is not included in the response.
<INHOLDINGFILE/>	*		Empty tag. If present, specifies that the Holdings (positions) CSV file should be included in the response if any holdings to export are found. If the tag is absent, the holdings file is not included in the response.
<INCTXFILE/>	*		Empty tag. If present, specifies that the Transactions CSV file should be included in the response. If absent, Transactions file is not included in the response.
<INCLIENTFILE/>	*		Empty tag. If present, specifies that Clients CSV file should be included in the response. If absent, Clients file is not included in the response.
<ININVOPTFILE/>	*		Empty tag. If present, specifies that the Investment options CSV file should be included in the response. If absent, the Investment options file is not included in the response.

* indicates that one of them is required.

<EXPORT_DATA_QUERY> Aggregate

The **<EXPORT_DATA_QUERY>** aggregate filters the objects to be retrieved. If omitted, all objects available to the caller are retrieved. The usage of EXPORT_DATA_QUERY is similar to the usage of GET_DATA_QUERY in the Get Data operation, however, only ACCOUNT_QUERY is allowed inside of an EXPORT_DATA_QUERY.

Data included in the response is filtered according to the accounts identified in the ACCOUNT_QUERY. So, if you identify a single Account to be retrieved using **<ACCOUNT_QUERY>**, and specify the **<INACCOUNTFILE/>** element only, the response data includes details for all the Accounts that were specified in the **<ACCOUNT_QUERY>**. Similarly, if you specify **<INCTXFILE/>** instead of **<INACCOUNTFILE/>**, the information for the transactions within the specified set of accounts is included in the response.

<EXPORT_DATA_QUERY> must contain the following:

Tag	Required	Field	Description
<ACCOUNT_QUERY>	√	Account.ID	Identifies the Accounts for which data should be retrieved. More than one <ACCOUNT> can be included.

The **ACCOUNT_QUERY** identifies accounts to be retrieved, grouped by Financial Profile:

```

<ACCOUNT_QUERY>
  <FP_ID>45226</FP_ID>
  <ID>123</ID>
  <ID>125</ID>
  <FP_ID>45227</FP_ID>
  <ID>126</ID>
  <ID>127</ID>
</ACCOUNT_QUERY>
    
```

<INCTXFILE> Aggregate

The presence of the INCTXFILE tag indicates that a Transactions CSV file should be created if any transactions to export are found. The INCTXFILE tag may also optionally contain a nested start date and end date. Specifically, the **<INCTXFILE>** aggregate:

- Can be an empty tag (in which case, **all Transactions** for the selected accounts are returned).

OR

- Can contain either one or both of the following to restrict the set of Transactions returned:

Tag	Required	Field	Description
<TX_START_DATE>		Transaction.EXECUTION_DATE	See <INCTXFILE> aggregate notes below.
<TX_END_DATE>		Transaction.EXECUTION_DATE	See <INCTXFILE> aggregate notes below.

<INCTXFILE> aggregate notes:

- **<TX_START_DATE>** and (optionally) **<TX_END_DATE>** are used to constrain the set of Transactions returned by **EXECUTION_DATE**:
 - The date range is applied to the **EXECUTION_DATE** field of the Transaction.
 - **YYYYMMDD**: The format used to identify the date, where **YYYY** is a four-digit year such as 2003, **MM** is a two-digit month identifier ranging from 01 (January) to 12 (December), and **DD** is a two-digit day identifier ranging from 01 to 31.
 - If there is a start date but no end date, the end date is set to the day of the API call.
 - The start date must be before or the same as the end date.
 - The date range is inclusive of start date and exclusive of end date. If a start date is specified, then all Transactions with execution dates on or after that date are included. If an end date is also specified, then only Transactions with an execution date before the given end date are included.

Response: <DATAEXPORTRS_A>

The <DATAEXPORTRS_A> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<RECEIPT>		RECEIPT64	Receipt provided in request.
<RECEIPT_EXP>		TIMESTAMP	Expiration time of <RECEIPT>, including time zone. Only present if <RECEIPT> is present.
<CLAIM_WAIT>		NUMBER	Number of milliseconds to wait before attempting to retrieve the results. Only present if <RECEIPT> is present.

Errors

This operation may return any of the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
65699	Error	The request is invalid or formatted incorrectly
65707	Error	The date is invalid or formatted incorrectly
65723	Error	The transaction date range is invalid
65731	Error	The transaction start date is invalid
65739	Error	The transaction end date is invalid
65843	Error	Only dates prior to today are allowed
65939	Error	The requested user was not found
65947	Error	Caller is not authorized to make this request
66099	Error	The <object name> could not be found
67466	Warning	Account file not completed
67474	Warning	Position file not completed
67482	Warning	Transaction file not completed
67499	Error	Bulk Export files not generated
67507	Error	At least one type of file must be included
67514	Warning	Investment Option file not completed
67522	Warning	Client file not completed
67722	Warning	Multiple files not completed

Sample XML

The following is a sample **<DATAEXPORTRQ>** request that gets three files (accounts, holdings, and transactions) for two accounts, with an historic date and a transaction date range.

```
<DATACONNECTRQ>  
  VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>JoeSmith1</LOGIN_NAME>  
    <LOGIN_PW>JoeSmith1</LOGIN_PW>  
  </LOGINRQ>  
  <DATAEXPORTRQ_A>  
    <EXPORT_DATA_QUERY>  
      <ACCOUNT_QUERY>  
        <FP_ID>202</FP_ID>  
        <ID>123456</ID>  
        <ID>123457</ID>  
      </ACCOUNT_QUERY>  
    </EXPORT_DATA_QUERY >  
    <EXPORT_DATA_AS_OF_DATE>20140604</EXPORT_DATA_AS_OF_DATE>  
    <INACCOUNTFILE/>  
    <INCHOLDINGFILE/>  
    <INCTXFILE>  
      <TX_START_DATE>20140504</TX_START_DATE>  
      <TX_END_DATE>20140604</TX_END_DATE>  
    </INCTXFILE>  
  </DATAEXPORTRQ_A>  
</DATACONNECTRQ>
```

The following is a sample of an initial **<DATAEXPORTRS_A>** response:

<DATACONNECTRS>

<VERSION>VERSION4.0</VERSION>

<LOGINRS>...</LOGINRS>

<DATAEXPORTRS_A>

<STATUS>

<ERRCODE>0</ERRCODE>

<ERRMSG>Operation in progress</ERRMSG>

</STATUS>

<RECEIPT>4456858471129290880</RECEIPT>

<RECEIPT_EXP>20030621183522 [-5:EST]</RECEIPT_EXP>

<CLAIM_WAIT>1000</CLAIM_WAIT>

</DATAEXPORTRS_A>

</DATACONNECTRS>

A Claim Data request submitting the given receipt should follow. The Claim Data request may need to be repeated until a final DATAEXPORTRS_A response is provided.

The following is a sample of a final **<DATAEXPORTRS_A>** response which indicates that the file is ready.

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>...</LOGINRS>  
    <DATAEXPORTRS_A>  
      <STATUS>  
        <ERRCODE>0</ERRCODE>  
        <ERRMSG>Success</ERRMSG>  
      </STATUS>  
      <RECEIPT>4456858471129290880<RECEIPT>  
    </DATAEXPORTRS_A>  
</DATACONNECTRS>
```

This operation should be followed by a Claim File request which will obtain the needed ZIP file.

Import Data (Asynchronous)

Purpose

Import Investors in CSV-delimited format.

Restrictions

- This is an asynchronous operation. Please see the [Asynchronous Operations](#) section for general information on handling asynchronous operations.
- This operation is available to Advisor and Assistant user types. It is not available to Administrators.

Behavior

This operation exhibits the following behaviors:

1. The request is always asynchronous.
2. The final response of this request, when successful, is a ZIP file containing a README.txt and possibly a CSV file. A CSV file is only produced when at least one data record is found for the object type. A description of the ZIP file contents is provided in the [Response: <DATAIMPORTRS_A>](#) section.
3. Because this operation is asynchronous and has a non-standard final response format, you must use the following sequence of calls to obtain the desired data:
 - a. Submit a <DATAEXPORTRQ_A> request. This request will return to you a RECEIPT to be used in subsequent operations and CLAIM_WAIT as a suggested polling interval.
 - b. Submit a <DATACLAIMRQ> request with the receipt. Submit additional <DATACLAIMRQ> requests with the receipt to poll every CLAIM_WAIT duration for job completion until DATACLAIMRS says the operation is no longer in progress (either it completed successfully or took an error).
 - c. Submit a <FILECLAIMRQ> request to obtain the final response that contains the exported data. Do not skip the DATACLAIMRQ step. If you attempt a <FILECLAIMRQ> request without having received confirmation from DATACLAIMRQ that the operation is complete, then FILECLAIMRQ will tell you the receipt is expired.

Request: <DATAIMPORTRQ_A>

The request consists of the following items:

- Type of object being imported (INVESTORS)
- Include the data to be imported in in-line CSV format (IMPORT_FILE).

The **<DATAIMPORTRQ_A >** request can contain the following:

Tag	Required	Field	Description
INVESTORS	√		Indicates that the IMPORT_FILE tag has Investors data. This data is used to create an investor that is assigned to the target Advisor. The optional <GENERATE_LOGINS> aggregate may be used.
IMPORT_FILE	√		Data in CSV format to be imported. See Import File Format .

√ indicates required field

The **<GENERATE_LOGINS>** aggregate for INVESTORS is optional. When it is used:

1. The system will generate a unique login and pre-expired password for the new users and send a welcome email to the new user with the login and pre-expired password.
2. When generating the unique login for the user, the system will try the user's email address as the login. If the email address is not *unique within* the system, then the system will attempt to create a unique login by truncating the email address to 28 characters and adding a "-nnn" string to the end where nnn is a randomly generated number from 000 to 999. If the system exhausts all possible generated logins without creating a unique one, then the row will not be imported. The result file will show a 'failure' for that row.
3. Rows in the import file that do not have an email address will fail to import.
4. GENERATE_LOGINS option can only be used to import users for firms that have 'new clients default to SSO' option OFF. Otherwise an error is returned by DataConnect at the Import operation level (i.e. it does not attempt to import anything in the file).

When used, the **<GENERATE_LOGINS>** aggregate can use any of the following optional tags to customize the email sent to investors.

Tag	Required	Field	Description
CC_SELF		BOOLEAN	Defaults to 0 (false). If set to 1 (true), will send a copy of the email to the sender with the login and password masked out. Regardless of this setting, a copy of the email will be sent to the Firm's BCC email address if one is defined for the Firm. If the sender is an Assistant, a copy will be sent to the Advisor with username and password masked.
EMAIL_SUBJECT *		CHAR900	If present, this text is used as the email subject for the welcome email instead of the default email subject.
EMAIL_PREPEND_MSG *		CHAR2048	If present, this text is prepended to the welcome email content.
INC_PRODUCT_LINK		BOOLEAN	Defaults to 0 (false). If set to 1 (true), then AccountView URL link is included in the welcome email.

* Do not use if Firm is set to not allow users to modify text of email sent to clients.

Import File Format

The import file is a .csv file with one header row and one row for each client to be created, up to a maximum of 10,000 rows. Any rows beyond 10,000 will be ignored and the README will indicate that not all rows were processed. The import file must minimally contain the required columns, and the column headers must match the Column header listed in the table below. The columns can be in any order, and extra columns not listed below are ignored, but tolerated.

Column Header	Data Type	Description	Required Column
FIRST_NAME	CHAR64	Investor's first name	√
MIDDLE_NAME	CHAR64	Investor's middle name	
LAST_NAME	CHAR64	Investor's last name	√
EMAIL	CHAR64	Investor's email address, must contain one @ character	
TAX_ID	CHAR32	Investor's tax ID (SSN or TIN). Should be numeric, with or without hyphen separators (e.g. 000-00-0000)	√*

√ indicates required field

√* this field is required if the Firm for this Investor has been configured to require TAX_ID

Import will determine if an Investor already exists for the target advisor using the following matching logic:

- If TAX_ID is provided in the Import file then it will be used to match to existing investors.
- If TAX_ID is not provided in the Import file, then FIRST_NAME, MIDDLE_NAME, and LAST_NAME will be used to match to existing Investors (but only existing Investors that have no TAX_ID value).

Matching is case-insensitive and Tax IDs are normalized to remove punctuation and non-numeric characters for matching (i.e. a Tax Id of 010-22-1234 will match a Tax Id of 010221234).

If Import determines through this matching process that an Investor in a row in the Import file matches an Investor that already exists for the target advisor, then the candidate investor will not be created. The Import result file will, for that row, contain a LOAD_STATUS column value of "Succeeded" and a LOAD_DETAILS column value of "Import record ignored because it matched an existing investor".

Response: <DATAIMPORTRS_A>

The <DATAIMPORTRS_A> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<RECEIPT>		RECEIPT64	Receipt provided in request.
<RECEIPT_EXP>		TIMESTAMP	Expiration time of <RECEIPT>, including time zone. Only present if <RECEIPT> is present.
<CLAIM_WAIT>		NUMBER	Number of milliseconds to wait before attempting to retrieve the results. Only present if <RECEIPT> is present.

- The RECEIPT associated with the DATAIMPORTRS_A operation which has already been established to be completed is used in the <FILECLAIMRQ> request to retrieve the <DATAIMPORTRS_A> generated .csv files.

A normal response to a FILECLAIMRS will be a ZIP file containing:

Filename	Required	Description
README.txt	√	A brief log file
Importstatus_investors.csv		The Investor import file with additional columns LOAD_STATUS and LOAD_DETAILS that give an operation status for the import of that row. The possible values for LOAD_STATUS and LOAD_DETAILS are shown in the following table.

LOAD_STATUS	LOAD_DETAILS
Succeeded	New investor record created from import record.
Succeeded	Import record ignored because it matched an existing investor.
Succeeded	New investor created from import record but failed to send welcome email to investor
Failed	Failed to import because a required field has no data: <name of field>
Failed	Failed to import because of invalid data in field.
Failed	Failed to import because of system error.
Failed	Failed to import because exhausted maximum attempts to generate unique login

Errors

This operation may return any of the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
65563	Error	The <field name> cannot exceed <max> characters
65947	Error	Caller is not authorized to make this request
65963	Error	User type not supported for this operation.
67995	Error	GENERATE_LOGINS cannot be used; usage conflicts with firm configuration that creates new Investors as SSO by default

Sample XML

The following is a sample **<DATAIMPORTRQ>** request for investors.

<DATACONNECTRQ>

<VERSION>Version4.0**<VERSION>**

<LOGINRQ>...**</LOGINRQ>**

<DATAIMPORTRQ_A>

<INVESTORS/>

<IMPORT_FILE>

"FIRST_NAME","MIDDLE_NAME","LAST_NAME","EMAIL","TAX_ID"

"Mary","C","Jones ","MCJones@email.com",""

"Mary","","Jones ","MJones@email.com",""

"Kip","S","Thorne ","KipperT@email.com","012345678"

"Rainer","","Weiss ","Weiss_R@email.com",""

"Barry","","Barish ","BarBar@email.com","123456789"

</IMPORT_FILE>

</DATAIMPORTRQ_A>

</DATACONNECTRQ>

The same can be expressed without the quotation marks:

<DATACONNECTRQ>

<VERSION>Version4.0**<VERSION>**

<LOGINRQ>...**</LOGINRQ>**

<DATAIMPORTRQ_A>

<INVESTORS/>

<IMPORT_FILE>

FIRST_NAME,MIDDLE_NAME,LAST_NAME,EMAIL,TAX_ID

Mary,C,Jones,MCJones@email.com,

Mary,,Jones,MJones@email.com,

Kip,S,Thorne,KipperT@email.com,012345678

Rainer,,Weiss,Weiss_R@email.com,

Barry,,Barish,BarBar@email.com,123456789

</IMPORT_FILE>

</DATAIMPORTRQ_A>

</DATACONNECTRQ>

The following is a sample of an initial **<DATAIMPORTRS_A>** response:

<DATACONNECTRS>

<VERSION>VERSION4.0**</VERSION>**

<LOGINRS>...**</LOGINRS>**

<DATAIMPORTRS_A>

<STATUS>

<ERRCODE>0**</ERRCODE>**

<ERRMSG>Operation started**</ERRMSG>**

</STATUS>

<RECEIPT>4846791201899833473**</RECEIPT>**

<RECEIPT_EXP>20180817110528 [-5:EDT]**</RECEIPT_EXP>**

<CLAIM_WAIT>1000**</CLAIM_WAIT>**

</DATAIMPORTRS_A>

</DATACONNECTRS>

A Claim Data request submitting the given receipt should follow. The Claim Data request may need to be repeated until a final DATAIMPORTRS_A response is provided.

The following is a sample of **<DATACLAIMRQ>** response:

<DATACONNECTRS>

<VERSION>VERSION4.0**</VERSION>**

<LOGINRS>...**</LOGINRS>**

```
<DATACLAIMRQ>
  <RECEIPT>4846791201899833473</RECEIPT>
</DATACLAIMRQ>
</DATACONNECTRQ>
```

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <DATAIMPORTRS_A>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </DATAIMPORTRS_A>
</DATACONNECTRS>
```

The following is a sample of <FILECLAIMRQ> request:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <FILECLAIMRQ>
    <RECEIPT>4846791201899833473</RECEIPT>
  </FILECLAIMRQ>
```

List Available Tax Lots

Purpose

Request a list of tax lot files.

Restrictions

1. This operation is available to Advisor, Assistant, and Administrator user types.
2. The requesting user's firm must have the **Enable Tax Lots (New Solution)** enabled to use this tax lot operation.

Behavior

This operation exhibits the following behaviors:

1. The following tax lot file types are available through this operation:
 - a. Open lots: a detailed breakdown of the position indicating acquisition date and cost of each lot
 - b. Closed lots: a historical record indicating when an open lot was sold or transferred out of an account. Closed lot files vary by financial institution and may be delivered as delta files (new/updated closed lots since the last file) or full files that include all the closed lots over a longer period.
2. The list can be filtered by account credential or a specific financial institution, but not both.
3. The list can combine filter options with a timestamp to request a list of files that were created on or after the timestamp.
4. From the list of available tax lot files, the user can download a single zipped file using the existing <FILECLAIMRQ>.

Request: <TAXLOTFILELISTRQ>

Tag	Required	Data Type	Description
<AC_ID>	*	AccountCredential.ID	Account Credential identifier. Use this identifier to request listing of the tax lot files for this Account Credential. Cannot be used with <FI_ID>.
<FI_ID>	*	FinancialInstitution.ID	Financial Institution identifier. Use this identifier to request a listing of the tax lot files for this Financial Institution. Cannot be used with <AC_ID>.
<CREATION_TIMESTAMP>		TIMESTAMP	Request a listing of the tax lot files that were created on or after this timestamp. This parameter can be combined with AC_ID or FI_ID.

"*" = AC_ID and FI_ID are optional parameters. You can specify none, or one, but not both.

Response: <TAXLOTFILELISTRS>

The <TAXLOTFILELISTRS> contains the following.

Tag	Required	Data Type	Description
ID	√	ID	The unique ID for this TAXLOTFILE
AC_ID	√	ID	Correlates to the Account Credential in the operation
FP_ID	√	ID	Correlates to the Financial Profile in the operation
FI_ID	√	ID	Correlates to the Financial Institution the custodian file came from
FILE_TYPE	√	CHAR128	Tax lot File type, one of: OPEN_LOTS, CLOSED_LOTS
FILE_CONTENT_STYLE	√	CHAR128	Tax lot file content style, one of: FULL, COMPLETE, NEW_ONLY, KEYED_DELTA.
FILE_NAME	√	CHAR128	The name of the file.
CREATION_DATETIME	√	TIMESTAMP	The timestamp at which the file was created.
FILE_EXPIRATION_DATETIME	√	TIMESTAMP	The timestamp at which the export file will expire.
CUSTODIAN_FILE_NAME	√	CHAR128	File name from the financial institution.
CUSTODIAN_FILE_POST_TIME	√	TIMESTAMP	The timestamp when the file was made available by the custodian.
CUSTODIAN_FILE_DATEFROM	√	DATE	Beginning business day date for the range of data included in the file.
CUSTODIAN_FILE_DATETO	√	DATE	Ending business day date for the range of data included in the file.
FI_NAME	√	CHAR128	Institution name

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
65779	Error	The <field name> is outside the valid range valid range of 0 to 9999999999999999
65867	Error	The timestamp is invalid or formatted incorrectly
65947	Error	Caller is not authorized to make this request
65963	Error	User type not supported for this operation.
67355	Error	Tax lot gathering is not enabled for this firm.

Sample XML

The following is a sample **<TAXLOTFILELISTRQ>** request using FI_ID and the optional creation timestamp.

```
<DATACONNECTRQ>  
  <VERSION>4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>Chris_Advisor</LOGIN_NAME>  
    <LOGIN_PW>ChrisAdvisorPW</LOGIN_PW>  
  </LOGINRQ>  
  <TAXLOTFILELISTRQ>  
    <FI_ID>1234</FI_ID>  
    <CREATION_TIMESTAMP>20241002140001 [-4:EDT]</CREATION_TIMESTAMP>  
  </TAXLOTFILELISTRQ>  
</DATACONNECTRQ>
```

The following is a sample corresponding **<TAXLOTFILELISTRS>** response.

```
<DATACONNECTRS>  
  <VERSION>4.0</VERSION>  
  <LOGINRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>  
    </STATUS>  
  </LOGINRS>  
  <TAXLOTFILELISTRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>
```

```
</STATUS>
<TAXLOTFILELIST>
  <TAXLOTFILE>
    <ID>76802</ID>
    <AC_ID>1234</AC_ID>
    <FP_ID>5678</FP_ID>
    <FI_ID>7890</FI_ID>
    <FILE_TYPE>CLOSED_LOTS</FILE_TYPE>
    <FILE_CONTENT_STYLE>COMPLETE</FILE_CONTENT_STYLE>
    <FILE_NAME>BAACLOSEDLOTS_123456_20240819_COMPLETE.zip</FILE_NAME>
    <CREATION_DATETIME>20240913103304 [-4:EDT]</CREATION_DATETIME>
    <FILE_EXPIRATION_DATETIME>20241013103304 [-4:EDT]</FILE_EXPIRATION_DATETIME>
    <CUSTODIAN_FILE_NAME>abc</CUSTODIAN_FILE_NAME>
    <CUSTODIAN_FILE_POST_TIME>20240912103304 [-4:EDT]</CUSTODIAN_FILE_POST_TIME>
    <CUSTODIAN_FILE_DATEFROM>20240701</CUSTODIAN_FILE_DATEFROM>
    <CUSTODIAN_FILE_DATETO>20240901</CUSTODIAN_FILE_DATETO>
    <FI_NAME>Bank of America</FI_NAME>
  </TAXLOTFILE>
  .
  .
  .
  .
</TAXLOTFILELIST>
</TAXLOTFILELISTRS>
</DATACONNECTRS>
```

Modification Operations

This section defines operations for creating, modifying, or deleting data in DataConnect.

Add User

Purpose

Creates a new User. The User consists of a Person object and optionally the following objects: Login, Financial Profile, and Profile Access. **Add User** provides for creation of appropriate objects for each of the different User types: Investor, Advisor, Assistant, and Consultant.

Restrictions

1. Customers using DataConnect are contractually bound to provide for the presentation and acceptance of the appropriate (ByAllAccounts or Private Label) User Agreement, Privacy statement, and Security statement. Since the customer is building the User Interface (UI) that interacts with the User, ByAllAccounts has no control over the presentation and user acceptance of these items.
2. Users may only be created in accordance with the usage style specified for the ByAllAccounts Customer (firm). The firm specifies either the Investor-managed or Advisor-managed usage style. In the Investor-managed style, Investors and Consultants may be created. In the Advisor-managed usage style, Investors, Consultants, Assistants, and Advisors may be created. An attempt to create a User that is not consistent with the firm's usage style results in an error.
3. Login names must be unique throughout the ByAllAccounts service.

Behavior

This operation exhibits the following behavior:

1. For the new User:
 - The new User is an Investor, Advisor, Assistant, or Consultant.
 - A Financial Profile is automatically created for Investor Users. If no **OWN_FP_ACCESS** is specified by the request, the User's access to their own profile is set as follows:
 - Investor-managed usage style: Access = READWRITE
 - Advisor-managed usage style: Access = READWRITE
 - The User may optionally have a **LOGIN**.
 - Investors in an Advisor-managed firm may optionally be assigned to one Advisor.
 - All alert settings for this User are turned **OFF**.
2. Optionally, the **LOGIN_PW** can be pre-expired, meaning that at the User's first attempt to log in, a new password must be established.
3. To pre-expire a password, use **<PW_EXPIRE_DATE>** to set a password expiration date prior to the current day.
4. If no password expiration date is set, the password never expires. Use **<PW_EXPIRE_DATE>** to set the password expiration date to a point in the future to force Users to change their passwords at or after a certain time.

User Notifications

If the **<NOTIFY_USER/>** option is sent and the function completes successfully, the following user notifications are sent:

1. If the new User has a Login, a message is sent to the User's email address. By default, the email welcomes the user to the ByAllAccounts service and provides Login information, a link to Login page (if configured on the firm), and password (if pre-expired). Passwords that are not pre-expired are not included in the notification.
2. When the email recipient is an Investor, the message may use a customized email template that controls the contents of the email's subject and message text. For details about customizing email templates, refer to the AccountView Customization Guide.
http://www.byallaccounts.net/Manuals/Accountview/AV_Customization_Guide.PDF.
3. If Person is an Investor with an associated Advisor, a message is sent to the Advisor notifying the Advisor of the new client.
4. If the firm is configured to have all emails sent to investors be blind carbon copied (Bcc'd) to a specified email address, then a copy of the message is sent to that email address. If present in the email, the user's login and password are automatically masked.

Request: <USERADDRQ>

The **<USERADDRQ>** can contain the following:

Tag	Required	Field	Description
<NOTIFY_USER/>			If present, User notifications relating to the ADD operation are sent. If absent, no notification is sent.
<PERSON>	√	See <PERSON> aggregate below.	
<LOGIN>		See <LOGIN> aggregate below.	
<OWN_FP_ACCESS>		Profile_Access.ACCESS	Access granted to this User's Financial Profile. Can be specified only if the Person.ROLE is INVESTOR and the firm's usage model is Advisor-managed.
<CLIENT_FP_ACCESS>		Profile_Access.ACCESS	For an ASSISTANT, the access level granted to the assistant for the advisor's clients. Can be specified only if Person.ROLE is ASSISTANT and the firm's usage model is Advisor-managed. The only accepted values are READ and READWRITE.

<ADVISOR_USER_IDENT>			Contains the <USER_IDENT> of the related Advisor. Can be specified only if the Person.ROLE is INVESTOR or ASSISTANT and the firm's usage model is Advisor-managed. For Investors this is the advisor who manages this client's assets. For assistants this is the advisor to whom the Assistant has access.
<CONSULTANT_TO>		Profile_Access.ACCESS	The USER_IDENT ID of an existing person. Required if ROLE is CONSULTANT, otherwise ignored.

The <PERSON> aggregate contains the following:

Tag	Required	Field	Description
<FIRM_TAG1>		Person.FIRM_TAG1	
<FIRM_TAG2>		Person.FIRM_TAG2	
<FIRM_TAG3>		Person.FIRM_TAG3	
<ROLE>	√	Person.ROLE	ROLE is set at creation time and cannot be changed later.
<FIRST_NAME>	√	Person.FIRST_NAME	
<MIDDLE_NAME>		Person.MIDDLE_NAME	
<LAST_NAME>	√	Person.LAST_NAME	
<EMAIL_ADDRESS>		Person.EMAIL_ADDRESS	Required if a login name is provided.
<PHONE>		Person.PHONE	
<TAX_ID>		Person.TAX_ID	
<IS_SSO>		Person.IS_SSO	Determines whether the person has single sign on (SSO) access. Can be 1 to indicate true, or 0 to indicate false. If not explicitly set in <USERADDRQ>, the default value is as set for the user type at the firm level. Can only be set by an administrator.

The **<LOGIN>** aggregate contains the following:

Tag	Required	Field	Description
<LOGIN_NAME>	√	Login.LOGIN_NAME	
<LOGIN_PW>	√	Login.LOGIN_PW	
<PW_EXPIRE_DATE>		Login.PW_EXPIRE_DATE	
<PASSWORD_HINT>		Login.PASSWORD_HINT	

Response: <USERADDRS>

The **<USERADDRS>** response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<ID>		Person.ID	Unique ID for the new User.
<FP_ID>		Financial_Profile.ID	ID for the Financial Profile created (if any) for the new User.

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
56803	Error	The new password does not meet minimum length requirements.
56811	Error	The new password cannot be a single repeated character.
56819	Error	The new password cannot be all letters or all numbers.
56835	Error	The password has too many sequential or repeating characters (e.g. AAAA or 1234).
56843	Error	The new password cannot be the same as the login.
56851	Error	The new password must have at least one letter.
65555	Error	The <field name> is required
65563	Error	The <field name> cannot exceed <max> characters
65707	Error	The date is invalid or formatted incorrectly
65755	Error	The user type must be one of INVESTOR, ADVISOR, or CONSULTANT.
65763	Error	Email addresses must be in the form x@x.x and cannot contain spaces
66275	Error	A user with the login name already exists
66418	Warning	The user was successfully created, but an error occurred while sending email to the user
66651	Error	Invalid FP Access value. Valid values are: NONE, READ, READLIMITEDWRITE, and READWRITE
66658	Warning	CONSULTANT_TO value is ignored, user does not have CONSULTANT role
66666	Warning	ADVISOR_USER_IDENT value is ignored, an advisor cannot be assigned to this user
66675	Error	CONSULTANT_TO user could not be found
66683	Error	ADVISOR_USER_IDENT user could not be found
66691	Error	CONSULTANT_TO user has no financial profile
66699	Error	The OWN_FP_ACCESS value was ignored; it can only be specified if the user has role INVESTOR and the firm's usage model is Advisor-managed
66707	Error	The new user was not notified because no LOGIN was defined
66715	Error	Advisors may not be created in a firm that uses the Investor-managed model
66722	Warning	User was added as a client of the calling Advisor. ADVISOR_USER_IDENT value was ignored
67139	Error	Invalid Client FP Access value. This field is only valid for Assistants. Valid values are: READ and READWRITE
67147	Error	The CLIENT_FP_ACCESS value was ignored; it can only be specified if the user has role ASSISTANT and the firm's usage model is Advisor-managed
67155	Error	Assistants may not be created in a firm that uses the Investor-managed model
67305	Error	Only administrators may explicitly set IS_SSO value.
67930	Warning	The new user was not notified because the user has single sign on access.

Sample XML

This section contains several examples for **<USERADDRQ>** request. Each demonstrates adding a different combination of user type and permission. In these examples, **<IS_SSO>** is not specified so the user gets the default single sign on (SSO) permission for their user type as set for their Firm.

Add an Advisor user

The following sample **<USERADDRQ>** request creates an Advisor user.

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  <USERADDRQ>  
    <PERSON>  
      <ROLE>ADVISOR</ROLE>  
      <FIRST_NAME>Chris</FIRST_NAME>  
      <LAST_NAME>Mitchell</LAST_NAME>  
      <EMAIL_ADDRESS>ChrisMitchell@aol.com</EMAIL_ADDRESS>  
    </PERSON>  
    <LOGIN>  
      <LOGIN_NAME>CMitchell</LOGIN_NAME>  
      <LOGIN_PW>POFDW&</LOGIN_PW>  
    </LOGIN>  
  </USERADDRQ>  
</DATACONNECTRQ>
```

The following is a corresponding sample **<USERADDRS>** response:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  <USERADDRS>
```

```
<STATUS>
  <ERRCODE>0</ERRCODE>
  <ERRMSG>Success</ERRMSG>
</STATUS>
<ID>8084</ID>
</USERADDRS>
</DATACONNECTRS>
```

Add a READWRITE Assistant

The following sample **<USERADDRQ>** request creates a read-write assistant user.

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERADDRQ>
    <PERSON>
      <ROLE>ASSISTANT</ROLE>
      <FIRST_NAME>Arlen</FIRST_NAME>
      <MIDDLE_NAME>Alex</MIDDLE_NAME>
      <LAST_NAME>Staines</LAST_NAME>
      <EMAIL_ADDRESS>AAStaines@aol.com</EMAIL_ADDRESS>
    </PERSON>
    <LOGIN>
      <LOGIN_NAME>AStaines</LOGIN_NAME>
      <LOGIN_PW>6M%Y!?!2-R</LOGIN_PW>
      <PW_EXPIRE_DATE>20101231</PW_EXPIRE_DATE>
    </LOGIN>
    <CLIENT_FP_ACCESS>READWRITE</CLIENT_FP_ACCESS>
```

```
</USERADDRQ>
</DATACONNECTRQ>
```

The following is a corresponding sample **<USERADDRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERADDRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <ID>22044</ID>
  </USERADDRS>
</DATACONNECTRS>
```

Add a READONLY Assistant

The following **<USERADDRQ>** request creates an read-only assistant user.

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERADDRQ>
    <PERSON>
      <ROLE>ASSISTANT</ROLE>
      <FIRST_NAME>Alex</FIRST_NAME>
      <MIDDLE_NAME>Baylo</MIDDLE_NAME>
      <LAST_NAME>McGuire</LAST_NAME>
      <EMAIL_ADDRESS>ABMcGuire@aol.com</EMAIL_ADDRESS>
```

```
</PERSON>
<LOGIN>
  <LOGIN_NAME>AMcGuire</LOGIN_NAME>
  <LOGIN_PW>98P4PC10%</LOGIN_PW>
  <PW_EXPIRE_DATE>20101231</PW_EXPIRE_DATE>
</LOGIN>
<CLIENT_FP_ACCESS>READ</CLIENT_FP_ACCESS>
</USERADDRQ>
</DATACONNECTRQ>
```

The following is a corresponding sample **<USERADDRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERADDRS>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Success</ERRMSG>
  </STATUS>
  <ID>22045</ID>
  </USERADDRS>
</DATACONNECTRS>
```

Add a READWRITE Investor

The following is a sample **<USERADDRQ>** request creates a read-write investor user.

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
    <LOGINRQ>...</LOGINRQ>
  <USERADDRQ>
    <PERSON>
      <ROLE>INVESTOR</ROLE>
      <FIRST_NAME>Adley</FIRST_NAME>
      <MIDDLE_NAME>Barrett</MIDDLE_NAME>
      <LAST_NAME>Ludwig</LAST_NAME>
      <EMAIL_ADDRESS>ABLudwig@aol.com</EMAIL_ADDRESS>
      <PHONE>111-555-1111</PHONE>
      <TAX_ID>11111</TAX_ID>
    </PERSON>
    <LOGIN>
      <LOGIN_NAME>ALudwig</LOGIN_NAME>
      <LOGIN_PW>ZSAYU:E&amp;</LOGIN_PW>
      <PW_EXPIRE_DATE>20101231</PW_EXPIRE_DATE>
    </LOGIN>
    <OWN_FP_ACCESS>READWRITE</OWN_FP_ACCESS>
  </USERADDRQ>
</DATACONNECTRQ>
```

The following is a corresponding sample **<USERADDRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERADDRS>
```

```
<STATUS>
  <ERRCODE>0</ERRCODE>
  <ERRMSG>Success</ERRMSG>
</STATUS>
<ID>22046</ID>
<FP_ID>19693</FP_ID>
</USERADDRS>
</DATACONNECTRS>
```

Add a READONLY Investor

The following sample **<USERADDRQ>** request creates a read-only investor user.

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERADDRQ>
    <PERSON>
      <ROLE>INVESTOR</ROLE>
      <FIRST_NAME>Ash</FIRST_NAME>
      <MIDDLE_NAME>Beck</MIDDLE_NAME>
      <LAST_NAME>Tracy</LAST_NAME>
      <EMAIL_ADDRESS>AshTracy@aol.com</EMAIL_ADDRESS>
      <PHONE>222-222-2222</PHONE>
      <TAX_ID>22222</TAX_ID>
    </PERSON>
    <LOGIN>
      <LOGIN_NAME>ABTracy</LOGIN_NAME>
      <LOGIN_PW>Kamp;IMH?</LOGIN_PW>
      <PW_EXPIRE_DATE>20101231</PW_EXPIRE_DATE>
    </LOGIN>
```

```
<OWN_FP_ACCESS>READ</OWN_FP_ACCESS>
</USERADDRQ>
</DATACONNECTRQ>
```

The following is a corresponding sample **<USERADDRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERADDRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <ID>22047</ID>
    <FP_ID>19694</FP_ID>
  </USERADDRS>
</DATACONNECTRS>
```

Add a CREDENTIAL-WRITE Investor

The following sample **<USERADDRQ>** request creates a credential-write investor:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERADDRQ>
    <PERSON>
      <ROLE>INVESTOR</ROLE>
      <FIRST_NAME>Taylor</FIRST_NAME>
      <MIDDLE_NAME>Tatum</MIDDLE_NAME>
    </PERSON>
  </USERADDRQ>
</DATACONNECTRQ>
```

```
<LAST_NAME>Tierney</LAST_NAME>
<EMAIL_ADDRESS>TTierney@aol.com</EMAIL_ADDRESS>
<PHONE>333-333-3333</PHONE>
<TAX_ID>33333</TAX_ID>
</PERSON>
<LOGIN>
  <LOGIN_NAME>TTierney</LOGIN_NAME>
  <LOGIN_PW>DCHR%!TP3</LOGIN_PW>
  <PW_EXPIRE_DATE>20101231</PW_EXPIRE_DATE>
</LOGIN>
<OWN_FP_ACCESS>READLIMITEDWRITE</OWN_FP_ACCESS>
</USERADDRQ>
</DATACONNECTRQ>
```

The following is a corresponding sample **<USERADDRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERADDRS>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Success</ERRMSG>
  </STATUS>
  <ID>22048</ID>
  <FP_ID>19695</FP_ID>
  </USERADDRS>
</DATACONNECTRS>
```

Unsubscribe User

Purpose

Unsubscribes a User from the service. The treatment of the User in this operation varies by User type:

- **Investor:** All information is removed, including Portfolios, Accounts, Holdings, and Transactions. Once removed, there is no means to restore this information. If any Consultants exist for the Investor's Financial Profile, they are removed as well. Note: It is not advisable to unsubscribe a system-created Unassigned Investor.
- **Advisor:** The Advisor's information is removed (Person, Login, Profile Access, Financial Profile) and their clients are left unassigned. If any Consultants exist for the Advisor's Financial Profile, they are removed as well. The clients' Financial Profiles and User information are left intact.
- **Assistant:** The Assistant's User Information (Person, Login, Profile Access) is removed from the system. No changes are made to the Advisor or Investors or Financial Profiles to which the Assistant had access.
- **Consultant:** Consultant's User information (Person, Login, Profile Access) is removed from the system. No changes are made to the Financial Profile to which the Consultant had access.

Restrictions

None.

Behavior

This operation exhibits the following behaviors:

1. The **Unsubscribe User** operation is permanent. Once information is deleted, it cannot be recovered.
2. The **Unsubscribe User** operation can be used in one of the following ways:
 - An Administrative User unsubscribes the User, in which case no **LOGIN_NAME** or **LOGIN_PW** are provided.
 - An Advisor unsubscribes one of their clients, in which case no **LOGIN_NAME** or **LOGIN_PW** are provided.
 - The User unsubscribes himself by providing their current **LOGIN_NAME** and **LOGIN_PW**.

User Notifications

The following notifications are sent if the **<NOTIFY_USER/>** option is chosen and this operation completes successfully:

1. A message is sent to the person's email address that User was unsubscribed.
2. When the email recipient is an Investor, the message may use a customized email template that controls the subject and content of the message. For details about customizing email templates, refer to the AccountView Customization Guide. http://www.byallaccounts.net/Manuals/Accountview/AV_Customization_Guide.PDF.
3. A message is sent to each Consultant's email address that they no longer have access to the service.
4. If the firm is configured to have all emails sent to investors be blind carbon copied (Bcc'd) to a specified email address, then a copy of the message is sent to that email address. If present in the email, the user's login and password are automatically masked.

Request: <USERUNSUBSCRIBERQ>

The <USERUNSUBSCRIBERQ> can contain the following:

Tag	Required	Field	Description
<NOTIFY_USER/>			Empty tag. If present, then User notifications relating to the delete operation are sent. If absent, no notification is sent.
<USER_IDENT>	√	One of: Person.ID, Person.FIRM_TAG1, or Login.LOGIN_NAME	Identifies the User to be unsubscribed. See <USER_IDENT> aggregate description .
<LOGIN_NAME>	*	Login.LOGIN_NAME	LOGIN_NAME of the User to be unsubscribed.
<LOGIN_PW>	*	Login.LOGIN_PW	LOGIN_PW corresponding to LOGIN_NAME

* If either <LOGIN_NAME> or <LOGIN_PW> are present, both must be present. These items are not required for sufficiently privileged Administrator or Advisor to unsubscribe another User. However, if they are provided, the credentials are verified. If they are not valid, the Unsubscribe operation does not proceed.

Response: <USERUNSUBSCRIBERS>

The <USERUNSUBSCRIBERS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description .
<USER_IDENT>			The <USER_IDENT> values supplied by the request to identify the User. See <USER_IDENT> aggregate description .
<ID>		Person.ID	

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
56003	Error	The login name or password is incorrect
65779	Error	The <field name> is outside the valid range valid range of 0 to 999999999999999999
65939	Error	The requested user was not found
66434	Warning	The user was successfully unsubscribed, but an error occurred while sending email to the user
67930	Warning	The new user was not notified because the user has single sign on access.

Sample XML

The following is a sample <USERUNSUBSCRIBERQ> request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERUNSUBSCRIBERQ>
    <USER_IDENT>
      <PERSON_ID>2300</PERSON_ID>
    </USER_IDENT>
    <LOGIN_NAME>jsmith</LOGIN_NAME>
    <LOGIN_PW>asd98uvv3</LOGIN_PW>
  </USERUNSUBSCRIBERQ>
</DATACONNECTRQ>
```

The following is a corresponding sample <USERUNSUBSCRIBERS> response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>
  <USERUNSUBSCRIBERS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <USER_IDENT>
      <PERSON_ID>2300</PERSON_ID>
    </USER_IDENT>
    <ID>2300</ID>
  </USERUNSUBSCRIBERS>
</DATACONNECTRS>
```

Modify User

Purpose

Modifies information for a User.

Restrictions

There is currently no way to request that a password be reset to an automatically generated password and sent to the email address of the corresponding Login.

Note: It is not advisable to modify a system-created Unassigned Investor.

Behavior

This operation exhibits the following behaviors:

1. This operation provides for the modification of sensitive data, namely **LOGIN_NAME**, **LOGIN_PW** (and corresponding **PASSWORD_HINT**) and **EMAIL_ADDRESS**. The current **LOGIN_PW** for a Login must be provided to effect a change to any of these sensitive fields.
2. If a new value is supplied for **LOGIN_PW** and no value is supplied for **PASSWORD_HINT**, **PASSWORD_HINT** is set to null when the **LOGIN_PW** is saved to the data store.
3. If changing **LOGIN_PW**, it can be optionally requested that the password be pre-expired. To do so, use **<PW_EXPIRE_DATE>** to set the password expiration date.
4. If **FIRM_TAG1** is used to identify the User to be modified, then it must identify only one User. If more than one User within the scope of the caller's access is identified by **FIRM_TAG1**, this operation returns an error.

Security

This operation varies according to the privileges of the User performing the operation:

Only Users with **ROLE ADMINISTRATOR**, **ADVISOR**, or **ASSISTANT** can modify Users other than themselves. Each is limited to modifying Users within their scope of control. For the **ADMINISTRATOR**, this is at the firm level. For the **ADVISOR**, only Users for whom the **ADVISOR** has access to their Financial Profile can be modified. For the **ASSISTANT**, only Users for whom the **Assistant** has **READWRITE** access to their Financial Profile can be modified.

User Notifications

The following notifications are sent if the **<NOTIFY_USER/>** option is sent and the function completes successfully:

1. If Login, email address, or password is changed, a message is sent to the person's email address that this change was made. If the email address is changed, a message is sent to the old email address.
2. When the email recipient is an Investor, the message may use a customized email template that controls the contents of the email's subject and message text. For details about customizing email templates, refer to the AccountView Customization Guide.
http://www.byallaccounts.net/Manuals/Accountview/AV_Customization_Guide.PDF.
3. If an Investor is assigned to a new Advisor, the new Advisor is notified.
4. If an Investor is unassigned from an Advisor, the Advisor is notified.

- If the firm is configured to have all emails sent to investors be blind carbon copied (Bcc'd) to a specified email address, then a copy of the message is sent to that email address. If present in the email, the user's login and password are automatically masked.

Request: <USERMODRQ>

The <USERMODRQ> can contain the following:

Tag	Required	Field	Description
<NOTIFY_USER/>			If present, User notifications relating to the UPDATE operation are sent. If absent, no notification is sent.
<USER_IDENT>	√	One of: Person.ID, Person.FIRM_TAG1, or Login.LOGIN_NAME	Identifies the User to be modified. See <USER_IDENT> aggregate description .
<CURRENT_LOGIN_PW>		Login.LOGIN_PW	The current password for LOGIN_NAME .
<PERSON>		See <PERSON> aggregate .	
<LOGIN>		See <LOGIN> aggregate .	
<OWN_FP_ACCESS>		Profile_Access.ACCESS	Access granted to this User's Financial profile. Can be specified only if this User's ROLE is INVESTOR and firm's usage model is Advisor-managed.
<CLIENT_FP_ACCESS>		Profile_Access.ACCESS	For an ASSISTANT, the access level granted to the assistant for the advisor's clients. Can be specified only if Person.ROLE is ASSISTANT and the firm's usage model is Advisor-managed. The only accepted values are READ and READWRITE.
<ADVISOR_USER_IDENT>			Contains the <USER_IDENT> of the related Advisor. Can be specified only if the Person.ROLE is INVESTOR and the firm's usage model is Advisor-managed. For Investors this is the advisor who manages this client's assets. Cannot be changed for Assistants.

Notes on <ADVISOR_USER_IDENT>:

To unassign an Advisor from an Investor and leave that Investor with no Advisor, specify <ADVISOR_USER_IDENT> as an empty tag.

The <PERSON> aggregate contains the following:

Tag	Required	Field	Description
<FIRM_TAG1>		Person.FIRM_TAG1	
<FIRM_TAG2>		Person.FIRM_TAG2	
<FIRM_TAG3>		Person.FIRM_TAG3	
<ROLE>		Person.ROLE	Cannot be changed. Any value supplied is ignored and a warning status is returned.
<FIRST_NAME>		Person.FIRST_NAME	
<MIDDLE_NAME>		Person.MIDDLE_NAME	
<LAST_NAME>		Person.LAST_NAME	
<EMAIL_ADDRESS>		Person.EMAIL_ADDRESS	
<PHONE>		Person.PHONE	
<TAX_ID>		Person.TAX_ID	
<IS_SSO>		Person.IS_SSO	Determines whether the person has single sign on (SSO) access. Can be 1 to indicate true, or 0 to indicate false. If not provided in <USERMODRQ>, the default value is as set for the user type at the firm level. Can only be changed by an administrator.

The <LOGIN> aggregate contains the following:

Tag	Required	Field	Description
<LOGIN_NAME>		Login.LOGIN_NAME	
<LOGIN_PW>		Login.LOGIN_PW	
<PW_EXPIRE_DATE>		Login.PW_EXPIRE_DATE	
<PASSWORD_HINT>		Login.PASSWORD_HINT	

Response: <USERMODRS>

The <USERMODRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			The <USER_IDENT> values supplied by the request to identify the User. See <USER_IDENT> aggregate description .
<ID>	√	Person.ID	Unique ID for the User.

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
56003	Error	The login name or password is incorrect
56027	Error	The password has expired
56803	Error	The new password does not meet minimum length requirements.
56811	Error	The new password cannot be a single repeated character.
56819	Error	The new password cannot be all letters or all numbers.
56827	Error	The new password must be different than your old password.
56835	Error	The password has too many sequential or repeating characters (e.g. AAAA or 1234).
56843	Error	The new password cannot be the same as the login.
56851	Error	The new password must have at least one letter.
65555	Error	The <field name> is required
65563	Error	The <field name> cannot exceed <max> characters
65763	Error	Email addresses must be in the form x@x.x and cannot contain spaces
65779	Error	The <field name> is outside the valid range valid range of 0 to 9999999999999999
65867	Error	The timestamp is invalid or formatted incorrectly
66107	Error	The requested user was not found
66426	Warning	The user was successfully modified, but an error occurred while sending email to the user
66651	Error	Invalid FP Access value. Valid values are: NONE, READ, READLIMITEDWRITE, and READWRITE
66658	Warning	CONSULTANT_TO value is ignored, user does not have CONSULTANT role
66666	Warning	ADVISOR_USER_IDENT value is ignored, an advisor cannot be assigned to this user
66675	Error	CONSULTANT_TO user could not be found
66683	Error	ADVISOR_USER_IDENT user could not be found
66691	Error	CONSULTANT_TO user has no financial profile
66699	Error	The OWN_FP_ACCESS value was ignored; it can only be specified if the user has role INVESTOR and the firm's usage model is Advisor-managed
66707	Error	The new user was not notified because no LOGIN was defined
66715	Error	Advisors may not be created in a firm that uses the Investor-managed model
66722	Warning	User was added as a client of the calling Advisor. ADVISOR_USER_IDENT value was ignored
67139	Error	Invalid Client FP Access value. This field is only valid for Assistants. Valid values are: READ and READWRITE
67147	Error	The CLIENT_FP_ACCESS value was ignored; it can only be specified if the user has role ASSISTANT and the firm's usage model is Advisor-managed
67155	Error	Assistants may not be created in a firm that uses the Investor-managed model
67305	Error	Only administrators may explicitly set IS_SSO value.
67930	Warning	The new user was not notified because the user has single sign on access.

Sample XML

The following is a sample **<USERMODRQ>** request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <USERMODRQ>
    <USER_IDENT>
      <FIRM_TAG1>67412322</FIRM_TAG1>
    </USER_IDENT>
    <PERSON>
      <FIRM_TAG1>67412323</FIRM_TAG1>
    </PERSON>
  </USERMODRQ>
</DATACONNECTRQ>
```

The following is a corresponding sample **<USERMODRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>
  <USERMODRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <USER_IDENT>
      <FIRM_TAG1>67412322</FIRM_TAG1>
    </USER_IDENT>
    <ID>2300</ID>
  </USERMODRS>
</DATACONNECTRS>
```

Modify Financial Profile

Purpose

Changes the name of a Financial Profile.

Restrictions

None.

Request: <PROFMODRQ>

The <PROFMODRQ> can contain the following:

Tag	Required	Field	Description
<FP_ID> or <USER_IDENT>	√	FinancialProfile.ID	Identifies the profile.
<NAME>	√	FinancialProfile.NAME	The new name of the Financial Profile.

Response: <PROFMODRS>

The <PROFMODRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>		Profile.ID	Unique identifier for the Financial Profile.

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
65555	Error	The <field name> is required
65563	Error	The <field name> cannot exceed <max> characters
66259	Error	The user already has a portfolio with this name

Sample XML

The following is a sample **<PROFMODRQ>** request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <PROFMODRQ>
    <FP_ID>17768</FP_ID>
    <NAME> Mark and Susan Smith</NAME>
  </PROFMODRQ>
</DATACONNECTRQ>
```

The following is a corresponding sample **<PROFMODRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>
  <PROFMODRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <ID>17768</ID>
  </PROFMODRS>
```

Add Portfolio

Purpose

Adds a new Portfolio for a User.

Restrictions

None.

Request: <PORTADDRQ>

The <PORTADDRQ> can contain the following:

Tag	Required	Field	Description
<FP_ID> or <USER_IDENT>	√	FinancialProfile.ID	Identifies the Profile to which the Portfolio should be added. The Profile can be identified directly by internal ID (FP_ID) or indirectly by identifying the INVESTOR (USER_IDENT) for whom the Financial Profile stores data.
<NAME>	√	Portfolio.NAME	The name of the new Portfolio (case-sensitive). If another Portfolio for this User with the same name already exists, this operation fails.

Response: <PORTADDRS>

The <PORTADDRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>		Portfolio.FP_ID	Financial Profile that contains the Portfolio.
<ID>		Portfolio.ID	Unique identifier for the Portfolio. Provided only if the Portfolio was created.

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
65555	Error	The <field name> is required
65563	Error	The <field name> cannot exceed <max> characters
65939	Error	The requested user was not found
66115	Error	The requested user has been unsubscribed. No further operations can be performed on the user
66259	Error	The user already has a portfolio with this name

Sample XML

The following is a sample **<PORTADDRQ>** request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <PORTADDRQ>
    <FP_ID>2300</FP_ID>
    <NAME>Retirement</NAME>
  </PORTADDRQ>
</DATACONNECTRQ>
```

The following is a corresponding sample **<PORTADDRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>
  <PORTADDRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <FP_ID>2300</FP_ID>
    <ID>1003</ID>
  </PORTADDRS>
</DATACONNECTRS>
```

Modify Portfolio

Purpose

Changes the name of a Portfolio.

Restrictions

None.

Request: <PORTMODRQ>

The <PORTMODRQ> can contain the following:

Tag	Required	Field	Description
<FP_ID> or <USER_IDENT>	√	FinancialProfile.ID	Financial Profile that contains the Portfolio.
<ID>	√	Portfolio.ID	Portfolio to be modified.
<NAME>	√	Portfolio.NAME	New name of the Portfolio (case-sensitive)

Response: <PORTMODRS>

The <PORTMODRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>		Portfolio.FP_ID	Financial Profile that contains the Portfolio.
<ID>		Portfolio.ID	Portfolio to be modified

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
65555	Error	The <field name> is required
65563	Error	The <field name> cannot exceed <max> characters
65779	Error	The <field name> is outside the valid range valid range of 0 to 999999999999999999
66099	Error	The portfolio could not be found
66107	Error	The requested user was not found
66123	Error	The portfolio has been archived and cannot be modified
66259	Error	The user already has a portfolio with this name

Sample XML

The following is a sample <PORTMODRQ> request:

```

<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <PORTMODRQ>
    <FP_ID>2300</FP_ID>
    <ID>1003</ID>
    <NAME>John's Retirement</NAME>
  </PORTMODRQ>
</DATACONNECTRQ>

```

```
</PORTMODRQ>  
</DATACONNECTRQ>
```

The following is a corresponding sample **<PORTMODRS>** response:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>...</LOGINRS>  
  <PORTMODRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>  
    </STATUS>  
    <FP_ID>2300</FP_ID>  
    <ID>1003</ID>  
  </PORTMODRS>  
</DATACONNECTRS>
```

Delete Portfolio

Purpose

Deletes a Portfolio. This operation deletes the Portfolio and all other objects (Accounts, Holdings, Transactions) contained in the Portfolio. This operation is permanent. Once the action is complete, the Portfolio and its subordinate objects cannot be restored.

Restrictions

None.

Request: <PORTDELREQ>

The <PORTDELREQ> can contain the following:

Tag	Required	Field	Description
<FP_ID> or <USER_IDENT>	√	FinancialProfile.ID	Internal ID for the Financial Profile that contains the Portfolio.
<ID>	√	Portfolio.ID	Internal ID for the Portfolio to be deleted.

Response: <PORTDELRSP>

The <PORTDELRSP> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>		Portfolio.FP_ID	
<ID>		Portfolio.ID	

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
65555	Error	The <field name> is required
65779	Error	The <field name> is outside the valid range valid range of 0 to 9999999999999999999
66099	Error	The portfolio could not be found
66107	Error	The requested user was not found

Sample XML

The following is a sample **<PORTDELREQ>** request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  <PORTDELREQ>  
    <FP_ID>2300</FP_ID>  
    <ID>1003</ID>  
  </PORTDELREQ>  
</DATACONNECTRQ>
```

The following is a corresponding sample **<PORTDELRS>** response:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>...</LOGINRS>  
  <PORTDELRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>  
    </STATUS>  
    <FP_ID>2300</FP_ID>  
    <ID>1003</ID>  
  </PORTDELRS>  
</DATACONNECTRS>
```

Add Account Credential

Purpose

Adds a new set of Account Credentials for a Financial Institution to a Financial Profile.

Restrictions

1. In select cases, access to the financial institution may be restricted so that only firms that are licensed to access it can do so. If the firm is not licensed for the FI, the response will be an error message.
2. Only some Financial Services support the AUTO_MANAGE capability. AUTO_MANAGE only applies if SUP_AAM is set to supported for the FI and auto-managing of credentials is enabled for the firm. If both are not true, an error will occur when AUTO_MANAGE is set to 1 (true).
 - DATA_BASIS is only included if AUTO_MANGAGE is set to 1 (true), in which case complex default rules apply. If Auto-Manage is set to 0 (false) or is not included in the request, Data Basis may not be specified. An attempt to specify Data Basis in such cases will result in an error. Additionally:
 - Data Basis can only be specified if the firm allows a choice of Data Basis, otherwise setting it will produce an error.
 - Data Basis can only be set to TRADE if the FI supports trade as indicated by the SUP_TRADE flag; likewise for SETTLEMENT and the FI SUP_SETTLEMENT flag. All Financial Institutions will support at least one type of basis.
 - If Auto-Manage is set to 1 (true), and Data Basis is not specified, then a value for Data Basis will always be set based on the following precedence:
 - If FI allows only one of TRADE or SETTLEMENT, as indicated by the SUP_TRADE and SUP_SETTLEMENT flags on the FI, then the allowed type is used.
 - If FI allows both types of basis, and the firm has an established preferred data basis, the preferred basis is used.
 - If FI allows both types of basis, and firm does not have an established preferred data basis, TRADE is used.
 - When Auto-Manage is set to 1 (true), the Data Basis setting will propagate to all future Account objects created under this Account Credential by the nightly Automatic Account Management (AAM) process.
3. GATHER_LOTS is only included if lot gathering is enabled for the firm and if account credentials are auto-managed. These rules apply:
 - If Auto_Manage is set to 1 (true) but no Gather Lots value is provided, Gather Lots defaults to 0 (false).
 - If Auto_Manage is set to 0 (false), or not included in the request, the Gather Lots flag is set to 0 (false).
 - If Auto_Manage is set to 1 (true) and Gather Lots is also set to 1 (true), then the Gather Lots setting of 1 (true) will propagate to all future Account objects created under this Account Credential by the nightly Automatic Account Management (AAM) process.
 - Even with this flag turned on, tax lot data will not be gathered during the routine nightly data aggregation process. The routine nightly data aggregation process will only retrieve the standard balances, holdings, and transactions data. In order to obtain holding lot data, an Account Update With Tax Lots (ACCTUPDTAXLOTRQ) must be used.

Request: <ACCTCREDADDRQ>

The <ACCTCREDADDRQ> can contain the following:

Tag	Required	Field	Description
<PROFILE_ID> or <USER_IDENT>	√	FinancialProfile.ID	Internal ID for the Financial Profile to which the Account Credential should be added.
<NAME>	√	NAME	Name of the Account Credentials. Must be unique among the set of Account Credential names for this User.
<FI_ID>	*	FI_ID	ID for the Financial Service that the Account Credential accesses.
<FI_REQUEST_NAME>	*	FI_REQUEST_NAME	Used to specify the name of a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this service. See related field FI_REQUEST_URL .
<FI_REQUEST_URL>		FI_REQUEST_URL	Used to specify the URL of the Login page for a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this service. See related field FI_REQUEST_NAME .
<ACCOUNT_LOGIN>		ACCOUNT_LOGIN	Login name used to access the Financial Institution Service.
<ACCOUNT_PIN>		ACCOUNT_PIN	Password (goes with ACCOUNT_LOGIN) used to access the Financial Institution Service.
<ACCOUNT_NEW_PIN>		ACCOUNT_NEW_PIN	Some services require the User to provide a new password on first login, this specifies the new password to use.
<ACCOUNT_SECOND_PIN>		ACCOUNT_SECOND_PIN	Some services require a second password to access the Account at that Financial Institution Service.

<ACCOUNT_LOGIN_2>		ACCOUNT_LOGIN_2	Some services require more than a single piece of account login information to access an Account. This field stores this second piece of account login information.
<AUTO_MANAGE>		BOOLEAN	Defaults to 0 (false), so accounts are not created automatically when new credentials are added. If set to 1 (true) accounts for this credential will be automatically maintained by the nightly aggregation process if the capability is enabled for the firm. If no value is set it defaults to 0 (false) when new credentials are added. See Restrictions for more information.
<DATA_BASIS>		DATA_BASIS	Defaults to null. Can be set to TRADE or SETTLEMENT under specific conditions. All Financial Institutions support at least one type of basis, but DATA_BASIS may only be set if firm allows a choice of Data Basis. If AUTO_MANAGE is set to 0 (false) or is not included, then DATA_BASIS must NOT be included and will default to null. If AUTO_MANAGE is set to 1 (true), then complex rules apply; see Restrictions for more information.
<GATHER_LOTS>		BOOLEAN	Defaults to 0 (false), and is only relevant when AUTO_MANAGE is set to 1 (true) and lot gathering is enabled for the firm. Establishes whether lots are gathered for auto-managed accounts. See Restrictions for more information.
<SQA>		SQA	Can have more than one. Provides a new set of SQA information that is used to create a new SQA object.

<FEEDRQ_NAME>		FEEDRQ_NAME	Feed request name. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
<FEEDRQ_EMAIL>		FEEDRQ_EMAIL	Feed request email. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
<FEEDRQ_FIRM>		FEEDRQ_FIRM	Feed request firm. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
<FEEDRQ_LOGIN>		FEEDRQ_LOGIN	Feed request login. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).

Notes:

* = Only one of <FI_ID> or (<FI_REQUEST_NAME> + <FI_REQUEST_URL>) can be provided. If providing FI request information, <FI_REQUEST_NAME> is required and <FI_REQUEST_URL> is optional. Please see the Document Type Definition for exact syntax.

The <SQA> aggregate contains the following:

Tag	Required	Field	Description
<QUESTION>	√	QUESTION	Security question that must be answered to access the Financial Institution service.
<ANSWER>	√	ANSWER	Answer to security question that is required to access the Financial Institution service.

Response: <ACCTCREDADDRS>

The <ACCTCREDADDRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<PROFILE_ID>		PROFILE_ID	See Account Credential object description.
<ID>		ID	ID for the new Account Credentials. Provided only if the operation was successful.

Errors

Error Code	Type	Error Message
67163	Error	Not licensed to perform operation at restricted financial institution
67331	Error	Automatic account management is not enabled for this firm
67339	Error	Automatic account management is not supported at this FI
67347	Error	Choosing data basis is not enabled for this firm
67355	Error	Tax lot gathering is not enabled for this firm
67363	Error	Tax lot gathering is not supported at this FI
67371	Error	Setting DATA_BASIS is not allowed for an account credential without automatic account management
67379	Error	Setting GATHER_LOTS is not allowed for an account credential without automatic account management
67836	Warning	Automatic account management no longer available for this account credential, therefore credential-level settings for DATA_BASIS and GATHER_LOTS will no longer apply

Sample XML

The following is a sample <ACCTCREDADDRQ> request:

<DATACONNECTRQ>

 <VERSION>VERSION4.0</VERSION>

 <LOGINRQ>...</LOGINRQ>

```
<ACCTCREDADDRQ>
  <USER_IDENT>
    <PERSON_LOGIN_NAME>Investor1</PERSON_LOGIN_NAME>
  </USER_IDENT>
  <NAME>Some FI's Credentials</NAME>
  <FI_REQUEST_NAME>Some FI M0040</FI_REQUEST_NAME>
  <FI_REQUEST_URL>Some FI M0040</FI_REQUEST_URL>
</ACCTCREDADDRQ>
</DATACONNECTRQ>
```

The following is a sample **<ACCTCREDADDRS>** request:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <ACCTCREDADDRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <USER_IDENT>
      <PERSON_LOGIN_NAME>Investor1</PERSON_LOGIN_NAME>
    </USER_IDENT>
    <PROFILE_ID>7637</PROFILE_ID>
    <ID>891</ID>
  </ACCTCREDADDRS>
</DATACONNECTRS>
```

Modify Account Credential

Purpose

Modifies the Account Credentials for a Financial Institution.

Restrictions

1. In select cases, access to the financial institution may be restricted so that only firms that are licensed to access it can do so. If the firm is not licensed for the FI, the response will be an error message.

Behavior

This operation exhibits the following behaviors:

1. If you submit a new SQA to be added to a credential, DataConnect will determine if that SQA already exists by performing an exact match (case-sensitive) search on the SQA Question text. If the SQA already exists, then DataConnect will modify that SQA instead of adding a new SQA.
2. In general, the behavior for Modify Account Credential is the same as for [Add Account Credential](#) unless the Financial Service is changed. Changing the Financial Service may cause other changes.
 - If the AUTO_MANAGE flag is tuned on but the new Financial Service does not support auto manage, the AUTO_MANAGE flag will be turned off.
 - The same is true for GATHER_LOTS and DATA_BASIS. If the new Financial Service does not support them, they will be turned off.
3. There are two main types of authentication used by Financial Institutions: OAuth and Login. A credential is assigned a base type of OAUTH or LOGIN based on the related Financial Institution. For Institutions that use OAuth for authorization, the credential type will be OAUTH. For all other Institutions and for new FI requests the credential type will be LOGIN. You are not allowed to set LOGIN type information (login, password, etc.) on an OAUTH type credential. When moving a credential from one FI to another FI (using this Modify Account Credential operation), there can be side effects as follows:
 - When the operation specifies changing the credential to use an FI that uses OAuth authentication, then all credential data (login, password, security questions and answers, or any previous OAUTH authorization) is discarded and all ACCOUNT_NUMBER_2 data on accounts linked to the credential are removed. This information, once removed, cannot be recovered.
 - In all other cases of changing the FI on a credential, the credential and account information is preserved. If the new FI has different credential requirements from the original FI then the credential may move to an incomplete state.

Request: <ACCTCREDMODRQ>

The <ACCTCREDMODRQ> can contain the following:

Tag	Required	Field	Description
<PROFILE_ID> or <USER_IDENT>	√	FinancialProfile.ID	Internal ID for the Financial Profile that contains the Account Credential.
<ID>	√	ID	ID for the Account Credentials.
<NAME>		NAME	Name of the Account Credentials.
<FI_ID>	*	FI_ID	ID for the Financial Service that the Account Credential accesses.
<FI_REQUEST_NAME>	*	FI_REQUEST_NAME	Used to specify name of a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this service. See related field FI_REQUEST_URL .
<FI_REQUEST_URL>		FI_REQUEST_URL	Used to specify the URL of the Login page for a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this service. See related field FI_REQUEST_NAME .
<ACCOUNT_LOGIN>		ACCOUNT_LOGIN	Login name used to access the Financial Institution Service.
<ACCOUNT_PIN>		ACCOUNT_PIN	Password (goes with ACCOUNT_LOGIN) used to access the Financial Institution Service.
<ACCOUNT_NEW_PIN>		ACCOUNT_NEW_PIN	Some services require the User to provide a new password on first login, this specifies the new password to use.
<ACCOUNT_SECOND_PIN>		ACCOUNT_SECOND_PIN	Some services require a second password to access the Account at that Financial Institution Service.

<ACCOUNT_LOGIN_2>		ACCOUNT_LOGIN_2	Some services require more than a single piece of account login information to access an Account. This field stores this second piece of account login information.
<AUTO_MANAGE>		BOOLEAN	Can only be set to 1 (true) when automatic account management is enabled for the firm. When set to 0 (false), accounts are not created automatically for the credential. If set to 1 (true) accounts for this credential will be automatically maintained by the nightly aggregation process. If no value is set it defaults to 0 (false). Complex rules apply; see Behavior for more information.
<DATA_BASIS>		DATA_BASIS	DATA_BASIS may only be set if firm allows a choice of Data Basis and AUTO_MANAGE is set to 1 (true). Can be set to TRADE or SETTLEMENT under those conditions. Complex rules apply; see Behavior for more information.
<GATHER_LOTS>		BOOLEAN	Only relevant when AUTO_MANAGE is set to 1 (true) and lot gathering is enabled for the firm and supported by the financial institution. Establishes whether lots are gathered for auto-managed accounts. Complex rules apply; see Behavior for more information.
<SQA>		SQA	Can have more than one. Causes the question and answer in a SQA object associated with this Account Credential to be changed. If no SQA object exists, a new one is created.

<FEEDRQ_NAME>		FEEDRQ_NAME	Feed request name. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are “BAA Feeds”. Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
<FEEDRQ_EMAIL>		FEEDRQ_EMAIL	Feed request email. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are “BAA Feeds”. Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
<FEEDRQ_FIRM>		FEEDRQ_FIRM	Feed request firm. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are “BAA Feeds”. Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
<FEEDRQ_LOGIN>		FEEDRQ_LOGIN	Feed request login. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are “BAA Feeds”. Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).

Notes:

* = Only one of <FI_ID> or (<FI_REQUEST_NAME> + <FI_REQUEST_URL>) can be provided. If providing FI request information, <FI_REQUEST_NAME> is required and <FI_REQUEST_URL> is optional. Providing <FI_ID> causes <FI_REQUEST_NAME> to be set to null and vice versa. Please see the Document Type Definition for exact syntax.

The **<SQA>** aggregate can contain the following:

Tag	Required	Field	Description
<ID>	*	ID	Unique numeric ID for this Security Question and Answer.
<QUESTION>		QUESTION	Security question that must be answered to access the Financial Institution service.
<ANSWER>		ANSWER	Answer to security question that is required to access the Financial Institution service.

Notes:

* = If **<ID>** is not provided a new SQA is created. When creating an SQA both the question and answer must be provided. If only **<ID>** is provided then the SQA is deleted.

Response: <ACCTCREDMODRS>

The **<ACCTCREDMODRS>** response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<PROFILE_ID>	√	PROFILE_ID	See Account Credential object description.
<ID>		ID	ID for the Account Credentials.

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
66963	Error	Error setting data basis
67163	Error	Not licensed to perform operation at restricted financial institution
67331	Error	Automatic account management is not enabled for this firm
67339	Error	Automatic account management is not supported at this FI
67347	Error	Choosing data basis is not enabled for this firm
67355	Error	Tax lot gathering is not enabled for this firm
67363	Error	Tax lot gathering is not supported at this FI
67371	Error	Setting DATA_BASIS is not allowed for an account credential without automatic account management
67379	Error	Setting GATHER_LOTS is not allowed for an account credential without automatic account management
67836	Warning	Automatic account management no longer available for this account credential, therefore credential-level settings for DATA_BASIS and GATHER_LOTS will no longer apply

Sample XML

The following is a sample **<ACCTCREDMODRQ>** request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  <ACCTCREDMODRQ>  
    <USER_IDENT>  
      <PERSON_LOGIN_NAME>Investor1</PERSON_LOGIN_NAME>  
    </USER_IDENT>  
    <ID>14998</ID>  
    <ACCOUNT_LOGIN>Some FI M0040</ACCOUNT_LOGIN>  
  </ACCTCREDMODRQ>  
</DATACONNECTRQ>
```

The following is a sample **<ACCTCREDMODRS>** request:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>  
  <STATUS>  
  <ERRCODE>0</ERRCODE>  
  <ERRMSG>Success</ERRMSG>  
  </STATUS>  
  </LOGINRS>  
<ACCTCREDMODRS>  
  <STATUS>  
  <ERRCODE>0</ERRCODE>  
  <ERRMSG>Success</ERRMSG>
```

```
</STATUS>  
<USER_IDENT>  
<PERSON_LOGIN_NAME>Investor1</PERSON_LOGIN_NAME>  
</USER_IDENT>  
<PROFILE_ID>7637</PROFILE_ID>  
<ID>14998</ID>  
</ACCTCREDMODRS>  
</DATACONNECTRS>
```

Delete Account Credential

Purpose

Deletes the Account Credentials for a Financial Institution. If accounts exist that refer to these credentials, the caller has the option to delete those accounts or to leave the accounts in place without the credential reference.

Restrictions

None.

Request: <ACCTCREDELREQ>

The <ACCTCREDELREQ> can contain the following:

Tag	Required	Field	Description
<PROFILE_ID> or <USER_IDENT>	√	FinancialProfile.ID	Internal ID for the Financial Profile that contains the Account Credential.
<ID>	√	ID	ID of the Account Credential to be deleted.
<DELETE_ACCOUNTS>			When this tag is included, all accounts associated with the credential are deleted. When it is omitted, the reference is removed and the accounts are left without credentials.

Response: <ACCTCREDELRS>

The <ACCTCREDELRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<PROFILE_ID>		FP_ID	See Account Credential object description.
<ID>		ID	ID for the Account Credential.

Errors

This operation may return the following in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
65779	Error	The <field name> is outside the valid range valid range of 0 to 999999999999999999
65939	Error	The requested user was not found
66099	Error	The <object name> could not be found
67299	Error	The Account Credential could not be deleted because an Account associated with the Account Credential could not be deleted

Sample XML

The following is a sample **<ACCTCREDELREQ>** request:

<DATACONNECTRQ>

```
<VERSION>VERSION4.0</VERSION>
<LOGINRQ>...</LOGINRQ>
<ACCTCREDELREQ>
  <USER_IDENT>
    <PERSON_LOGIN_NAME>Investor1</PERSON_LOGIN_NAME>
  </USER_IDENT>
  <ID>77123</ID>
</ACCTCREDELREQ>
```

</DATACONNECTRQ>

The following is a sample **<ACCTCREDELRS>** response:

<DATACONNECTRS>

```
<VERSION>VERSION4.0</VERSION>
<LOGINRS>
```

```
<STATUS>  
<ERRCODE>0</ERRCODE>  
<ERRMSG>Success</ERRMSG>  
</STATUS>  
</LOGINRS>
```

```
<ACCTCREDDELRS>  
  <STATUS>  
    <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>  
  </STATUS>  
  <USER_IDENT>  
    <PERSON_LOGIN_NAME>Investor1</PERSON_LOGIN_NAME>  
  </USER_IDENT>  
  <PROFILE_ID>7637</PROFILE_ID>  
  <ID>77123</ID>  
</ACCTCREDDELRS>  
  
</DATACONNECTRS>
```

Add Account

Purpose

Adds a new Account for a User. Optionally, the caller can request that an on-demand Update (retrieve data from the FI) for the Account be performed.

Restrictions

1. In select cases, access to the financial institution may be restricted so that only firms that are licensed to access it can do so. If the firm is not licensed for the FI, the response will be an error message.
2. Only online Accounts can be added using this function. A Financial Service ID or Financial Service request (for support) must be provided when the Account is created.
3. An Account must have all required online Access Credentials for a Financial Service before data can be gathered from that service. However, since various individuals may provide these credentials at different times (e.g., the Advisor sets up the Account and enters the account number, while the Investor enters the Login and password at a later date), the **Add Account** function does not require any credential fields to successfully create the Account.
4. An Account must have all required online Access Credentials for a Financial Service before a Financial Services support request is submitted for that Account. (E.g., when a User requests support for a Financial Service not currently supported by ByAllAccounts, both the identifying information for the Financial Service (URL and optionally, name) as well as online Access Credentials for the Account at that service must be provided.)
5. GATHER_LOTS enables the collection of tax lot data via the Account Update with Tax Lots operation (ACCTUPDTAXLOTRQ). It can be set to 1 (true) or 0 (false)
 - The gather lots flag cannot be set to TRUE if the firm does not have the option to “allow tax lot data” turned on.
 - The gather lots field can only be set to TRUE if the Financial Institution supports tax lot data (see SUP_TAX_LOT flag on the Financial Institution).
 - If gather lots is not explicitly stated in the request, then it will always default to FALSE.
6. DATA_BASIS can be set to TRADE or SETTLEMENT. The data basis field indicates whether trade-based or settlement-based data should be gathered for the Account. It has the following behavior:
 - The data basis cannot be explicitly set if the firm does not have the option to “choose data basis” turned on.
 - The data basis can only be set to TRADE if the financial institution supports trade (see SUP_TRADE flag on Financial Institution) and it can only be set to SETTLEMENT if the financial institution supports settlement-based data (see SUP_SETTLE flag on Financial Institution). Financial Institutions will always support at least one of trade and settlement.
 - If the basis is not explicitly stated in the request, then the basis will be set to a default based on the following rules:
 - 1) If the Financial Institution only allows one type of basis (TRADE or SETTLEMENT but not both), then the allowed type of basis is used.
 - 2) If the Financial Institution offers both types of basis, and the firm does have a preferred data basis established, then the preferred data basis is used.

- 3) If the Financial Institution allows both types of basis, but the firm does not have a preferred data basis established (and/or does not allow choice of basis), the basis will be set to TRADE.

Request: <ACCTADDRQ>

The <ACCTADDRQ> can contain the following:

Tag	Required	Field	Description
<DO_UPDATE/>			Empty tag. If present, DO_UPDATE submits an Update Account from FI request to retrieve data for the Account from the FI. The Update Account from FI is only submitted if the Account is successfully created and sufficient credentials are available to access the Account at the FI. The Add Account operation does not wait for the Update to complete before returning to the User.
<FP_ID> or <USER_IDENT>	√	FinancialProfile.ID	Internal ID for the Financial Profile to which the Account should be added.
<NAME>	√	NAME	Name of the new Account (case-sensitive). Must be unique among the set of Account names for this User.
<PORTFOLIO_ID>		PORTFOLIO_ID	ID for the parent Portfolio for this Account.
<FI_ID> or <AC_ID> or <ACCOUNT_CREDENTIAL>	*	AC_ID	Specifies the credentials to be used to access the Account at the Financial Institution. Several options are available: <ul style="list-style-type: none"> ▪ FI_ID causes an ACCOUNT_CREDENTIAL object for that FI to be used for this Account. If more than one such object exists, an error occurs. ▪ AC_ID identifies the specific Account Credential object to be used. ▪ <ACCOUNT_CREDENTIAL> provides a new set of credential information that is used to create a new ACCOUNT_CREDENTIAL object. Refer to <ACCOUNT_CREDENTIAL> aggregate for more. If the new credentials cannot be created, the Add Account operation fails. <ACCOUNT_CREDENTIAL> aggregate

<ACCTADDRQ> (continued):

Tag	Required	Field	Description
<ACCOUNT_NUMBER>		ACCOUNT_NUMBER	
<ACCOUNT_NUMBER_2>		ACCOUNT_NUMBER_2	
<CAPTIVE>		CAPTIVE	If omitted, is set to a default value.
<GATHER_LOTS>		GATHER_LOTS	Defaults to 0 (false), and can only be set to 1 (true) when lot gathering is enabled for the firm and allowed by the financial institution. Establishes whether lots are gathered for the account. See Restrictions for more information.
<DATA_BASIS>		DATA_BASIS	Defaults to null. Can be set to TRADE or SETTLEMENT under specific conditions. See Restrictions for more information.
<DISCOVERED_ACCOUNT_ID>		DISCOVERED_ACCOUNT_ID	<ul style="list-style-type: none"> ▪ Optional, but recommended to pass this value as a best practice. When specified it is used to update the following fields in the Account with corresponding fields from the DiscoveredAccount Object ▪ TAX_ID (which maps to FI_SUPPLIED_ACCOUNT_TAXID) ▪ FI_SUPPLIED_ACCOUNT_TYPE ▪ ACCOUNT_TYPE ▪ FI_SUPPLIED_ADDRESS_STREET ▪ FI_SUPPLIED_ADDRESS_LINE2 ▪ FI_SUPPLIED_ADDRESS_LINE3 ▪ FI_SUPPLIED_ADDRESS_LINE4 ▪ FI_SUPPLIED_ADDRESS_LINE5 ▪ FI_SUPPLIED_ADDRESS_LINE6 ▪ FI_SUPPLIED_ADDRESS_CITY ▪ FI_SUPPLIED_ADDRESS_STATE ▪ FI_SUPPLIED_ADDRESS_ZIP_CODE ▪ FI_SUPPLIED_CLIENT_PHONE ▪ FI_SUPPLIED_CLIENT_DOB ▪ FI_SUPPLIED_CLIENT_EMAIL ▪ FI_SUPPLIED_REP_ID ▪ FI_SUPPLIED_FIRM_ID ▪ FI_SUPPLIED_ACCOUNT_TAXID

			<ul style="list-style-type: none"> ▪ FI_SUPPLIED_ACCOUNT_NAME ▪ FI_SUPPLIED_ACCOUNT_TITLE ▪ FI_SUPPLIED_CLIENT_FIRST ▪ FI_SUPPLIED_CLIENT_MIDDLE ▪ FI_SUPPLIED_CLIENT_LAST
<EXTERNAL_SERVICE_LEVEL>		CHAR20	The value for this field may be set in the request to Positional or Transactional. When the value is not specified, upon account creation if the authenticated user is an Investor the value defaults to Positional. If the authenticated user is not an Investor the value defaults to Transactional.

Notes:

- * = Only one of the specified tags may be submitted.
- The account's **CAPTIVE** field value defaults as follows:
 - For Accounts with no **FI_ID**: Non-captive.
 - For Accounts with an **FI_ID**: If the specified Financial Service is a captive FI for the User's firm, the Account is set to captive, otherwise it is set to non-captive.
- If no **PORTFOLIO_ID** is given, the Account is inserted into the first Portfolio created for this profile. If no Portfolio exists, a Portfolio with the name 'My Portfolio' is created and the Account is inserted into that Portfolio.

The <ACCOUNT_CREDENTIAL> aggregate can contain the following:

Tag	Required	Field	Description
<CREDENTIAL_NAME>		NAME	Sets Credential Name on the associated Account Credential. Defaults to the Account Name if not explicitly included in the Account Add Request.
<FI_ID>	*	FI_ID	ID for the Financial Service that the Account Credential accesses.
<FI_REQUEST_NAME>	*	FI_REQUEST_NAME	Used to specify the name of a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this service. See related field FI_REQUEST_URL .
<FI_REQUEST_URL>	*	FI_REQUEST_URL	Used to specify the URL of the Login page for a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this service. See related field FI_REQUEST_NAME .
<ACCOUNT_LOGIN>		ACCOUNT_LOGIN	Login name used to access the Financial Institution Service.
<ACCOUNT_PIN>		ACCOUNT_PIN	Password used to access the Financial Institution Service (with ACCOUNT_LOGIN).
<ACCOUNT_NEW_PIN>		ACCOUNT_NEW_PIN	Some services require the User to provide a new password on first login, this specifies the new password to use.
<ACCOUNT_SECOND_PIN>		ACCOUNT_SECOND_PIN	Some services require a second password to access the Account at that Financial Institution Service.
<ACCOUNT_LOGIN_2>		ACCOUNT_LOGIN_2	Some services require more than a single piece of account login information to access an Account. This field stores this second piece of account login information.

<SQA>		SQA	Can have more than one. Provides a new set of SQA information that is used to create a new SQA object.
FEEDRQ_NAME		CHAR128	Feed request name. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are “BAA Feeds”. Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_EMAIL		CHAR64	Feed request email. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are “BAA Feeds”. Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_FIRM		CHAR128	Feed request firm. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are “BAA Feeds”. Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_LOGIN		CHAR64	Feed request login. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are “BAA Feeds”. Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).

Notes:

* = Only one of <FI_ID> or (<FI_REQUEST_NAME> + <FI_REQUEST_URL>) can be provided. If providing FI request information, <FI_REQUEST_NAME> is required and <FI_REQUEST_URL> is optional. Please see the Document Type Definition for exact syntax.

The **<SQA>** aggregate can contain the following:

Tag	Required	Field	Description
<QUESTION>	√	QUESTION	Security question that must be answered to access the Financial Institution service.
<ANSWER>	√	ANSWER	Answer to security question that is required to access the Financial Institution service.

Response: <ACCTADDRS>

The **<ACCTADDRS>** response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>	√	FP_ID	See ACCOUNT object description.
<ID>		ID	ID for the new Account. Provided only if the Add Account operation was successful.

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
65555	Error	The <field name> is required
65563	Error	The <field name> cannot exceed <max> characters
65779	Error	The <field name> is outside the valid range valid range of 0 to 9999999999999999
65787	Error	Boolean values must be 0 or 1
65939	Error	The requested user was not found
66099	Error	The <object name> could not be found
66123	Error	The portfolio has been archived and cannot be modified
66267	Error	The user already has an account with this name
66747	Error	Only one of the financial institution ID or its name can be specified
66763	Error	The financial institution URL cannot be specified since its ID has been specified. Financial institution names and URLs are used for institution support requests only.
66802	Warning	The financial institution does not use the <field name> field. The field was not saved.
66939	Error	The account could not be added because the FI does not support trade-based data
66947	Error	The account could not be added because the FI does not support settlement-based data

66963	Error	Error setting data basis
67163	Error	Not licensed to perform operation at restricted financial institution
67347	Error	Choosing data basis is not enabled for this firm
67355	Error	Tax lot gathering is not enabled for this firm
67363	Error	Tax lot gathering is not supported at this FI
67891	Error	Discovered account could not be found
67899	Error	Account to be added does not match Discovered Account for financial profile
67907	Error	Account to be added does not match Discovered Account for credential
67915	Error	Account to be added does not match Discovered Account for account number
67923	Error	Account to be added does not match Discovered Account for account number 2
68027	Error	Invalid value for EXTERNAL_SERVICE_LEVEL: Must be one of Positional, Transactional
68067	Error	The feed onboarding request has been submitted and cannot be changed.
68075	Error	Feed onboarding requests are not supported at this FI.

Sample XML

The following is a sample **<ACCTADDRQ>** request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
    <ACCTADDRQ>  
      <FP_ID>2300</FP_ID>  
      <NAME>Mark's IRA</NAME>  
      <PORTFOLIO_ID>99</PORTFOLIO_ID>  
      <FI_ID>200</FI_ID>  
      <ACCOUNT_NUMBER>999888777</ACCOUNT_NUMBER>  
    </ACCTADDRQ>  
  </DATACONNECTRQ>
```

The following is a corresponding sample **<ACCTADDRS>** response:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>...</LOGINRS>  
  <ACCTADDRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>  
    </STATUS>  
    <FP_ID>2300</FP_ID>  
    <ID>1476</ID>  
  </ACCTADDRS>  
</DATACONNECTRS>
```

Modify Account

Purpose

Changes information for an Account.

1. An Account must have all required online Access Credentials for a Financial Service before data can be gathered from that service. However, since various individuals may provide these credentials at different times (e.g., the Advisor sets up the Account and enters the account number, while the Investor enters the Login and password at a later date), the **Modify Account** function does not require any credential fields to successfully modify the Account.
2. An Account must have all required online Access Credentials for a Financial Service before a Financial Services support request is submitted for that Account (e.g., when a User requests support for a Financial Service not currently supported by ByAllAccounts, both the identifying information for the Financial Service (URL and optionally, name) as well as online Access Credentials for the Account at that service must be provided.).

Restrictions

1. In select cases, access to the financial institution may be restricted so that only firms that are licensed to access it can do so. If the firm is not licensed for the FI, the response will be an error message.
2. Changes to the credential should be made on the credential itself, not on the account. Modifying a credential on the account to be a credential for another institution may modify settings on the credential including auto management, data basis, and gather lots settings depending on what is supported by the institution.

Behavior

This operation exhibits the following behaviors:

1. If you submit a new SQA to be added to a credential, DataConnect will determine if that SQA already exists by performing an exact match (case-sensitive) search on the SQA Question text. If the SQA already exists, then DataConnect will modify that SQA instead of adding a new SQA.
2. In general, the behavior of Modify Account is the same as for [Add Account](#) unless the Financial Service is changed. Changing the Financial Service may cause other changes.
 - If the GATHER_LOTS flag is tuned on but the new Financial Service does not support lot gathering, the GATHER_LOTS flag will be turned off.
 - The same is true for DATA_BASIS. If the new Financial Service does not support it, it will be turned off.
3. There are two main types of authentication used by Financial Institutions: OAuth and Login. A credential is assigned a base type of OAUTH or LOGIN based on the related Financial Institution. For Institutions that use OAuth for authorization, the credential type will be OAUTH. For all other Institutions and for new FI requests the credential type will be LOGIN. You are not allowed to set LOGIN type information (login, password, etc.) on an OAUTH type credential. When moving a credential from one FI to another FI (using this Modify Account operation), there can be side effects as follows:

- When the operation specifies changing the credential to use an FI that uses OAuth authentication, then all credential data (login, password, security questions and answers, or any previous OAUTH authorization) is discarded and all ACCOUNT_NUMBER_2 data on accounts linked to the credential are removed. This information, once removed, cannot be recovered.
- In all other cases of changing the FI on a credential, the credential and account information is preserved. If the new FI has different credential requirements from the original FI then the credential may move to an incomplete state.

Request: <ACCTMODRQ>

The <ACCTMODRQ> can contain the following:

Tag	Required	Field	Description
<DO_UPDATE/>			Empty tag. If present, DO_UPDATE submits an Update Account from FI request to retrieve data for the Account from the FI. The Update Account from FI is only submitted if the Account is successfully modified and sufficient credentials are available to access the Account at the FI. The Modify Account operation does not wait for the Update to complete before returning to the User.
<FP_ID> or <USER_IDENT>	√	FP_ID	Financial Profile that contains the Account. The FP_ID field cannot be modified.
<ID>	√	ID	Identifies the Account. The ID field cannot be modified.
<NAME>		NAME	
<PORTFOLIO_ID>		PORTFOLIO_ID	
<FI_ID> or <AC_ID> or <ACCOUNT_CREDENTIAL>		AC_ID	Specifies the credentials to be used to access the Account at the Financial Institution. Several options are available: <ul style="list-style-type: none"> ▪ FI_ID causes an existing ACCOUNT_CREDENTIAL object for that FI to be used for this Account. ▪ AC_ID identifies the specific Account Credential object to be used. ▪ <ACCOUNT_CREDENTIAL> causes the credentials in the Account Credential object associated with this Account to be changed. If no Account Credential object exists, a new one is created. Refer to <ACCOUNT_CREDENTIAL> aggregate for more. If the credentials cannot be modified/created, the Modify Account operation fails.

<ACCTMODRQ> continued

Tag	Required	Field	Description
<ACCOUNT_NUMBER>		ACCOUNT_NUMBER	
<ACCOUNT_NUMBER_2>		ACCOUNT_NUMBER_2	
<CAPTIVE>		CAPTIVE	
<GATHER_LOTS>		GATHER_LOTS	Establishes whether lots are gathered for the account. Can only be set to 1 (true) when lot gathering is enabled for the firm and allowed by the financial institution. See Restrictions for more information.
<DATA_BASIS>		DATA_BASIS	Explicitly set the data basis to TRADE or SETTLEMENT Can only be set when firm has the option to choose data basis turned on. Likewise, can only be set to TRADE if the financial institution supports trade (see SUP_TRADE flag on Financial Institution) and it can only be set to SETTLEMENT if the financial institution supports settlement-based data (see SUP_SETTLE flag on Financial Institution). Financial Institutions will always support at least one of the two. See Restrictions for more information.
<EXTERNAL_SERVICE_LEVEL>		CHAR20	The field value may be set in the request to Positional or Transactional.

The <ACCOUNT_CREDENTIAL> aggregate can contain the following:

Tag	Required	Field	DESCRIPTION
<CREDENTIAL_NAME>		NAME	Used to modify the credential name when modifying the account.
<FI_ID>	*	FI_ID	ID for the Financial Service that the Account Credential accesses.
<FI_REQUEST_NAME>	*	FI_REQUEST_NAME	Used to specify the name of a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this

			service. See related field FI_REQUEST_URL .
<FI_REQUEST_URL>	*	FI_REQUEST_URL	Used to specify the URL of the Login page for a Financial Institution Service that ByAllAccounts does not currently support. When all Account Credentials are provided, a request is made to add support for this service. See related field FI_REQUEST_NAME .
<ACCOUNT_LOGIN>		ACCOUNT_LOGIN	Login name used to access the Financial Institution Service.
<ACCOUNT_PIN>		ACCOUNT_PIN	Password (goes with ACCOUNT_LOGIN) used to access the Financial Service.
<ACCOUNT_NEW_PIN>		ACCOUNT_NEW_PIN	Some services require the User to provide a new password on first login. This specifies the new password to use.
<ACCOUNT_SECOND_PIN>		ACCOUNT_SECOND_PIN	Some services require a second password to access the Account at that Financial Service.
<ACCOUNT_LOGIN_2>		ACCOUNT_LOGIN_2	Some services require more than a single piece of account login information to access an account. This field stores this second piece of account login information.
<SQA>		SQA	Can have more than one. Causes the question and answer in a SQA object associated with this Account Credential to be changed. If no SQA object exists, a new one is created.
FEEDRQ_NAME		CHAR128	Feed request name. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_EMAIL		CHAR64	Feed request email. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding

			FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_FIRM		CHAR128	Feed request firm. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).
FEEDRQ_LOGIN		CHAR64	Feed request login. One of four input fields (prefixed with FEEDRQ) that, as a set, support AccountView in onboarding FIs that are "BAA Feeds". Providing the set in its entirety triggers an informational email that is sent to BAA Custodial Operations (CustOps).

Notes:

* = Only one of <FI_ID> or (<FI_REQUEST_NAME> + <FI_REQUEST_URL>) can be provided. If providing FI request information, <FI_REQUEST_NAME> is required and <FI_REQUEST_URL> is optional. Please see the Document Type Definition for exact syntax.

The <SQA> aggregate can contain the following:

Tag	Required	Field	Description
<ID>	*	ID	Unique numeric ID for this Security Question and Answer.
<QUESTION>		QUESTION	Security question that must be answered to access the Financial Institution service.
<ANSWER>		ANSWER	Answer to security question that is required to access the Financial Institution service.

Notes:

* = If <ID> is not provided a new SQA is created. When creating an SQA both the question and answer must be provided. If only <ID> is provided then the SQA is deleted.

Response: <ACCTMODRS>

The <ACCTMODRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>		FP_ID	See ACCOUNT object description.
<ID>	√	ID	See ACCOUNT object description.

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
65555	Error	The <field name> is required
65563	Error	The <field name> cannot exceed <max> characters
65779	Error	The <field name> is outside the valid range valid range of 0 to 999999999999999999
65787	Error	Boolean values must be 0 or 1
65939	Error	The requested user was not found
66099	Error	The <object name> could not be found
66123	Error	The <object name> has been archived and cannot be modified
66267	Error	The user already has an account with this name
66747	Error	Only one of the financial institution ID or its name can be specified
66763	Error	The financial institution URL cannot be specified since its ID has been specified. Financial institution names and URLs are used for institution support requests only.
66802	Warning	The financial institution does not use the <field name> field. The field was not saved.
66963	Error	Error setting data basis
67163	Error	Not licensed to perform operation at restricted financial institution
67347	Error	Choosing data basis is not enabled for this firm
67355	Error	Tax lot gathering is not enabled for this firm
67363	Error	Tax lot gathering is not supported at this FI
68027	Error	Invalid value for EXTERNAL_SERVICE_LEVEL: Must be one of Positional, Transactional
68067	Error	The feed onboarding request has been submitted and cannot be changed.
68075	Error	Feed onboarding requests are not supported at this FI.

Sample XML

The following is a sample **<ACCTMODRQ>** request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <ACCTMODRQ>
    <FP_ID>2300</FP_ID>
    <ID>872</ID>
    <ACCOUNT_CREDENTIAL>
      <ACCOUNT_PIN>jasmine123</ACCOUNT_PIN>
    </ACCOUNT_CREDENTIAL>
  </ACCTMODRQ>
</DATACONNECTRQ>
```

The following is a corresponding sample **<ACCTMODRS>** response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>
  <ACCTMODRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <FP_ID>2300</FP_ID>
    <ID>872</ID>
  </ACCTMODRS>
</DATACONNECTRS>
```

Delete Account

Purpose

Deletes an Account. This operation deletes the Account and all other objects (Holdings, Transactions) contained in the Account. This operation is permanent. Once the action is complete, the Account and its subordinate objects cannot be restored.

Restrictions

None.

Request: <ACCTDELREQ>

The <ACCTDELREQ> can contain the following:

Tag	Required	Field	Description
<FP_ID> or <USER_IDENT>	√	FP_ID	Financial Profile that contains the Account.
<ID>	√	ID	Account to be deleted.
<DELETE_CRED_IF_LAST>			Flag on the request. It is either included (as an empty tag) or excluded to indicate true or false. If the last account related to the credential is deleted, delete the credential.

Response: <ACCTDELRS>

The <ACCTDELRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>	√	FP_ID	See ACCOUNT object description.
<ID>	√	ID	See ACCOUNT object description.

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
65779	Error	The <field name> is outside the valid range valid range of 0 to 999999999999999999
65939	Error	The requested user was not found
66099	Error	The <object name> could not be found

Sample XML

The following is a sample **<ACCTDELREQ>** request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  <ACCTDELREQ>  
    <FP_ID>2300</FP_ID>  
    <ID>1003</ID>  
  </ACCTDELREQ>  
</DATACONNECTRQ>
```

The following is a corresponding sample **<ACCTDELRS>** response:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>...</LOGINRS>  
  <ACCTDELRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>  
    </STATUS>  
    <FP_ID>2300</FP_ID>  
    <ID>1003</ID>  
  </ACCTDELRS>  
</DATACONNECTRS>
```

Aggregation Operations

This section defines operations that direct DataConnect to either verify Account Credentials or gather account data from a Financial Service. Data is gathered from the Financial Services and stored in Account, Holding, and Transaction objects. Once stored, the data can be retrieved from DataConnect. See the [Retrieval Operations section](#) for more information.

Test Account Credential

Purpose

The Test Account Credential operation is an asynchronous operation that tests whether an Account Credential authenticates successfully at the Financial Institution. The Test Account Credential operation can only be executed on a single Account Credential.

The Test Account Credential may also include an In-Session Activation Code (ISAC) interaction. The ISAC interaction may occur in Test Account also, and it is described in more detail in [In-Session Activation Codes](#).

The Test Account Credential will include an OAuth interaction if the Financial Institution for the Account Credential uses OAuth for authentication (instead of a login/password/SQA/ISAC). To support this requirement, the Test Account Credential operation will require an OAuth interaction. It is recommended that applications that use DataConnect provide a user interface to accept and handle the OAuth interaction. Not providing this capability will limit the accounts that can be configured successfully using DataConnect.

The OAuth interaction is initiated when the calling application includes the START_OAUTH element in the Test Account Credential initial request. Any request to Test Account Credential for an OAuth FI must include the START_OAUTH element. The first response to the Test Account Credential operation will return two data elements: OAUTH_MICROSITE_URL and OAUTH_EXP. The calling application must launch the OAUTH_MICROSITE_URL for the user, and the user must complete that interaction before the OAUTH_EXP expiration time. At the financial institution's microsite, the user will authorize accounts for aggregator access typically by selecting one or more accounts and choosing other options such as "and authorize all future accounts". The microsite options will vary by financial institution. Upon completion of the authorization, the aggregation service will be called back directly with a temporary authorization token that it will then convert to a permanent authorization token.

While the user is interacting with the OAUTH_MICROSITE_URL, Claim Data requests for this Test Credential operation will produce the same OAUTH_MICROSITE_URL and OAUTH_EXP data elements and the successful error message "Operation paused, awaiting OAuth". The calling application should continue to poll using Claim Data until it receives information indicating that the Test Account Credential operation has completed or timed out.

If the OAuth setup process completes successfully, then the Account Credential will have the CREDENTIAL_COMPLETE field set to 1 (true) and the credential can then be used for account discovery and account aggregation operations. If the OAUTH setup process does not complete successfully, then the Account Credential will have the CREDENTIAL_COMPLETE field set to 0 (false) and the OAuth setup process must be started again to obtain the necessary authorization.

An end user may revoke or change an OAuth authorization by logging into the financial institution directly. Again, what the user can do here will vary by financial institution. If the user makes changes to add authorized accounts, these accounts are not automatically discovered by the aggregation system. The caller must complete Account Discovery and Account Add operations to add those accounts to the system. Any accounts whose authorization is revoked at the financial institution will continue to exist in the aggregation system but will no longer aggregate successfully.

A firm must have a Morningstar-level or Firm-level credential configured with the financial institution to use OAuth at that institution. If the firm lacks such a credential and attempts a Test Credential operation at that financial institution, the operation will fail with the error message "Test Account Credential encountered an OAuth configuration error".

Restrictions

This is an asynchronous operation. Please see the [Asynchronous Operations section](#) for general information on handling asynchronous operations.

Behavior

This operation exhibits the following behaviors:

The **<ACCTCREDTESTRS_A>** status of "Operation Timed Out" means that the test took longer than expected and DataConnect is returning the current results. The test runs until it is completed. Therefore, do not issue another **<ACCTCREDTESTRQ_A>** when this error is received. Instead, issue a **<DATAGETRQ_A>** for the Account credentials. In the meantime, look at **LAST_AUTHENTICATION_ATTEMPT** to determine which Account credentials were accessed thus far. If **LAST_AUTHENTICATION_ATTEMPT** is after the time of the initial request, the Account credential was accessed.

Request: <ACCTCREDTESTRQ_A>

Tag	Required	Field	Description
<PROFILE_ID> or <USER_IDENT>	√	USER_ID	User who owns the Account.
<ID>	√	ID	The Credential to be tested (there can be only one).
<ALLOW_USERINPUT_RESPONSE/>			Empty tag. If present, then an ACCTCREDTESTRS_A with a USERINPUT aggregate may be returned, requiring an USERINPUTRESULTRQ to continue the

			operation (eventually a final ACCTCREDTESTRS_A with the final status will be returned).
<START_OAUTH/>			Empty tag. If the Financial Institution has a SUP_OAUTH value of SUPPORTED then this flag must be provided otherwise an error will be returned. If the Financial Institution does not support OAuth then this will be ignored. If present, then an ACCTCREDTESTRS_A response with an OAUTH_MICROSITE_URL and an OAUTH_EXP may be returned, requiring the calling application to launch the OAUTH_MICROSITE_URL so the end user can complete the authorization directly at the Financial Institution.

After the request is made, a series of DATACLAIMRQ responses are made until either a final response or an ACCTCREDTESTRS_A requesting more info is obtained.

Response: <ACCTCREDTESTRS_A>

The <ACCTCREDTESTRS_A> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<RECEIPT>	*		Receipt which may then be used in the Claim Data Request.
<RECEIPT_EXP>	*		Timestamp of when receipt expires.
<CLAIM_WAIT>	*		Number of milliseconds to wait before attempting to retrieve the results of the Test Account Credential operation via the Claim Data request. Present only if <RECEIPT> is present.
<USERINPUT>	*		See USERINPUT Aggregate .
<USERINPUT_EXP>	*		Timestamp of when USERINPUT expires.
<TIMEOUT_INTERVAL>	*		Milliseconds before the first KEEPALIVE should be sent, if full result not sent before then.
<ACCOUNT_CREDENTIAL_STATUS>	*		See <ACCOUNT_CREDENTIAL_STATUS> aggregate.

<OAUTH_MICROSITE_URL>	*		If the Financial Institution uses OAuth for authentication, then you must launch this URL to enable the end user to authorize accounts for the aggregation service at the Financial Institution.
<OAUTH_EXP>	*		Timestamp of when the OAUTH_MICROSITE_URL is no longer valid for completing the authorization. Ten (10) minutes will be allotted for the user to complete the interaction with the OAUTH_MICROSITE_URL to authorize the account access.

*Either (RECEIPT, RECEIPT_EXP, CLAIM_WAIT) or (RECEIPT, RECEIPT_EXP, USERINPUT, USERINPUT_EXP, TIMEOUT_INTERVAL) or (RECEIPT, OAUTH_MICROSITE_URL, OAUTH_EXP) or (ACCOUNT_CREDENTIAL_STATUS) will be included.

If ALLOW_USERINPUT_RESPONSE was used in the initial request, then the response containing USERINPUT and USERINPUT_EXP is possible, but not guaranteed to occur. This response leads to an additional interaction which is described in the [In-Session Activation Codes](#) section.

If START_OAUTH was used in the initial request, then the response containing OAUTH_MICROSITE_URL and OAUTH_EXP will occur if the Financial Institution uses the OAuth method of authentication, otherwise it will not occur. This response leads to an additional out-of-band interaction which is described in the [Purpose](#) section.

Eventually, the Claim Data responses will yield the final result:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS> aggregate .
<USER_IDENT>			USER_IDENT information provided in the request.
<PROFILE_ID>	√		
<ACCOUNT_CREDENTIAL_STATUS>			See <ACCOUNT_CREDENTIAL_STATUS> aggregate .

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
0	Success	Success
0	Success	Operation started
0	Success	Operation in progress
0	Success	Operation paused, awaiting OAuth.
65570	Error	The operation did not complete in its allotted time
66099	Error	The Account Credential could not be found
66107	Error	The requested user was not found
66115	Error	The requested user has been unsubscribed. No further operations can be performed on the user
66778	Warning	The account access operation did not complete in its allotted time. The access attempt is still in progress. You should use DATAGETRQ to retrieve the account status rather than retrying the operation.
66915	Error	Test Account Credentials is not supported at the requested institution.
67315	Error	Test Account Credentials is not allowed for this credential because it is not complete
67395	Error	Test Account Credential requires the START_OAUTH flag for this Account Credential since it is of type OAuth.
67403	Error	Test Account Credential encountered an OAuth configuration error.
67411	Error	Test Account Credential failed because the target Account Credential is configured as OAuth-type but the FI does not support OAuth
67427	Error	Test Account Credential failed due to invalid OAuth firm configuration.
67435	Error	Test Account Credential failed due to invalid OAuth Financial Institution configuration.
67443	Error	Test Account Credential failed because the OAuth Credential was not found.
68099	Error	Credential Test timed out waiting for OAuth response
69923	Error	Test Account Credentials request could not be completed because the account credential was not found

Sample XML

This sample is the simple version of the Test Account Credential operation. The ALLOW_USERINPUT_RESPONSE is not provided in the initial request in this example. Note that even with ALLOW_USERINPUT_RESPONSE included, there may be no actual request for user input, in which case the exchange will still follow the path shown here.

The following is a sample request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>jsmith1</LOGIN_NAME>
    <LOGIN_PW>jsmith1</LOGIN_PW>
  </LOGINRQ>
  <ACCTCREDTESTRQ_A>
    <PROFILE_ID>12</PROFILE_ID>
    <ID>17</ID>
  </ACCTCREDTESTRQ_A>
</DATACONNECTRQ>
```

The following is a sample response, including Receipt:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <ACCTCREDTESTRS_A>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Operation started</ERRMSG>
    </STATUS>
  </ACCTCREDTESTRS_A>
</DATACONNECTRS>
```

```
</STATUS>  
<PROFILE_ID>12</PROFILE_ID>  
<RECEIPT>4456858471129290880</RECEIPT>  
<RECEIPT_EXP>20140512125804 [-5:EDT]</RECEIPT_EXP>  
<CLAIM_WAIT>1000</CLAIM_WAIT>  
</ACCTCREDTESTRS_A>  
</DATACONNECTRS>
```

The following is a sample showing submit the Receipt via Claim Data Request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
<LOGINRQ>  
  <LOGIN_NAME>jsmith1</LOGIN_NAME>  
  <LOGIN_PW>jsmith1</LOGIN_PW>  
</LOGINRQ>  
<DATACLAIMRQ>  
  <RECEIPT>4456858471129290880</RECEIPT>  
</DATACLAIMRQ>  
</DATACONNECTRQ>
```

The following is the final response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <ACCTCREDTESTRS_A>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <FP_ID>17</FP_ID>
    <ACCOUNT_CREDENTIAL_STATUS>
      <ID>12</ID>
      <LAST_AUTHENTICATION_ATTEMPT>20140512105808
      [-5:EDT]</LAST_AUTHENTICATION_ATTEMPT>
      <AUTHENTICATION_STATUS_ERRCODE>1006</AUTHENTICATION_STATUS_
      ERRCODE>
    </ACCOUNT_CREDENTIAL_STATUS>
  </ACCTCREDTESTRS_A>
```

Sample XML with START_OAUTH

This sample is the simple version of the Test Account Credential operation that illustrates the case where the FI uses OAuth for authentication.

The following is a sample request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>jsmith1</LOGIN_NAME>
    <LOGIN_PW>jsmith1</LOGIN_PW>
  </LOGINRQ>
  <ACCTCREDTESTRQ_A>
    <PROFILE_ID>12</PROFILE_ID>
    <ID>17</ID>
    <START_OAUTH>
  </ACCTCREDTESTRQ_A>
</DATACONNECTRQ>
```

The following is a sample response, including Receipt:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <ACCTCREDTESTRS_A>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Operation started</ERRMSG>
    </STATUS>
  </ACCTCREDTESTRS_A>
</DATACONNECTRS>
```

```
</STATUS>  
<PROFILE_ID>12</PROFILE_ID>  
<RECEIPT>4456858471129290880</RECEIPT>  
<RECEIPT_EXP>20140512125804 [-5:EDT]</RECEIPT_EXP>  
<CLAIM_WAIT>1000</CLAIM_WAIT>  
</ACCTCREDTESTRS_A>  
</DATACONNECTRS>
```

The following is a sample showing submit the Receipt via Claim Data Request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
<LOGINRQ>  
  <LOGIN_NAME>jsmith1</LOGIN_NAME>  
  <LOGIN_PW>jsmith1</LOGIN_PW>  
</LOGINRQ>  
<DATACLAIMRQ>  
  <RECEIPT>4456858471129290880</RECEIPT>  
</DATACLAIMRQ>  
</DATACONNECTRQ>
```

The following is the response with the OAUTH information:

<DATACONNECTRS>

<VERSION>VERSION4.0**</VERSION>**

<LOGINRS>

<STATUS>

<ERRCODE>0**</ERRCODE>**

<ERRMSG>Success**</ERRMSG>**

</STATUS>

</LOGINRS>

<ACCTCREDTESTRS_A>

<STATUS>

<ERRCODE>0**</ERRCODE>**

<ERRMSG>Operation paused, awaiting OAuth.**</ERRMSG>**

</STATUS>

<PROFILE_ID>12**</PROFILE_ID>**

<OAUTH_MICROSITE_URL>

http://www.somefi.com/abc/xyz/?response_type=code&redirect_uri=https://www.byallaccounts.net/WebPortfolio/OAUTHCallback&client_id=clientId&state=5c3d3b81-040d-4879-9efe-62543f42ed5b

</OAUTH_MICROSITE_URL>

<OAUTH_EXP>20201026232037 [-5:EDT] **<OAUTH_EXP>**

</ACCOUNT_CREDENTIAL_STATUS>

</ACCTCREDTESTRS_A>

</DATACONNECTRS>

The first time you receive the above information you should launch the user into the OAUTH_MICROSITE_URL and then continue to poll DataConnect using DATACLAIM until the OAUTH operation completes (or times out). A successful final response would look like this:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <ACCTCREDTESTRS_A>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
    <PROFILE_ID>12</PROFILE_ID>
    <ACCOUNT_CREDENTIAL_STATUS>
      <ID>17</ID>
      <LAST_AUTHENTICATION_ATTEMPT>20140512105808
      [-5:EDT]</LAST_AUTHENTICATION_ATTEMPT>
      <AUTHENTICATION_STATUS_ERRCODE>1006</AUTHENTICATION_STATUS_
      ERRCODE>
    </ACCOUNT_CREDENTIAL_STATUS>
  </ACCTCREDTESTRS_A>
</DATACONNECTRS>
```

Test Account

Purpose

Attempts to log in to an Account's Financial Service to verify that the online Access Credentials stored with the Account are correct. The length of this operation depends upon the response time of the Financial Services that ByAllAccounts contacts to obtain account information. Therefore, this operation is implemented as an asynchronous operation.

The Test Account may also include an In-Session-Activation Code (ISAC) interaction, which is described in more detail in [In-Session Activation Codes](#).

Restrictions

This operation has the following restrictions:

4. This is an asynchronous operation. Please see the [Asynchronous Operations section](#) for general information on handling asynchronous operations.
5. There is an additional consideration for Account Test using In-Session Activation Codes. In general, multiple accounts may be submitted in the Test Account operation, and these accounts do not have to have any particular relationship to each other. However, when ALLOW_USERINPUT_RESPONSE is included in the request, then an additional requirement is imposed: all of the Accounts must relate to the same Account Credential. If accounts relating to different account credentials are included in an Account Test with the ALLOW_USERINPUT_RESPONSE tag included, then an error is returned.

Behavior

This operation exhibits the following behaviors:

The **<ACCTTESTRS_A>** status of "Operation Timed Out" means that the test took longer than expected and DataConnect is returning the current results. The test runs until it is completed. Therefore, do not issue another **<ACCTTESTRQ_A>** when this error is received. Instead, issue a **<DATAGETRQ_A>** for the Accounts. In the meantime, look at **LAST_UPDATED** and **LAST_UPDATE_ATTEMPT** to determine which Accounts were accessed thus far. If **LAST_UPDATED** is after the time of the initial request, the update was successful. If **LAST_UPDATE_ATTEMPT** is after the time of the initial request, the Account was accessed.

Request: <ACCTTESTRQ_A>

The <ACCTTESTRQ_A> is the initial request for the asynchronous **Test Account** operation and contains the following:

Tag	Required	Field	Description
<FP_ID> or <USER_IDENT>	√	USER_ID	User who owns the Account.
<ID> or <ALL/>	√	ID	The Account(s) to be tested (can be more than one). If <ALL/> is specified, tests all User Accounts that have sufficient credentials.
<ALLOW_USERINPUT_RESPONSE/>			Empty tag. If present in the initial request, it serves the same purpose as it does in the Test Account Credential operation. An ACCTTESTRS_A with a USERINPUT aggregate may be returned, requiring an USERINPUTRESULTRQ to continue the operation (eventually a final ACCTTESTRS_A with the final status will be returned).

Response: <ACCTTESTRS_A>

The <ACCTTESTRS_A> response contains the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS> aggregate .
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>	√		ID of the Financial Profile for account being tested.
<RECEIPT>			Present only if <STATUS> is successful. Receipt for the client to inquire later about the result of this operation.
<RECEIPT_EXP>			Expiration time of <RECEIPT>
<CLAIM_WAIT>			Number of milliseconds to wait before attempting to retrieve the results of the Test Account operation via the Claim Data request. Present only if <RECEIPT> is present.

This response provides the receipt needed to claim the results of the **Test Account** operation at a later time. A <DATACLAIMRQ> must be subsequently submitted to check on the status of the **Test Account** operation and to retrieve the final results of the operation.

Once the operation is complete, a **<ACCTTESTRS_A>** response containing the following is received:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS> aggregate .
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>	√		ID of the Financial Profile for account being tested.
<ACCOUNT_STATUS>			See <ACCOUNT_STATUS> aggregate . One for each Account tested.

The **Test Account** result may include a USERINPUT but only if **ALLOW_USERINPUT_RESPONSE** was used in the initial request.

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT STATUS> aggregate .
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>	√		ID of the Financial Profile for account being tested.
<RECEIPT>	*		Receipt which may then be used in the Claim Data Request or the Claim File Request
<RECEIPT_EXP>	*		Timestamp of when receipt expires
<USERINPUT>	*		See USERINPUT Aggregate .
<USERINPUT_EXP>	*		Timestamp of when UserInput expires
<TIMEOUT_INTERVAL>	*		Milliseconds before the first KEEPALIVE should be sent, if full result not sent before then.

If a User Input type response is received, then a USERINPUTRESULTRQ must be submitted. This exchange will be the same as for Test Credential. See the section on [In-Session Activation Codes \(ISAC\)](#).

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
0	Success	Success
0	Success	Operation started
0	Success	Operation in progress
65570	Warning	The operation did not complete in its allotted time
66099	Error	The <object name> could not be found
66107	Error	The requested user was not found
66115	Error	The requested user has been unsubscribed. No further operations can be performed on the user
66770	Warning	The account does not contain sufficient credentials to be accessed online
66778	Warning	The account access operation did not complete in its allotted time. The access attempt is still in progress. You should use DATAGETRQ to retrieve the account status rather than retrying the operation.
66787	Error	The user does not have any online accounts at the institution(s)
66795	Error	None of the accounts contain sufficient credentials to be accessed online
66819	Error	Account IDs and ALL cannot be specified in the same request
67571	Error	Account test could not be performed because the ALLOW_USERINPUT_RESPONSE flag was included but accounts relating to multiple credentials were requested

Sample XML

The following is a sample **<ACCTTESTRQ_A>** request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  <ACCTTESTRQ_A>  
    <FP_ID>2300</FP_ID>  
    <ID>1003</ID>  
    <ID>1004</ID>  
  </ACCTTESTRQ_A>  
</DATACONNECTRQ>
```

The following is a corresponding sample **<ACCTTESTRS_A>** status response providing the asynchronous operation receipt to client:

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>...</LOGINRS>  
  <ACCTTESTRS_A>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Operation started</ERRMSG>  
    </STATUS>  
    <RECEIPT>4456858471129290880</RECEIPT>  
    <RECEIPT_EXP>20030621183522 [-5:EST]</RECEIPT_EXP>  
  </ACCTTESTRS_A>  
</DATACONNECTRS>
```

Client later submits the receipt to determine if operation is complete:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>...</LOGINRQ>  
  <DATACLAIMRQ>  
    <RECEIPT>4456858471129290880</RECEIPT>  
  </DATACLAIMRQ>  
</DATACONNECTRQ>
```

The server responds with **Test Account Status** now that operation is complete:

<DATACONNECTRS>

<VERSION>VERSION4.0</VERSION>

<LOGINRS>...</LOGINRS>

<ACCTTESTRS_A>

<STATUS>

<ERRCODE>0</ERRCODE>

<ERRMSG>Success</ERRMSG>

</STATUS>

<FP_ID>2300</FP_ID>

<ACCOUNT_STATUS>

<ID>1003</ID>

<LAST_UPDATE_ATTEMPT>20030723200800</LAST_UPDATE_ATTEMPT>

<LAST_UPDATED>20030723200800</LAST_UPDATED>

<UPDATE_STATUS_ERRCODE>0</UPDATE_STATUS_ERRORCODE>

<UPDATE_STATUS_INFO>Test successful on July 23, 2003

</UPDATE_STATUS_INFO>

</ACCOUNT_STATUS>

<ACCOUNT_STATUS>

<ID>1004</ID>

<LAST_UPDATE_ATTEMPT>20030723200800</LAST_UPDATE_ATTEMPT>

<LAST_UPDATED>20030721150800</LAST_UPDATED>

<UPDATE_STATUS_ERRCODE>1007</UPDATE_STATUS_ERRORCODE>

<UPDATE_STATUS_INFO>WebPortfolio could not log in using the credentials you provided. Error from institution: invalid User name, PIN, or account number entered.

</UPDATE_STATUS_INFO>

</ACCOUNT_STATUS>

</ACCTTESTRS_A>

</DATACONNECTRS>

Update Account from FI

Purpose

Gathers data for all Accounts for the specified Financial Services (for a particular User). The ByAllAccounts service automatically gathers data once per day for all Accounts with sufficient credentials. The primary reason for forcing the ByAllAccounts Service to gather data on-demand is that a new Account may be added and the User would like to view the information for this Account immediately.

The length of this operation is dependent on the response time of the Financial Services that ByAllAccounts contacts to obtain account information. Therefore, this operation is implemented as an asynchronous operation.

Restrictions

1. This is an asynchronous operation. Please see the [Asynchronous Operations section](#) for general information on handling asynchronous operations.
2. DataConnect clients should use this operation only to request immediate updates on behalf of the end user and should not use this operation to implement regular periodic updates of Accounts because this function is already handled by the ByAllAccounts service.
3. Specifying an Account Credential ID is useful when working with an Auto-Managed Account Credential. Such credentials, when initially created, should not have any accounts manually created. Instead, an Account Update for that credential should be submitted, as this will trigger creation of all available Accounts for that Account Credential. An account credential that is not auto-managed may be specified in the request; in this case, all accounts associated with the account credential will be submitted for aggregation.
 - If the Account ID of an individual account associated with an auto-managed credential is submitted for aggregation, the effect will be the same as it would have been if the Account Credential ID were given. Specifically, all accounts related to that account's AC_ID will be updated, and accounts related to that account's AC_ID may be created or marked as closed. This behavior occurs because Auto-Managed accounts can never be aggregated individually; all of the account's related to an Auto-Managed credential must be aggregated collectively.

Behavior

This operation exhibits the following behaviors:

The **<ACCTUPDRS_A>** may have a status of "Operation Timed Out" meaning that the update took longer than expected and DataConnect is returning the current results. The update runs until it is completed. Therefore, do not issue another **<ACCTUPDRQ_A>** when this error is received. Instead, issue a **<DATAGETRQ>** for the Accounts. In the meantime, look at **LAST_UPDATED** and **LAST_UPDATE_ATTEMPT** to determine which Accounts were accessed thus far. If **LAST_UPDATED** is after the time of the initial request, the update was successful. If **LAST_UPDATE_ATTEMPT** is after the time of the initial request, the Account was accessed.

Request: <ACCTUPDRQ_A>

The <ACCTUPDRQ_A> is the initial request for the asynchronous **Update Account from FI** operation and contains the following:

Tag	Required	Field	Description
<FP_ID> or <USER_IDENT>	√	FP_ID	Financial Profile for which the update operation is to be performed.
<FI_ID>, <ACCOUNT_ID>, <AC_ID>, or <ALL/>	√	Account_Credential.FI_ID or ID	Identifies the Financial Services or Accounts to update. Multiple FI_ID , multiple ACCOUNT_ID , multiple AC_ID , or ALL may be specified. If a FI_ID (Financial Service) is specified, data is gathered for all User Accounts for this Financial Service. If multiple AC_ID (Account Credential IDs) are specified, all User Accounts associated with those Account Credentials will be updated. If a specified Account Credential is auto-managed, then account management will also occur as part of this operation. If the specified Account Credential is not auto-managed, then all associated accounts will be updated, but no creation of accounts will occur. If <ALL/> is specified, the Update operation is performed for all Accounts with sufficient credentials. Refer to Restrictions for more details.

Response: <ACCTUPDRS_A>

The <ACCTUPDRS_A> response is either a status response or an operation complete response. The initial response to the <ACCTUPDRQ_A> request contains the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS> aggregate .
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>	√	FP_ID	
<RECEIPT>	√	RECEIPT64	Receipt for the client to inquire at a later time about the result of this operation. Only issued if <STATUS> is successful.
<RECEIPT_EXP>		TIMESTAMP	Expiration time of <RECEIPT>
<CLAIM_WAIT>		NUMBER	Number of milliseconds to wait before attempting to retrieve the results of the Update Account from FI operation via the Claim Data request. Present only if <RECEIPT> is present.

This response provides the receipt needed to claim the results of the **Update Account from FI** operation at a later time. A <DATACLAIMRQ> must be subsequently submitted to check on the status of the **Update Account from FI** operation and retrieve the final results of the operation. Once the operation is completed, an <ACCTUPDRS_A> response containing the following is received:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS> aggregate .
<FP_ID>	√	USER_ID	
<ACCOUNT_STATUS>			See <ACCOUNT_STATUS> aggregate . One for each Account updated.
<ACCOUNT_CREDENTIAL_STATUS>			See <ACCOUNT_CREDENTIAL_STATUS> aggregate .

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
0	Success	Success
0	Success	Operation started
0	Success	Operation in progress
65570	Warning	The operation did not complete in its allotted time
65779	Error	The <field name> is outside the valid range valid range of 0 to 9999999999999999
66099	Error	The <object name> could not be found
66107	Error	The requested user was not found
66115	Error	The requested user has been unsubscribed. No further operations can be performed on the user
66770	Warning	The account does not contain sufficient credentials to be accessed online
66778	Warning	The account access operation did not complete in its allotted time. The access attempt is still in progress. You should use DATAGETRQ to retrieve the account status rather than retrying the operation.
66787	Error	The user does not have any online accounts at the institution(s)
66795	Error	None of the accounts contain sufficient credentials to be accessed online
66811	Error	Financial service IDs and ALL cannot be specified in the same request

Sample XML

The **<ACCTUPDRQ_A>/< ACCTUPDRS_A>** message set is very similar to the **<ACCTTESTRQ_A>/<ACCTTESTRS_A>**. Please refer to that message set for sample XML.

Update Account Tax Lots

Purpose

This operation gathers data for all accounts for the specified Financial Services for a particular User, including tax lot data. The ByAllAccounts service automatically gathers data once per day for all accounts with sufficient credentials. However, the ByAllAccounts service will never automatically gather tax lot data as part of its routine process. This operation may be used to trigger gathering of tax Lot data for accounts at Financial Services where tax lot data is available.

This operation is not an asynchronous operation, and the results cannot be claimed by polling with a **Claim Data** operation. An initial response indicating the request was received will be returned, and then at some point later the data will eventually update, at which time **Data Get** requests for Tax Lot data will reflect the new tax lot data. The operation may take much longer to complete than a typical Update Account operation.

Restrictions

1. This is not an asynchronous operation like Update Account operation. The Account Update with Tax Lots operation will return a response indicating whether the request was successfully initiated or not. If successfully initiated, it will eventually complete and update the tax lot data for the specified accounts.
2. DataConnect clients should use this operation only to request fresh tax lot data when needed. The operation should not be run unless tax lots are required. If fresh tax lot data is not required, then a regular Update Account operation can be used.
3. Only accounts with their gather_lots flag set to TRUE will have tax lot data available, and the gather_lots flag can only be set to TRUE for an account held at a Financial Institution which supports tax lots (see SUP_TAX_LOT). Thus, regardless of the method used to specify the set of accounts for update, only accounts with the gather_lots flags set to TRUE will be updated, and other accounts will be unaffected. If no accounts with set to TRUE are specified, then an error will be reported.
4. The operation will always return an error for firms that do not have tax lot gathering enabled, as no accounts can have tax lot enabled in that case.

Behavior

This operation exhibits the following behaviors:

This operation only provides a response indicating that the operation has successfully been initiated. In order to see the status of the tax lot data, issue a **<DATAGETRQ>** for the Accounts and look at **LAST_UPDATED** and **LAST_UPDATE_ATTEMPT** to determine which Accounts were accessed thus far. If **LAST_UPDATED** is after the time of the initial request, the update was successful. If **LAST_UPDATE_ATTEMPT** is after the time of the initial request, the Account was accessed.

Request: <ACCTUPDTAXLOTRQ>

The <ACCTUPDTAXLOTRQ> contains the following:

Tag	Required	Field	Description
<ADVISOR_USER_IDENT>	*	USER_IDENT	User Ident of an Advisor. If provided, then all accounts held within all financial profiles accessible to this Advisor will be used. If this option is used, then the two items below should NOT be included.
<FP_ID> or <USER_IDENT>	**	FP_ID	Financial Profile for which the update operation is to be performed.
<FI_ID>, <ACCOUNT_ID>, <AC_ID>, or <ALL/>	**	Account_Credential.FI_ID or ID	Identifies the Financial Services or accounts to update. Multiple FI_ID , multiple ACCOUNT_ID , or ALL may be specified. If FI_ID (Financial Service) is specified, data is gathered for all user accounts for this Financial Service. If <ALL/> is specified, the Update operation is performed for all Accounts with sufficient credentials. If AC_ID is specified, all accounts related to the given credential are updated. If the account Credential is auto-managed, this will also trigger automatic creation/deletion of accounts.

Either just * must be included, or any number of repetitions of the ** elements.

Response: <ACCTUPDTAXLOTRS>

The <ACCTUPDTAXLOTRS> response is either a status response or an operation complete response. The initial response to the <ACCTUPDTAXLOTRQ> request contains the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS> aggregate .
<USER_IDENT>	*		USER_IDENT information provided in the request.
<FP_ID>	*	FP_ID	FP_ID provided in the request. There can be any number of FP_IDs followed by IDs.
<ID>			ID of the account. May be multiple account IDs.
<ADVISOR_USER_IDENT>	*		ADVISOR_USER_IDENT information provided in the request.

* When a successful status is returned, at least one of ADVISOR_USER_IDENT, USER_IDENT, or FP_ID will be echoed.

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
0	Success	Success
65779	Error	The <field name> is outside the valid range valid range of 0 to 9999999999999999
66107	Error	The requested user was not found
66115	Error	The requested user has been unsubscribed. No further operations can be performed on the user
66770	Warning	The account does not contain sufficient credentials to be accessed online
66787	Error	The user does not have any online accounts at the institution(s)
66795	Error	None of the accounts contain sufficient credentials to be accessed online
66811	Error	Financial service IDs and ALL cannot be specified in the same request
66931	Error	None of the accounts have tax lot gathering enabled
66971	Error	No financial profiles found for the requested Advisor
67355	Error	Tax lot gathering is not enabled for this firm

Sample XML

Sample ACCTUPDTAXLOTRQ using ADVISOR_USER_IDENT:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>xxxx</LOGIN_NAME>
    <LOGIN_PW>yyyy</LOGIN_PW>
  </LOGINRQ>
<ACCTUPDTAXLOTRQ>
  <ADVISOR_USER_IDENT>
    <USER_IDENT>
      <PERSON_ID>1</PERSON_ID>
    </USER_IDENT>
  </ADVISOR_USER_IDENT>
</ACCTUPDTAXLOTRQ>
</DATACONNECTRQ>
```

Sample response:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
<LOGINRS>
  <STATUS>
    <ERRCODE></ERRCODE>
  </STATUS>
</LOGINRS>
<ACCTUPDTAXLOTRS>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Success</ERRMSG>
  </STATUS>
  <ADVISOR_USER_IDENT>
    <USER_IDENT>
      <PERSON_ID>1</PERSON_ID>
    </USER_IDENT>
  </ADVISOR_USER_IDENT>
</ACCTUPDTAXLOTRS>
</DATACONNECTRS>
```

Discover Accounts

Purpose

This operation allows the user to retrieve a list of accounts that are available to a particular login at a financial institution. These accounts are represented as **DiscoveredAccount** objects and the information provided there can subsequently be used to establish an account for aggregation in the ByAllAccounts service through the separate **Add Account** operation. An account credential may be added prior to executing this operation using **Add Account Credential**.

This operation returns information from a Financial Institution that describes a financial account available under the specified account credential. This set of information is referred to as a **DiscoveredAccount** object; this object is not persistent within the ByAllAccounts service but is used to structure information returned as a result of a discover accounts operation.

DiscoveredAccount Object

Tag	Required	Data Type	Description
<ID>	√	UID	A unique numeric identifier for the discovered account.
<ACCOUNT_NUMBER>	√	CHAR128	The ACCOUNT_NUMBER with which to build a new credential for this discovered account. Exactly one required.
<ACCOUNT_NUMBER_2>		CHAR128	The ACCOUNT_NUMBER_2 with which to build a new credential, up to one allowed; required only if required by the FI.
<ACCOUNT_NAME>		CHAR64	The name of the discovered account, if available.
<TAX_ID>		CHAR9	The tax ID associated with the discovered account, if available. Digits only, no special characters (i.e. dashes).
<FI_SUPPLIED_ACCOUNT_TYPE>		CHAR64	Custodian reported account type for this account.

<ACCOUNT_TYPE>		CHAR32	Account type as determined by the system from information available from the custodian. Possible types are listed in Notes on Account , page 41.
<DISCOVEREDACCOUNTINFO>		See <DISCOVEREDACCOUNTINFO> aggregate as described in Aggregates section that start on page 69	DiscoveredAccount may have zero or more of these subordinate objects. Provides contextual information about the discovered account for use in a user application for reference purposes. This information may not be required to identify the account for aggregation purposes.

Restrictions

1. This is an asynchronous operation. Please see the [Asynchronous Operations section](#) for general information on handling asynchronous operations.
2. Data returned by this operation is not persistent. If account discovery information is desired for the same account credential a second time, a second request must be issued and the server-side operation to gather the discovered account list is fully repeated.
3. This operation may only be performed on a single account credential at a time.
4. This operation will return an error if the account credential is associated with an FI that does not support account discovery. This operation The **Get Financial Service** request can be used to check if an FI supports account discovery.

Behavior

1. When performing an Account Discovery request, **ACCTDISCRQ_A**, if no accounts are found the final response will contain a 1004 errcode along with the empty **DISCOVEREDACCOUNTLIST**.
2. When performing an Incremental Account Discovery request, if some accounts already exist in the system and no new accounts are found the response will contain a 1005 errcode (success) along with the empty **DISCOVEREDACCOUNTLIST**. If no accounts already exist, the 1004 is returned.
3. For credentials of type OAUTH, the Account Discovery operation will cause any existing accounts on the target credential that do not contain an ACCOUNT_NUMBER_2 to be matched to available accounts from the institution and to have their ACCOUNT_NUMBER_2 field updated for any such matches. This update is performed as part of the Account Discovery operation and cannot be prevented. ACCOUNT_NUMBER_2 is used to store the institution's unique ID for this account as provided by the institution's OAUTH API.
4. For credentials of type OAUTH, when subsequently adding accounts from the results of the Account Discovery operation, you must provide the ACCOUNT_NUMBER_2 value from the discovered account to properly configure the account.

Request: <ACCTDISCRQ_A>

The <ACCTDISCRQ_A> is the initial request for the asynchronous **Discover Accounts** operation and contains the following:

Tag	Required	Field	Description
<FP_ID> or <USER_IDENT>	√	FP_ID	Financial Profile for which the Discover Accounts operation is to be performed.
<ID>	√	Account_Credential.ID	Identifies the Account Credential to use for the Discover Accounts operation.
<INCREMENTAL>			If included, indicates that the discover accounts operation should only return accounts that appear to have not already been created on the specified Account Credential. The determination of whether a discovered account is 'new' or not may vary by institution. The default behavior is to case-insensitively compare the concatenated account number and account number 2 fields of the discovered account to any existing accounts, with spaces and dashes ignored. Institution-specific logic may be implemented including inclusion of account number 3.

Response: <ACCTDISCRS_A>

The <ACCTDISCRS_A> response is either a status response or an operation complete response. The initial response to the <ACCTDISCRQ_A> request contains the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>	√	FP_ID	FP_ID information provided in the request.
<RECEIPT>	√	RECEIPT64	Receipt for the client to inquire at a later time about the result of this operation. Only issued if <STATUS> is successful.
<RECEIPT_EXP>		TIMESTAMP	Expiration time of <RECEIPT>
<CLAIM_WAIT>		NUMBER	Number of milliseconds to wait before attempting to retrieve the results of the Discover Accounts operation via the Claim Data request. Present only if <RECEIPT> is present.

This response provides the receipt needed to claim the results of the **Discover Account** operation at a later time. A **<DATACLAIMRQ>** must be subsequently submitted to check on the status of the **Discover Account** operation and retrieve the final results of the operation. Once the operation is completed, an **<ACCTDISCRS_A>** response containing the following is received:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS> aggregate .
<USER_IDENT>			USER_IDENT information provided in the request.
<FP_ID>	√	USER_ID	
<DISCOVEREDACCOUNT_STATUS>	√		See <DISCOVEREDACCOUNT_STATUS> aggregate
<DISCOVEREDACCOUNTLIST>	√		One per response. See <DISCOVEREDACCOUNTLIST> aggregate as described in Aggregates , page 69.

Errors

This operation may return the following errors in the **<STATUS>** portion of the response.

Error Code	Severity	Error Message
0	Success	Success
0	Success	Operation started
0	Success	Operation in progress
65570	Warning	The operation did not complete in its allotted time
65779	Error	The <field name> is outside the valid range valid range of 0 to 9999999999999999
66099	Error	The <object name> could not be found
66107	Error	The requested user was not found
66115	Error	The requested user has been unsubscribed. No further operations can be performed on the user
66770	Error	The account credential does not contain sufficient credentials to be accessed online
66899	Error	Profile does not have access to this account credential
66907	Error	Account Discovery is not supported at this FI.
67307	Error	Account discovery not allowed for this credential because it is not complete
67323	Error	Account discovery failed due to an unexpected response from the Aggregation Server

Sample XML

The following is a sample **<ACCTDISCRQ_A>** request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <ACCTDISCRQ_A>
    <FP_ID>2300</FP_ID>
    <ID>1000</ID>
  </ACCTDISCRQ_A>
</DATACONNECTRQ>
```

The following is a corresponding sample **<ACCTDISCRS_A>** status response providing the asynchronous operation receipt to client:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>
  <ACCTDISCRS_A>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Operation started</ERRMSG>
    </STATUS>
    <FP_ID>2300</FP_ID>
    <RECEIPT>4456858471129290880</RECEIPT>
    <RECEIPT_EXP>20030621183522 [-5:EST]</RECEIPT_EXP>
  </ACCDISCTRS_A>
</DATACONNECTRS>
```

Client later submits the receipt to determine if operation is complete:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>...</LOGINRQ>
  <DATACLAIMRQ>
  <RECEIPT>4456858471129290880</RECEIPT>
  </DATACLAIMRQ>
</DATACONNECTRQ>
```

The server responds with a **DISCOVEREDACCOUNT_STATUS** and **DISCOVEREDACCOUNTLIST** if the operation completes successfully. If an error is reported in the **STATUS** then the **DISCOVEREDACCOUNT_STATUS** and **DISCOVEREDACCOUNTLIST** tags are not returned. It is possible for the **DISCOVEREDACCOUNTLIST** to be empty if an error was encountered while searching for discovered accounts. This error will be reported in the **DISCOVEREDACCOUNT_STATUS** segment.

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>
  <ACCTDISCRS_A>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>

    <FP_ID>2300</FP_ID>

    <DISCOVEREDACCOUNT_STATUS>
      <ID>1000</ID>
      <DISCOVEREDACCOUNT_STATUS_TIMESTAMP>20090621113200
      </DISCOVEREDACCOUNT_STATUS_TIMESTAMP>
      <DISCOVEREDACCOUNT_STATUS_INFO>Account discovery operation successful on Jun 21, 2009
      11:31:38
```

```
</DISCOVEREDACCOUNT_STATUS_INFO>
<DISCOVEREDACCOUNT_STATUS_ERRCODE>1005</DISCOVEREDACCOUNT_STATUS_ERRCODE>
</DISCOVEREDACCOUNT_STATUS>

<DISCOVEREDACCOUNTLIST>
  <DISCOVEREDACCOUNT>
    <ACCOUNT_NUMBER>12345</ACCOUNT_NUMBER>
    <ACCOUNT_NUMBER_2>ABCD</ACCOUNT_NUMBER_2>
    <ACCOUNT_NAME>John Doe's IRA</ACCOUNT_NAME>
    <SSN>987654321</SSN>
    <DISCOVEREDACCOUNTINFO>
      <LABEL>CityStateZip</LABEL>
      <VALUE>Woburn, MA, 01801</VALUE>
    </DISCOVEREDACCOUNTINFO>
  </DISCOVEREDACCOUNT>
  <DISCOVEREDACCOUNT>
    <ACCOUNT_NUMBER>98765</ACCOUNT_NUMBER>
    <ACCOUNT_NUMBER_2>WXYZ</ACCOUNT_NUMBER_2>
    <ACCOUNT_NAME>Jane Doe's IRA</ACCOUNT_NAME>
    <SSN>987654322</SSN>
    <DISCOVEREDACCOUNTINFO>
      <LABEL>CityStateZip</LABEL>
      <VALUE>Waltham, MA, 02453</VALUE>
    </DISCOVEREDACCOUNTINFO>
  </DISCOVEREDACCOUNT>
</DISCOVEREDACCOUNTLIST>
</ACCTDISCRS_A>
</DATACONNECTRS>
```

If a failure had occurred during the data gathering process then the operation itself may return Success but an error may be returned within the DISCOVEREDACCOUNT_STATUS aggregate, for example:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>...</LOGINRS>
  <ACCTDISCRS_A>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>

    <FP_ID>2300</FP_ID>

  <ACCTDISCRS_A>

    <DISCOVEREDACCOUNT_STATUS>
      <ID>1000</ID>
      <DISCOVEREDACCOUNT_STATUS_TIMESTAMP>20090621113200
      </DISCOVEREDACCOUNT_STATUS_TIMESTAMP>
      <DISCOVEREDACCOUNT_STATUS_INFO>Account discovery operation failed on Jun 21,
      2009 11:31:38. Login or password is incorrect.
      </DISCOVEREDACCOUNT_STATUS_INFO>
      <DISCOVEREDACCOUNT_STATUS_ERRCODE>1007
      </DISCOVEREDACCOUNT_STATUS_ERRCODE>
    </DISCOVEREDACCOUNT_STATUS>

  <DISCOVEREDACCOUNTLIST>
  </DISCOVEREDACCOUNTLIST>

</ACCTDISCRS_A>
</DATACONNECTRS>
```

In-Session Activation Codes (ISAC)

A financial institution may require the user to complete an In-Session Activation Code interaction to access their accounts. To support this requirement, the Test Account Credential and Test Account operations can include an In-Session Activation Code (ISAC) interaction when the institution requires it and the calling application has declared it is prepared to accept it. Applications who use DataConnect can choose to provide a user interface to complete the ISAC interaction. If the application cannot provide this user interface then it will limit the accounts that can be configured successfully using DataConnect.

The ISAC interaction is made possible when the calling application includes the `ALLOW_USERINPUT_RESPONSE` element in the Test Account Credential or Test Account initial request. If the Test operation triggers an ISAC interaction, then the institution's first request for information from the user will be returned as the result of the Test operation. Because Test operations are asynchronous the initial ISAC protocol message will arrive when the caller performs a Claim Data operation to get the result of the Test. The Claim Data will produce a response message that contains the `USERINPUT` aggregate.

The `USERINPUT` aggregate defines the structure for a two-way form that enables communication with the end user. The `USERINPUT` aggregate contains many sub-elements to define the form and is described in [USERINPUT Aggregate](#).

Each stage of the ISAC protocol consists of the financial institution asking the user to provide some information (`USERINPUT` aggregate) and the user in turn providing that information (`USERINPUTRESULTRQ` operation containing a response `USERINPUT` aggregate). This message pair repeats until the ISAC interaction fails or succeeds. After the ISAC interaction is completed the outer message protocol for the initiating Test operation continues.

In the case of Test Credential, the caller will receive `ACCTCREDTESTRS_A` with a `USERINPUT` element in the response. The caller must submit a `USERINPUTRESULTRQ` request to continue the operation. The `USERINPUTRESULTRQ` echoes the information in the `USERINPUT` block the user has already received with modifications to reflect the selected values.

The `USERINPUTRESULTRQ` message will contain a receipt and either a list of `INPUTs`, or a `USERCOMMAND`. The `USERCOMMAND` may be one of "QUIT" or "KEEPALIVE". If a QUIT command is submitted, the user may still continue to submit `DATACLAIM` operations to attain the final state of their `ACCTCREDTESTRQ`, which will indicate an error status on the Account Credential. The `KEEPALIVE` command simply prevents the operation from timing out, but only up until the `USERINPUT_EXP` time is reached; when this time is reached then the operation will time out and subsequent `USERINPUTRESULTRQ` messages will then receive an error. `DATACLAIMRQ` messages will yield the `ACCTCREDTESTRS_A` with the final result, which will show an error status on the Account Credential. The `TIMEOUT_INTERVAL` indicates when the next `KEEPALIVE` command should be sent in order to prevent the timeout.

Purpose

An ISAC is a one-time, temporary identification code to be requested and entered to establish that a user is authorized to log into an account. The USERINPUTRESULTRQ message is used by the caller to provide the user's response to the financial institution's request for information.

Request: <USERINPUTRESULTRQ>

The USERINPUTRESULTRQ should contain the following:

Tag	Required	Field	Description
<RECEIPT>	√		Receipt which may then be used in the Claim Data Request or the Claim File Request
<USERINPUTRESULT>	√		See USERINPUT Aggregate .

The USERINPUTRESULT aggregate will contain a list of the following:

Tag	Required	Field	Description
<USERINPUT>	*		Multiple allowed. See USERINPUT Aggregate .
<USERCOMMAND>	*		Only one allowed. A command indicating keep alive or quit, see <USERCOMMAND> aggregate

*Either a list of USERINPUTs is present OR a single USERCOMMAND is present, never both

The USERCOMMAND aggregate will contain:

Tag	Required	Field	Description
<COMMAND>	√		One of "QUIT" or "KEEPALIVE"
<COMMENT>			Any string value, useful for logging purposes

Response: <USERINPUTRESULTRS>

The <USERINPUTRESULTRS> response will contain the following:

Tag	Required	Data Type	Description
<STATUS>	√		See <STATUS> aggregate description. For the results of the actual test, see <ACCOUNT_STATUS>.
<RECEIPT>		RECEIPT64	Receipt for the client to inquire about the status of the ongoing operation. Only issued if <STATUS> is successful. This will be the same receipt as the one returned by the original operation.
<CLAIM_WAIT>		NUMBER	Number of milliseconds to wait before attempting to retrieve the results of the ongoing Test Credential or Test Account request via the Claim Data request. Present only if <RECEIPT> is present.

After receiving the USERINPUTRESULTRS, the caller should resume issuing Claim Data operations to determine the status of the operation. Subsequent responses indicating a need for User Input may occur, triggering the sequence again. The ISAC sequence may occur any number of times.

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error	Severity	Message
65699	Error	The request is invalid or formatted incorrectly
65947	Error	Caller is not authorized to make this request
66579	Error	The receipt has expired
67539	Error	Checkbox value must be either true or false
67547	Error	User input result could not be submitted because the operation has timed out
67555	Error	User input result could not be submitted because the user input for this receipt was not found
67579	Error	User input result could not be processed

USERINPUT Aggregate

The USERINPUT aggregate and its sub-elements appear in ACCTCREDTESTRS_A and ACCTTESTRS_A (which come from the server) and also in the USERINPUTRESULTRQ (which is constructed by the user).

The USERINPUT aggregate is described here because the USERINPUT aggregate is used to manage ISAC interactions.

The USERINPUT aggregate is defined using multiple aggregates including INPUT, INPUTCHECKBOX, INPUTTEXT, INPUTRADIOGROUP, and INPUTRADIO.

The USERINPUT aggregate will contain a list of INPUTs:

Tag	Required	Data Type	Description
INPUT	√	Aggregate	Multiple allowed. See INPUT aggregate.

The INPUT aggregate will contain exactly one of INSTRUCTIONS, INPUTRADIOGROUP, INPUTCHECKBOX, or INPUTTEXT:

Tag	Required	Data Type	Description
INSTRUCTIONS	*	CHAR512	A string providing general instructions that apply to any/all INPUT objects which follow.
INPUTTEXT	*	Aggregate	See <INPUTTEXT> aggregate.
INPUTCHECKBOX	*	Aggregate	See <INPUTCHECKBOX> aggregate.
INPUTRADIOGROUP	*	Aggregate	See <RADIOGROUP> aggregate.

* Exactly one of these is required.

The INPUTCHECKBOX aggregate will contain:

Tag	Required	Data Type	Description
INSTRUCTIONS		CHAR512	A string providing instructions that relate to this checkbox-like value prompt.
IDENTIFIER	√	NUMBER	A string identifying this INPUTCHECKBOX.
LABEL	√√	CHAR512	A label for this checkbox.
VALUE	√	CHAR256	A value for this checkbox, which will be the string "true" or the string "false".

√√ indicates the field is only required in ACCTCREDTESTRS_A and ACCTTESTRS_A but not in USERINPUTRESULTRQ.

The INPUTTEXT aggregate will contain:

Tag	Required	Data Type	Description
INSTRUCTIONS		CHAR512	A string providing instructions that relate to this input text-like value prompt.
IDENTIFIER	√	NUMBER	A string identifying this INPUTTEXT object.
LABEL	√√	CHAR512	A label for this text box.
VALUE	√	CHAR1024	A value for this text box. Required, but may be empty string.

√√ indicates the field is required in ACCTCREDTESTRS_A and ACCTTESTRS_A but not in USERINPUTRESULTRQ.

The INPUTRADIOGROUP aggregate will contain:

Tag	Required	Data Type	Description
IDENTIFIER	√	NUMBER	A string identifying this INPUTTEXT object.
VALUE	√	CHAR256	The value should be the IDENTIFIER of the selected INPUTRADIO item, or empty to indicate no selection.
INPUTRADIO	√√	Aggregate	A list of INPUTRADIO items to select from, see INPUTRADIO aggregate.

√√ indicates the field is required in ACCTCREDTESTRS_A and ACCTTESTRS_A but not in USERINPUTRESULTRQ.

The INPUTRADIO aggregate will contain:

Tag	Required	Data Type	Description
INSTRUCTIONS		CHAR512	A string providing instructions that relate to this input radio item.
IDENTIFIER	√	NUMBER	A string identifying this INPUTRADIO object.
LABEL	√√	CHAR512	The label for this input radio label.

Notes:

√√ indicates the field is required in ACCTCREDTESTRS_A and ACCTTESTRS_A but not in USERINPUTRESULTRQ.

Sample XML

This code example shows a Test Account Credential operation with in-session activation codes.

As shown, when formulating the USERINPUTRESULTRQ, the user must respond to each of the INPUT objects provided in the preceding ACCTCREDTESTRQ_A, except for the INSTRUCTIONS type which requires no response.

This is a sample of the exchange.

Step 1: Submit the initial Test Account Credential request

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>jsmith1</LOGIN_NAME>
    <LOGIN_PW>jsmith1</LOGIN_PW>
  </LOGINRQ>
  <ACCTCREDTESTRQ_A>
    <FP_ID>12</FP_ID>
    <ID>17</ID>
    <ALLOW_USERINPUT_RESPONSE/>
  </ACCTCREDTESTRQ_A>
</DATACONNECTRQ>
```

Step 1A: Receive the Test Account Credential Response, including Receipt:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
```

```
<ACCTCREDTESTRS_A>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Operation started</ERRMSG>
  </STATUS>
  <PROFILE_ID>12</PROFILE_ID>
  <RECEIPT> 4456858471129290880</RECEIPT>
  <RECEIPT_EXP>20140512125804 [-5:EDT]</RECEIPT_EXP>
  <CLAIM_WAIT>1000</CLAIM_WAIT>
</ACCTCREDTESTRS_A>
</DATACONNECTRS>
```

Step 2: Submit the Receipt via Claim Data Request:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>jsmith1</LOGIN_NAME>
    <LOGIN_PW>jsmith1</LOGIN_PW>
  </LOGINRQ>
  <DATACLAIMRQ>
    <RECEIPT>4456858471129290880</RECEIPT>
  </DATACLAIMRQ>
</DATACONNECTRQ>
```

Step 2A: Receive a Test Account Credential response requesting User Input:

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <ACCTCREDTESTRS_A>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Operation paused, awaiting user input result.</ERRMSG>
    </STATUS>
    <FP_ID>28671</FP_ID>
    <RECEIPT>4456858471129290880</RECEIPT>
    <USERINPUT>
      <INPUT>
        <INSTRUCTIONS>The institution needs to send you an identification code.
        </INSTRUCTIONS>
      </INPUT>
      <INPUT>
        <INSTRUCTIONS>This is a time-sensitive process that may require more input from you.
        </INSTRUCTIONS>
      </INPUT>
      <INPUT>
        <INSTRUCTIONS>Choose how you would like to receive the code:
        </INSTRUCTIONS>
      </INPUT>
      <INPUT>
```

```
<INPUTTEXT>
  <INSTRUCTIONS>Use the code 'ABC123', otherwise a 1007 error will be returned.
  </INSTRUCTIONS>
  <IDENTIFIER>1</IDENTIFIER>
  <LABEL>Enter Code : </LABEL>
  <VALUE></VALUE>
</INPUTTEXT>
</INPUT>
<INPUT>
  <INPUTCHECKBOX>
    <INSTRUCTIONS>Check the box, otherwise a 1007 error will be returned.
    </INSTRUCTIONS>
    <IDENTIFIER>2</IDENTIFIER>
    <LABEL>2345</LABEL>
    <VALUE>>false</VALUE>
  </INPUTCHECKBOX>
</INPUT>
<INPUT>
  <INPUTRADIOGROUP>
    <IDENTIFIER>3</IDENTIFIER>
    <VALUE></VALUE>
    <INPUTRADIO>
      <INSTRUCTIONS>Select one of the options, otherwise a 1007 will be returned
      </INSTRUCTIONS>
      <IDENTIFIER>4</IDENTIFIER>
      <LABEL>xxbar@xxxxxxfoobar.com</LABEL>
    </INPUTRADIO>
    <INPUTRADIO>
      <IDENTIFIER>5</IDENTIFIER>
      <LABEL>XXX-XXX-7890</LABEL>
    </INPUTRADIO>
  </INPUTRADIOGROUP>
</INPUT>
```

<INPUT>

<INSTRUCTIONS>If none of the above methods work, please contact technical support for assistance.

</INSTRUCTIONS>

</INPUT>

</USERINPUT>

<USERINPUT_EXP>20140516120852 [-5:EDT]</USERINPUT_EXP>

<TIMEOUT_INTERVAL>90000</TIMEOUT_INTERVAL>

</ACCTCREDTESTRS_A>

</DATACONNECTRS>

Step 3: Submit the KEEPALIVE request after the timeout interval has elapsed (if a user input result is ready for submission before the TIMEOUT_INTERVAL elapses, then this step does not occur. If the timeout_interval elapses multiple times, then this step is repeated up until the expiration time. After the expiration_time, no further KEEPALIVE requests are accepted and the final result can be obtained via DATACLAIMRQ).

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>jsmith1</LOGIN_NAME>
    <LOGIN_PW>jsmith1</LOGIN_PW>
  </LOGINRQ>
  <USERINPUTRESULTRQ>
    <RECEIPT>4456858471129290880</RECEIPT>
    <USERINPUTRESULT>
      <USERCOMMAND>
        <COMMAND>KEEPALIVE</COMMAND>
        <COMMENT>periodic keepalive request</COMMENT>
      </USERCOMMAND>
    </USERINPUTRESULT>
  </USERINPUTRESULTRQ>
</DATACONNECTRQ>
```

Step 3A: The response to a KEEPALIVE request.

```
<DATACONNECTRS>
  <VERSION>VERSIONX.X</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
```

```
<ERRMSG>Success</ERRMSG>
</STATUS>
</LOGINRS>
<USERINPUTRESULTRS>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Success (user input result processed)</ERRMSG>
  </STATUS>
  <RECEIPT>4456858471129290880</RECEIPT>
</USERINPUTRESULTRS>
</DATACONNECTRS>
```

Step 4: Submit the User Input Response. In this version of the User Input Response, all tags from the original user input request are included and VALUE tags are added to signal the responses to each item.

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>jsmith1</LOGIN_NAME>
    <LOGIN_PW>jsmith1</LOGIN_PW>
  </LOGINRQ>
  <USERINPUTRESULTRQ>
    <RECEIPT>4456858471129290880</RECEIPT>
    <USERINPUTRESULT>
      <USERINPUT>
        <INPUT>
          <INSTRUCTIONS>The institution needs to send you an identification code.
        </INSTRUCTIONS>
        </INPUT>
        <INPUT>
          <INSTRUCTIONS>This is a time-sensitive process that may require more input from you.
        </INSTRUCTIONS>
```

</INPUT>

<INPUT>

<INSTRUCTIONS>Choose how you would like to receive the code:

</INSTRUCTIONS>

</INPUT>

<INPUT>

<INPUTTEXT>

<INSTRUCTIONS>Use the code 'ABC123', otherwise a 1007 error will be returned.

</INSTRUCTIONS>

<IDENTIFIER>1</IDENTIFIER>

<LABEL>Enter Code : </LABEL>

<VALUE>ABC123</VALUE>

</INPUTTEXT>

</INPUT>

<INPUT>

<INPUTCHECKBOX>

<INSTRUCTIONS>Check the box, otherwise a 1007 error will be returned.

</INSTRUCTIONS>

<IDENTIFIER>2</IDENTIFIER>

<LABEL>2345</LABEL>

<VALUE>>true</VALUE>

</INPUTCHECKBOX>

</INPUT>

<INPUT>

<INPUTRADIOGROUP>

<IDENTIFIER>3</IDENTIFIER>

<VALUE>4</VALUE>

<INPUTRADIO>

<INSTRUCTIONS>Select one of the options, otherwise a 1007 will be returned

</INSTRUCTIONS>

<IDENTIFIER>4</IDENTIFIER>

<LABEL>xxbar@xxxxxxfoobar.com</LABEL>

</INPUTRADIO>

```
<INPUTRADIO>
  <IDENTIFIER>5</IDENTIFIER>
  <LABEL>XXX-XXX-7890</LABEL>
</INPUTRADIO>
</INPUTRADIOGROUP>
</INPUT>
<INPUT>
  <INSTRUCTIONS>If none of the above methods work, please contact technical
  support for assistance.
</INSTRUCTIONS>
</INPUT>
</USERINPUT>
</USERINPUTRESULT>
</USERINPUTRESULTRQ>
</DATACONNECTRQ>
```

Step 4A: This is an alternative to the previous request which will have the same impact. This response provides the minimum necessary information to convey the responses to each input item.

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>jsmith1</LOGIN_NAME>
    <LOGIN_PW>jsmith1</LOGIN_PW>
  </LOGINRQ>
  <USERINPUTRESULTRQ>
    <RECEIPT>4456858471129290880</RECEIPT>
  <USERINPUTRESULT>
    <USERINPUT>
      <INPUT>
        <INPUTTEXT>
```

```
<IDENTIFIER>1</IDENTIFIER>
  <VALUE>ABC123</VALUE>
</INPUTTEXT>
</INPUT>
<INPUT>
  <INPUTCHECKBOX>
    <IDENTIFIER>2</IDENTIFIER>
    <VALUE>>true</VALUE>
  </INPUTCHECKBOX>
</INPUT>
<INPUT>
  <INPUTRADIOGROUP>
    <IDENTIFIER>3</IDENTIFIER>
    <VALUE>4</VALUE>
  </INPUTRADIOGROUP>
</INPUT>
</USERINPUT>
</USERINPUTRESULT>
</USERINPUTRESULTRQ>
</DATACONNECTRQ>
```

Step 4B: User Input Result Response (successful)

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <USERINPUTRESULTRS>
```

```
<STATUS>
<ERRCODE>0</ERRCODE>
<ERRMSG>Success (user input result processed)</ERRMSG>
</STATUS>
<RECEIPT>4456858471129290880</RECEIPT>
<CLAIM_WAIT>1000</CLAIM_WAIT>
</USERINPUTRESULTRS>
</DATACONNECTRS>
```

Step 5: Submit the receipt again until final response (or another request for user input) is received:

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>jsmith1</LOGIN_NAME>
    <LOGIN_PW>jsmith1</LOGIN_PW>
  </LOGINRQ>
  <DATACLAIMRQ>
    <RECEIPT>4456858471129290880</RECEIPT>
  </DATACLAIMRQ>
</DATACONNECTRQ>
```

Step 5A: Final Response

```
<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
```

```
<ACCTCREDTESTRS_A>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Success</ERRMSG>
  </STATUS>
  <FP_ID>17</FP_ID>
  <ACCOUNT_CREDENTIAL_STATUS>
    <ID>12</ID>
    <LAST_AUTHENTICATION_ATTEMPT>20140512105808 [-5:EDT]
    </LAST_AUTHENTICATION_ATTEMPT>
    <AUTHENTICATION_STATUS_ERRCODE>1006</AUTHENTICATION_STATUS_ERRCODE>
  </ACCOUNT_CREDENTIAL_STATUS>
</ACCTCREDTESTRS_A>
</DATACONNECTRS>
```

Note: Instead of a USERINPUTRESULT in step 4, or the KEEPALIVE in step 5, the user may optionally submit a QUIT command instead. After the QUIT command is received and relayed to the aggregation server, a DATACLAIMRQ may still be submitted to obtain the final result from the Aggregation Server (the Aggregation server will report an error due to not receiving the user input). Note that a QUIT command is only meaningful when the Aggregation Server is awaiting user input; the USERINPUTRESULTRQ operation with a QUIT command cannot be used to cancel aggregation operations that are not in a waiting-for-user-input state.

```
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>jsmith1</LOGIN_NAME>
    <LOGIN_PW>jsmith1</LOGIN_PW>
  </LOGINRQ>
  <USERINPUTRESULTRQ>
    <RECEIPT>4456858471129290880</RECEIPT>
    <USERINPUTRESULT>
      <USERCOMMAND>
        <COMMAND>QUIT</COMMAND>
        <COMMENT>user wants to cancel operation</COMMENT>
```

```
</USERCOMMAND>  
</USERINPUTRESULT>  
</USERINPUTRESULTRQ>  
</DATACONNECTRQ>
```

The response to a QUIT request:

```
<DATACONNECTRS>  
  <VERSION>VERSIONX.X</VERSION>  
  <LOGINRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success</ERRMSG>  
    </STATUS>  
  </LOGINRS>  
  <USERINPUTRESULTRS>  
    <STATUS>  
      <ERRCODE>0</ERRCODE>  
      <ERRMSG>Success (user input result processed)</ERRMSG>  
    </STATUS>  
    <RECEIPT>4456858471129290880</RECEIPT>  
  </USERINPUTRESULTRS>  
</DATACONNECTRS>
```

Single Sign-on Operations

The DataConnect API contains these operations that support Single Sign-on (SSO):

- Session Authenticate Request
- Session Expire
- SAML Session Authenticate Request

These operations are described in this section. They can be used to implement SSO with ByAllAccounts products including AccountView, the Morningstar Connect components, and the Consumer User Interface (CUI2) application.

For details about implementing SSO for AccountView, refer to http://www.byallaccounts.net/Manuals/Accountview/AccountView_SingleSignOn.pdf.

SSO - Session Authenticate Request

Purpose

Establishes an application session for a specific user.

Restrictions

This operation has the following restrictions:

- The operation is used as part of the single sign-on mechanism only.
- The operation is only valid for users belonging to a firm for which the product has been licensed.
- A configuration option is available at the Firm level to set whether the user whose credentials are provided in LOGINRQ must be an Administrator. When the option is set to YES, the LOGINRQ must be made by an administrator. When it is set to NO, the LOGINRQ must be made by an investor, advisor, or assistant on their own behalf.
- SESSIONAUTHRQ identifies the user for whom the ByAllAccounts product session will be established. The target user can be identified using one of the following options:
 - USER_IDENT aggregate, which enables identification of a user by PERSON_ID, PERSON_FIRM_TAG1, or PERSON_LOGIN_NAME.or
 - LOGIN_NAME and LOGIN_PASSWORD for the target user.

Request: <SESSIONAUTHRQ>

The <SESSIONAUTHRQ> can contain the following:

Tag	Required	Field	Description
<LOGIN_NAME>	*	LOGIN_NAME	Login name for the user who is going to access the ByAllAccounts product.
<LOGIN_PW>	*	LOGIN_PW	Password that goes with LOGIN_NAME .
<USER_IDENT>	*	USER_IDENT	Used to provide identifying information for the user when establishing a session. See <USER_IDENT> aggregate description .
<READ_ONLY>		READ_ONLY	Establishes an investor session as read only. Can only be used when the investor's firm is licensed for REST API. Either present or not. The <READ_ONLY> tag has no content. Only an administrator authenticating an investor or an investor self-authenticating can submit the <READ_ONLY> tag. When present, the generated session is read-only and data cannot be modified even if the target investor has write permissions.

* Request must either include both <LOGIN_NAME> and <LOGIN_PW> together or just <USER_IDENT> alone.

Response: <SESSIONAUTHRS>

The <SESSIONAUTHRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<APPLICATION_URL>		CHAR1000	The value returned is based on the user type and license status of the firm. For Morningstar Connect, ignore the URL
<SESSION_ID>		CHAR64	The SESSION ID to use with the ByAllAccounts product.
<CSRF_TOKEN>		CHAR64	The CSRF Token to use with the ByAllAccounts product.

Errors

This operation may return the following errors in the **<STATUS>** portion of the response:

Error Code	Severity	Error Message
56003	Error	The login name or password is incorrect
56027	Error	The password has expired
66987	Error	Single sign on not allowed by this firm for this user type
67003	Error	The user specified in the login request is not authorized to authenticate the target user
67019	Error	User's login is disabled
67043	Error	User not configured for this login mode (user must login through SAML single sign-on)
67051	Error	Read only session may only be requested when investor's firm is licensed for REST API
67059	Error	Read only session may only be requested by administrators or investors
67067	Error	Read only session may only be requested for investors
67091	Error	AccountView product is not licensed for this user

Sample XML

Using USER_IDENT

The following is a sample **<SESSIONAUTHRQ>** request that establishes a session for the target user using USER_IDENT.

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>AdminUser</LOGIN_NAME>  
    <LOGIN_PW>adminpassword123</LOGIN_PW>  
  </LOGINRQ>  
  <SESSIONAUTHRQ>  
    <USER_IDENT>  
      <PERSON_ID>62469</PERSON_ID>  
    </USER_IDENT>  
  </SESSIONAUTHRQ>  
</DATACONNECTRQ>
```

Using LOGIN_NAME and LOGIN_PW

The following is a sample <SESSIONAUTHRQ> request that establishes a session for the target user using the login name and password.

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>jsmith</LOGIN_NAME>  
    <LOGIN_PW>mypassword123</LOGIN_PW>  
  </LOGINRQ>  
  <SESSIONAUTHRQ>  
    <LOGIN_NAME>jsmith</LOGIN_NAME>  
    <LOGIN_PW>mypassword123</LOGIN_PW>  
  </SESSIONAUTHRQ>  
</DATACONNECTRQ>
```

Sample Response

The following is a sample response.

```
<DATACONNECTRS>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRS>  
    <STATUS>  
  <ERRCODE>0</ERRCODE>  
  <ERRMSG>Success</ERRMSG>  
  </STATUS>  
  </LOGINRS>  
  <SESSIONAUTHRS>  
    <STATUS>  
  <ERRCODE>0</ERRCODE>  
  <ERRMSG>Success</ERRMSG>  
  </STATUS>  
  <APPLICATION_URL>https://www.byallaccounts.net...</APPLICATION_URL>  
  <SESSION_ID>B6B0948EF74DA6238B32E77669F6C9ED.s1a</SESSION_ID>  
<CSRF_TOKEN>BC53B58BC4C0E9FA6E2799D13183A68421EAC68E0503E41</CSRF_TOKEN>  
  </SESSIONAUTHRS>  
</DATACONNECTRS>
```

SSO - Session Expire Request

Purpose

Terminates the session.

Restrictions

This operation has the following restrictions:

- The operation is only valid for sessions established using SESSIONAUTHRQ.

Request: <SESSIONEXPIRERQ>

The <SESSIONEXPIRERQ> can contain the following:

Tag	Required	Field	Description
<LOGIN_NAME>		LOGIN_NAME	Optional. If provided, it is checked against the name of the target user if one was provided when the session was established. Note: This is not the same as the LOGIN_NAME used in LOGINRQ.
<SESSION_ID>	√	CHAR64	The SESSION ID returned by the original SESSIONAUTHRQ
<CSRF_TOKEN>	√	CHAR64	The CSRF Token returned by the original SESSIONAUTHRQ

Response: <SESSIONEXPIRERS>

The <SESSIONEXPIRERS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
66995	Error	Session could not be found

Sample XML

Session expire when USER_IDENT is used

The following is a sample **<SESSIONEXPIRERQ>** request. Note that the LOGIN_NAME and LOGIN_PW are for the user named in the LOGINRQ in the SESSIONAUTHRQ.

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>AdminUser</LOGIN_NAME>  
    <LOGIN_PW>adminpassword123</LOGIN_PW>  
  </LOGINRQ>  
  <SESSIONEXPIRERQ>  
    <SESSION_ID>B6B0948EF74DA6238B32E77669F6C9ED.s1a</SESSION_ID>  
<CSRF_TOKEN>BC53B58BC4C0E9FA6E2799D13183A68421EAC68E0503E41  
</CSRF_TOKEN>  
  </SESSIONEXPIRERQ>  
</DATACONNECTRQ>
```

Session expire when LOGIN_NAME and LOGIN_PW are used

The following is a sample **<SESSIONEXPIRERQ>** request:

```
<DATACONNECTRQ>  
  <VERSION>VERSION4.0</VERSION>  
  <LOGINRQ>  
    <LOGIN_NAME>jsmith</LOGIN_NAME>  
    <LOGIN_PW>mypassword123</LOGIN_PW>  
  </LOGINRQ>  
  <SESSIONEXPIRERQ>  
<LOGIN_NAME>jsmith</LOGIN_NAME>  
<SESSION_ID>B6B0948EF74DA6238B32E77669F6C9ED.s1a</SESSION_ID>  
<CSRF_TOKEN>BC53B58BC4C0E9FA6E2799D13183A68421EAC68E0503E41  
</CSRF_TOKEN>  
  </SESSIONEXPIRERQ>  
</DATACONNECTRQ>
```

Session expire response

The following is a sample **<SESSIONEXPIRERS>** response for the above requests:

```
<DATACONNECTRS>
```

```
<VERSION>VERSION4.0</VERSION>
<LOGINRS>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Success</ERRMSG>
  </STATUS>
</LOGINRS>
<SESSIONEXPIRERS>
  <STATUS>
    <ERRCODE>0</ERRCODE>
    <ERRMSG>Success</ERRMSG>
  </STATUS>
</SESSIONEXPIRERS>
</DATACONNECTRS>
```

SAML SSO – SAML Session Authenticate Request

Purpose

Establishes a ByAllAccounts session from a SAML assertion for an Investor user. The established session can be used with the BAA REST API or BAA UI Custom Elements by providing the returned session tokens to those products.

Restrictions

This operation has the following restrictions:

- LOGINRQ for this request can only specify the login of a DataConnect Ultra Administrator user. Typically, this DataConnect Ultra Administrator user will be associated to the investor's firm, however, the operation also supports use of a DataConnect Ultra Administrator that is associated with a Firm that is an ancestor of the investor's firm. If the Administrator is in an ancestor firm then the SAML IdP configured for the Administrator's firm must match that of the Investor's firm, otherwise the error "Incorrect SAML identity provider" (67099) will occur.
- The operation is only valid for establishing a session for users of type Investor who are SSO users.
- The Investor that is the target of this operation must belong to a Firm with the following configuration:
 - SAML support is enabled with a UIM SP identifier specified.
 - Licensed for at least one of the following: REST API and one of the following: Account Linking Custom Element, Account Summary Custom Element, or NetWorth Custom Elements.
- Each SAML assertion can only be used once to establish a session. If the same assertion is repeated, UIM returns an error resulting in the 67115 "An unexpected error returned from SAML provider" error.
- SESSIONEXPIRERQ cannot be used to terminate the session established by SAMLAUTHRQ.

Behavior

This operation is one step in a SAML authentication process involving your firm's SAML implementation, Morningstar UIM SAML, and ByAllAccounts DataConnect API. Use of this operation requires the following:

- The Investor user who is the target of this request must have been previously created in the ByAllAccounts system. The caller must have retained the BAA Person Id for that investor for subsequent use on each SAMLAUTHRQ call to establish a session for that investor.
- Your firm has been configured in Morningstar UIM for SAML. This configuration includes exchange of SAML metadata and certificates, creation of IDP and SP entities, and configuration of UIM to obtain the BAA Person Id attribute from the SAML payload, typically using the attribute label BAAPersonId.

Once these one-time onboarding activities have been completed, SAML authentication is established using the following steps:

1. Produce a signed SAML assertion for the Morningstar ByAllAccounts SAML service provider (SP) that identifies the target Investor via the BAAPersonId attribute.
2. Call DataConnect API SAMLAUTHRQ to validate the signed SAML assertion and obtain session tokens.
3. Provide the session tokens when launching the ByAllAccounts UI custom elements or REST API.
4. Handle session token expiration by repeating the authentication process.

The custom elements of the ByAllAccounts Connect Suite are described in [ByAllAccounts Connect Suite - Account summary with Account Linking](#).

Note that the established ByAllAccounts session will time out after 30 minutes of inactivity and has a maximum duration of 1 hour.

Request: <SAMLAUTHRQ>

The <SAMLAUTHRQ> request contains the following:

Tag	Required	Field	Description
<SAML_ASSERTION>	√	SAML_ASSERTION	Contains a base64-encoded SAML assertion from the identity provider (IP). The SAML assertion body must contain the ByAllAccounts Investor Id with the appropriate attribute label as established during the SAML onboarding process with UIM. Typically, this attribute label will be BAAPersonId (case-sensitive).

Response: <SAMLAUTHRS>

The <SAMLAUTHRS> response can contain the following:

Tag	Required	Field	Description
<STATUS>	√		See <STATUS> aggregate description.
<SESSION_ID>		CHAR64	The SESSION ID for the established session
<CSRF_TOKEN>		CHAR64	The CSRF Token for the established session

Errors

This operation may return the following errors in the <STATUS> portion of the response:

Error Code	Severity	Error Message
65555	Error	SAML assertion is required
65947	Error	Caller is not authorized to make this request
66107	Error	The requested user was not found
67075	Error	SAML assertion must be a valid Base64 encoded string
67083	Error	The SAML target user must be an investor
67091	Error	SAML is not allowed for the target user's vendor
67099	Error	Incorrect SAML identity provider
67107	Error	Target user not licensed for REST API access
67115	Error	An unexpected error returned from SAML provider
67123	Error	Target user not specified

Sample XML

The following is a sample **<SAMLAUTHRQ>** request.

<DATACONNECTRQ>

<VERSION>VERSION4.0</VERSION>

<LOGINRQ>

<LOGIN_NAME>AdminUser</LOGIN_NAME>

<LOGIN_PW>adminpassword123</LOGIN_PW>

</LOGINRQ>

<SAMLAUTHRQ>

<SAML_ASSERTION>

PHNhbWxwOjJlc3BvbnNIIHhtbG5zOnNhbWxwPSJ1cm46b2FzaXM6bmFtZXM6dGM6U0FNTDoyLjA6cHJvdG9jb2wilEIE
PSJfzjVmYzU5YWl4YzlkNjExMzI4OTAiIjCBWZlXJzaW9uPSlyLjAilElzc3VISW5zdGFudD0iMjAyNC0wOC0xNVQxOToxNjow
My44MTFaliAgRGVzdGluYXRpb249Imh0dHBzOi8vbG9naW4tc3RnLm1vcmludG9zaW50aDAuY29tPC9zYW
jaz9jb25uZWNoaW9uPVNBTUwtU1AtQkFBLUdFTkVSSUMtU1RHlj48c2FtbDpJc3N1ZXIgeG1sbnM6c2FtbD0idXJuOm9h
c2lzOm5hbWVzOnRjOINBTUw6Mi4wOmFzc2VydGlvbil+dXJuOnNhbWwtaWRwLXRlc3QudXMtYXV0aDAuY29tPC9zYW
1sOkkzc3Vlcj48c2FtbHA6U3RhdHVzPjxzYW1scDpTdGF0dXN...cudzMub3JnLzlwMDEvWE1MU2NoZW1hLWluc3RhbmN
lIj48c2FtbDpBdHRyaWJ1dGUgUmFtZT0iVXNlcklkIj48c2FtbDpBdHRyaWJ1dGVWYXx1ZSB4c2k6dHlwZT0ieHM6c3Rya
W5nlj42NzQzNzwwc2FtbDpBdHRyaWJ1dGVWYXx1ZT48L3NhbWw6QXR0cmliidXRIPjxzYW1sOkF0dHJpYnV0ZSBOYW1I
PSJLZWVwQWxpdmVUkwiPjxzYW1sOkF0dHJpYnV0ZVZhbHVlIHhzaTp0eXBIPSJ4czpzdHJpbmciPmh0dHBzOi8vc3RnY
XAubWFpbnFjY291bnQuY29tL1diYkFwcC9zdG10L0FXUktlZXBBbGI2ZTwwc2FtbDpBdHRyaWJ1dGVWYXx1ZT48L3Nhb
Ww6QXR0cmliidXRIPjwvc2FtbDpBdHRyaWJ1dGVtZGF0ZW1lbnQ+PC9zYW1sOkFzc2VydGlvbj48L3NhbWxwOjJlc3Bvbn
NIPg==

</SAML_ASSERTION>

</SAMLAUTHRQ>

</DATACONNECTRQ>

Sample Response

The following is a sample response. The SAMLAUTHRS will contain a new SESSION_ID and CSRF_TOKEN, similar to the SESSIONAUTHRS, however there will be no APPLICATION_URL in the response as this is not required for invocation of the custom element

<DATACONNECTRS>

<VERSION>VERSION4.0</VERSION>

<LOGINRS>

...

</LOGINRS>

<SAMLAUTHRS>

<STATUS>

<ERRCODE>0</ERRCODE>

<ERRMSG>Success</ERRMSG>

```
</STATUS>  
<SESSION_ID>FA3694D89908124811FA8435CFA76BE2.s2a</SESSION_ID>  
<CSRF_TOKEN>B325758B91ADCBC64FAF63FBFDB6C2BC8DE466C599BDA697</CSRF_TOKEN>  
</SAMLAUTHRS>  
</DATACONNECTRS>
```

DATACONNECT USAGE CONSIDERATIONS

DataConnect Access

Users must be authorized to use DataConnect. When ByAllAccounts deploys the product, the following items are addressed:

- **Access Credentials:** Select Users are given an Administrative User ID and Password that allows them to retrieve data for their users using DataConnect. It is the User's responsibility to manage and secure these credentials, since they provide access to a limited set of the Investors' personal and financial information.
- **Access Frequency:** Users are given a choice whether to use the bulk or on-demand styles. If they subsequently need to change their usage style, they must contact their ByAllAccounts relationship manager.
- **Access Time:** If Users choose bulk access, they must agree upon an expected access time and frequency with ByAllAccounts. If they need to make a change to this agreement, they must contact their ByAllAccounts relationship manager.

Compression

Response documents and error documents returned by DataConnect are always compressed in the ZIP compression format. Users must ensure that their program performs decompression of the returned data.

Data Availability

DataConnect can be invoked at any time. However, new data may not always be available for a User or Account due to the following limitations:

- The service may download data from Financial Services only during certain hours of the day (e.g., when the service website is available for access).
- Account data may only be as current as that available from the Financial Service. Many Financial Services update their online data within specific time intervals.

ByAllAccounts cannot guarantee that all accounts are updated with the previous day's information or that a particular account is updated by a particular time of day. ByAllAccounts updates account data once a day (during the early morning hours) at the time known to be best for each particular Financial Service. Therefore, there is a suggested "best time" to call the API in order to obtain fresh data. This does not imply that the API cannot be used at other times, only that the new data returned might not be different than the data returned in the previous call. The suggested "best time" depends on the Financial Services being used by the firm.

For bulk usage, it is strongly recommended to call the API after the suggested "best time" to reduce unnecessary bulk data downloading. The timestamp indicating freshness of the object data is provided on each Account (**Account.LAST_UPDATED**) field, Transaction (**Transaction.CREATION_DATE**) field, and Holding (**Holding.LAST_UPDATED**) field.

APPENDIX A: DATACONNECT ACCESS AND DOCUMENT TYPE DEFINITIONS

This section provides a quick reference to the DataConnect access point and DataConnect definitions.

The main DataConnect URL (case-sensitive):

<https://www.byallaccounts.net/dataconnect/WPServlet?RequestType=DataConnectV4>

Response documents and error documents returned by DataConnect are always compressed in the ZIP compression format.

To return the response documents and error documents in non-compressed format, specify [compressResponse=false](#) in the URL (as shown below),

<https://www.byallaccounts.net/dataconnect/WPServlet?RequestType=DataConnectV4&compressResponse=false>

Note that [compressResponse=false](#) can be used for all DataConnect operations except FILECLAIMRQ. The response for FILECLAIMRQ it will always have ZIP compression.

The DataConnect Document Type (DTD) definitions are described in two separate documents.

This document describes the inbound requests:

<http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/DataConnectRQ.dtd>

This document describes the responses that come back:

<http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/DataConnectRS.dtd>

APPENDIX B: DATACONNECT ERROR CODES

When errors occur in DataConnect, an error code and accompanying message are generated. These error codes are of the following severity: Success, Informational, Warning, and Error.

Error Code	Severity	Error Message
0	Success	Success
0	Success	Operation started
0	Success	Operation in progress
0	Success	Operation paused, awaiting OAuth.
56003	Error	The login name or password is incorrect
56011	Error	Caller's account is disabled. Contact technical support for assistance
56019	Error	Caller has been unsubscribed and can no longer log in
56027	Error	The password has expired
56803	Error	The new password does not meet minimum length requirements
56811	Error	The new password cannot be a single repeated character
56819	Error	The new password cannot be all letters or all numbers
56827	Error	The new password must be different than the old password
56835	Error	The password has too many sequential or repeating characters (e.g. AAAA or 1234)
56843	Error	The new password cannot be the same as the login
56851	Error	The new password must have at least one letter
65547	Error	An internal error occurred
65555	Error	One or more required fields are missing
65563	Error	The <field name> cannot exceed <max> characters
65570	Warning	The operation did not complete in its allotted time
65579	Error	The receipt has expired
65699	Error	The request is invalid or formatted incorrectly
65707	Error	The date is invalid or formatted incorrectly
65715	Error	The transaction ID range is invalid
65723	Error	The transaction date range is invalid
65731	Error	The transaction start date is invalid
65739	Error	The transaction end date is invalid

Appendix B: DataConnect Error Codes (continued):

Error Code	Severity	Error Message
65747	Error	The DataConnect version is unsupported or invalid
65755	Error	The user type must be one of INVESTOR, ADVISOR, or CONSULTANT.
65763	Error	Email addresses must be in the form x@x.x and cannot contain spaces
65771	Error	The phone number is not valid
65779	Error	The <field name> is outside the valid range valid range of 0 to 9999999999999999
65787	Error	Boolean values must be 0 or 1
65795	Error	The request referred to unsupported or invalid DTD
65803	Error	The DOCTYPE is missing or incomplete
65811	Error	The user type must be one of INVESTOR, ADVISOR, or CONSULTANT
65819	Error	The request was empty
65843	Error	Only dates prior to today are allowed
65875	Error	The FILETYPE is invalid
65939	Error	The requested user was not found
65947	Error	Caller is not authorized to make this request
65955	Error	DataConnect product is not licensed
65963	Error	User type not supported for this operation.
66099	Error	The requested data could not be found
66107	Error	The requested user was not found
66115	Error	The requested user has been unsubscribed. No further operations can be performed on the user
66123	Error	The <object name> has been archived and cannot be modified
66129	Informational	Some requested users were not found
66139	Error	The profile was not found
66259	Error	The user already has a portfolio with this name
66267	Error	The user already has an account with this name
66275	Error	A user with the login name already exists
66418	Warning	The user was successfully created, but an error occurred while sending email to the user
66426	Warning	The user was successfully modified, but an error occurred while sending email to the user

Appendix B: DataConnect Error Codes (continued):

Error Code	Severity	Error Message
66434	Warning	The user was successfully unsubscribed, but an error occurred while sending email to the user
66442	Warning	The user was successfully deleted, but an error occurred while sending email to the user
66450	Warning	The user was successfully created, but an error occurred while sending email to the advisor
66579	Error	The receipt has expired
66586	Warning	You are not set up to use dual logins. The secondary login information was not added to the user.
66603	Error	The current password must be supplied in order to modify the password
66611	Error	The current password must be supplied in order to modify the password hint
66619	Error	The current password must be supplied in order to modify the email address
66635	Error	The LOGIN_NAME cannot be modified
66643	Error	The ROLE cannot be modified
66651	Error	Invalid FP Access value. Valid values are: NONE, READ, READLIMITEDWRITE, and READWRITE
66658	Warning	CONSULTANT_TO value is ignored, user does not have CONSULTANT role
66666	Warning	ADVISOR_USER_IDENT value is ignored, an advisor cannot be assigned to this user
66675	Error	CONSULTANT_TO user could not be found
66683	Error	ADVISOR_USER_IDENT user could not be found
66691	Error	CONSULTANT_TO user has no financial profile
66699	Error	The OWN_FP_ACCESS value was ignored; it can only be specified if the user has role INVESTOR and the firm's usage model is Advisor-managed
66707	Error	The new user was not notified because no LOGIN was defined
66715	Error	Advisors may not be created in a firm that uses the Investor-managed model
66722	Warning	User was added as a client of the calling Advisor. ADVISOR_USER_IDENT value was ignored
66731	Error	User doesn't have a Login, requested operation cannot be performed
66739	Error	The user does not have any portfolios
66747	Error	Only one of the financial institution ID or its name can be specified
66763	Error	The financial institution URL cannot be specified since its ID has been specified. Financial institution names and URLs are used for institution support requests only.

Appendix B: DataConnect Error Codes (continued):

Error Code	Severity	Error Message
66770	Warning	The account does not contain sufficient credentials to be accessed online
66778	Warning	The account access operation did not complete in its allotted time. The access attempt is still in progress. You should use DATAGETRQ to retrieve the account status rather than retrying the operation.
66787	Error	The user does not have any online accounts at the institution(s)
66795	Error	None of the accounts contain sufficient credentials to be accessed online
66802	Warning	The financial institution does not use the specified account credential field
66811	Error	Financial service IDs and ALL cannot be specified in the same request
66819	Error	Account IDs and ALL cannot be specified in the same request
66834	Warning	The account was created but there was an error when requesting the Update from FI .
66843	Error	The account credential could not be deleted because there are accounts that reference it.
66851	Error	Either the financial institution ID or its name must be specified
66856	Success	Account DO_UPDATE request was submitted successfully
66865	Information	Account DO_UPDATE request not submitted because account does not have all required credentials
66899	Error	Profile does not have access to this account credential
66907	Error	Account Discovery is not supported at this FI.
66915	Error	Test Account Credentials is not supported at the requested institution.
66931	Error	None of the accounts have tax lot gathering enabled
66939	Error	The account could not be added because the FI does not support trade-based data
66947	Error	The account could not be added because the FI does not support settlement-based data
66963	Error	Error setting data basis
66971	Error	No financial profiles found for the requested Advisor
66987	Error	Single sign on not allowed by this firm for this user type
66995	Error	Session could not be found
67003	Error	The user specified in the login request is not authorized to authenticate the target user
67019	Error	User's login is disabled
67043	Error	User not configured for this login mode (user must login through SAML single sign-on)

Appendix B: DataConnect Error Codes (continued):

Error Code	Severity	Error Message
67051	Error	Read only session may only be requested when investor's firm is licensed for REST API
67059	Error	Read only session may only be requested by administrators or investors
67067	Error	Read only session may only be requested for investors
67075	Error	SAML assertion must be a valid Base64 encoded string
67083	Error	The SAML target user must be an investor
67091	Error	AccountView product is not licensed for this user or SAML is not allowed for the target user's vendor
67099	Error	Incorrect SAML identity provider
67107	Error	Target user not licensed for REST API access
67115	Error	An unexpected error returned from SAML provider
67123	Error	Target user not specified
67139	Error	Invalid Client FP Access value. This field is only valid for Assistants. Valid values are: READ and READWRITE
67147	Error	The CLIENT_FP_ACCESS value was ignored; it can only be specified if the user has role ASSISTANT and the firm's usage model is Advisor-managed
67155	Error	Assistants may not be created in a firm that uses the Investor-managed model
67163	Error	Not licensed to perform operation at restricted financial institution
67299	Error	The Account Credential could not be deleted because an Account associated with the Account Credential could not be deleted
67305	Error	Only administrators may explicitly set IS_SSO value.
67307	Error	Account discovery not allowed for this credential because it is not complete
67315	Error	Test Account Credential is not allowed for this credential because it is not complete.
67323	Error	Account discovery failed due to an unexpected response from the Aggregation Server
67331	Error	Automatic account management is not enabled for this firm
67339	Error	Automatic account management is not supported at this FI
67347	Error	Choosing data basis is not enabled for this firm
67355	Error	Tax lot gathering is not enabled for this firm

Appendix B: DataConnect Error Codes (continued):

Error Code	Severity	Error Message
67363	Error	Tax lot gathering is not supported at this FI
67371	Error	Setting DATA_BASIS is not allowed for an account credential without automatic account management
67379	Error	Setting GATHER_LOTS is not allowed for an account credential without automatic account management
67395	Error	Test Account Credential requires the START_OAUTH flag for this Account Credential since it is of type OAuth.
67403	Error	Test Account Credential encountered an OAuth configuration error.
67411	Error	Test Account Credential failed because the target Account Credential is configured as OAuth-type but the FI does not support OAuth
67427	Error	Test Account Credential failed due to invalid OAuth firm configuration.
67435	Error	Test Account Credential failed due to invalid OAuth Financial Institution configuration.
67443	Error	Test Account Credential failed because the OAuth Credential was not found.
67466	Warning	Account file not completed
67474	Warning	Position file not completed
67482	Warning	Transaction file not completed
67499	Error	Bulk Export files not generated
67507	Error	At least one type of file must be included
67514	Warning	Investment Option file not completed
67522	Warning	Client file not completed
67539	Error	Checkbox value must be either true or false
67547	Error	User input result could not be submitted because the operation has timed out
67555	Error	User input result could not be submitted because the user input for this receipt was not found
67571	Error	Account test could not be performed because the ALLOW_USERINPUT_RESPONSE flag was included but accounts relating to multiple credentials were requested
67579	Error	User input result could not be processed
67635	Error	At least one of first name, middle name, or last name must be provided.
67643	Error	The user type must be one of INVESTOR, ADVISOR, ASSISTANT, or CONSULTANT
67722	Warning	Multiple files not completed

Appendix B: DataConnect Error Codes (continued):

Error Code	Severity	Error Message
67739	Error	Access was not performed from an allowed IP address.
67836	Warning	Automatic account management no longer available for this account credential, therefore credential-level settings for DATA_BASIS and GATHER_LOTS will no longer apply
67859	Error	Export file could not be found
67891	Error	Discovered account could not be found
67899	Error	Account to be added does not match Discovered Account for financial profile
67907	Error	Account to be added does not match Discovered Account for credential
67915	Error	Account to be added does not match Discovered Account for account number
67923	Error	Account to be added does not match Discovered Account for account number 2
67930	Warning	The new user was not notified because the user has single sign on access.
67938	Warning	Response contains partial data set
67995	Error	GENERATE_LOGINS cannot be used; usage conflicts with firm configuration that creates new Investors as SSO by default
68027	Error	Invalid value for EXTERNAL_SERVICE_LEVEL: Must be one of Positional, Transactional
68067	Error	The feed onboarding request has been submitted and cannot be changed.
68075	Error	Feed onboarding requests are not supported at this FI.
68099	Error	Credential Test timed out waiting for OAuth response
69923	Error	Test Account Credentials request could not be completed because the account credential was not found

APPENDIX C: ACCOUNT UPDATE STATUS ERROR CODES

The **Account.UPDATE_STATUS_ERRCODE** field is used to store the status from the most recent **Test Account** or **Update Account from FI** operation. The following table lists the error codes and messages that may be returned in the **UPDATE_STATUS_ERRCODE** field of the [Account object](#). Note that the actual error message may be different, depending on what is returned from the site. An error category is included to indicate the likely source of each error. A table listing these error categories follows the table below.

Error Code	Problem/Error Category	Possible Error Message
1005	Success (No Problem)	Account updated successfully.
400	Internal Product	Internal Error: Incorrect download command.
401	Financial Service	Internal Error: Incorrect download command authorization.
403	Financial Service	Internal Error: Incorrect download command access.
404	Financial Service	Internal Error: Download target not found.
500	Financial Service	Internal Error: Download command caused server error.
503	Financial Service	The institution's server is not available. Try again later.
1001	Internal Product	Unable to parse downloaded data.
1002	Internal Product	Internal Error: WebPortfolio server incorrectly configured.
1003	Financial Service or Credential	Unable to navigate remote web site. If this account has worked previously the financial institution's online services may be temporarily unavailable -- if the condition persists beyond a few hours, contact Technical Support.
1004	Financial Service or Credential	The data downloaded contained no holding balance or status information. This may be a temporary problem with the institution's server or may indicate a problem in WebPortfolio's support for the web site. If the condition persists, please notify Technical Support.
1006	Successful Test	Account test succeeded.
1007	Credential	WebPortfolio is unable to log in to this institution using the credentials you provided. Until the credentials are corrected, WebPortfolio will not log in to your account automatically (to avoid account lockout). Verify the information you entered, including the Financial Institution you selected, in WebPortfolio and try manually updating the account. If the condition persists, please notify Technical Support.
1008	Financial Service or Credential	WebPortfolio is unable to process your account information and has determined that the financial institution's website has changed.

Appendix C: Account Update Status Error Codes (continued):

Error Code	Error Category	Error Message
1009	Credential	Unable to process additional accounts due to an account limitation being exceeded.
1010	Account	Account is no longer available in the financial institution's account list.
2000	Financial Service	The institution's server is not available. Try again later.
2001	Credential	Invalid account number. Check that the number is correct. If the number includes punctuation marks (such as dashes) or blanks, try removing them. See the institution's instructions for more details.
2002	Credential	Unable to access account. If the number includes punctuation marks (such as dashes) or blanks, try removing them. See the institution's instructions for more details.
2003	Credential	The account number is not available to this online login.
2004	Credential	Information not available. This account has been closed.
2005	Credential	This account does not allow data to be downloaded. Contact your financial institution.
2018	Internal Product	Internal error: The specified server ID does not exist.
2019	Internal Product	Internal error: Duplicate <TRNUID> .
2020	Internal Product	Internal error: Unparseable date time.
2021	Internal Product	Internal error: Message set version not supported.
2023	Internal Product	Internal error: The specified FITID/BILLID does not exist.
2025	Internal Product	Internal error: <BRANCHID> required for this country system.
12250	Internal Product	The institution does not support investment transaction download.
12251	Internal Product	The institution does not support investment position download.
12252	Internal Product	The institution does not support downloading investment positions for the specified date.
12253	Internal Product	The institution does not support open order download.
12254	Internal Product	The institution does not support investment balances download.
15000	Credential	The next account access must specify a new online password. See the institution's instructions for more details.
15500	Credential	Unable to log in. Either the online login or password is incorrect. See the institution's instructions for more details.
15501	Credential	Unable to log in. Your online login was being accessed from elsewhere. The institution does not support concurrent access.
15502	Credential	Unable to log in. Your online access is locked out by the institution. Contact your financial institution.
15505	Internal Product	Internal error: Country system not supported
15506	Internal Product	Internal error: No information requested.
15507	Credential	Unable to log in. You must supply a new password. See the institution's instructions for more details.

Appendix C: Account Update Status Error Codes (continued):

Category codes are used to classify the error codes returned in the **UPDATE_STATUS_ERRCODE** field of the [Account object](#). They indicate the likely source of the error.

Problem/Error Category	Description
Credential	<p>Error codes in this category indicate a problem with the online Access Credentials for the Account, including:</p> <ul style="list-style-type: none">• ACCOUNT_LOGIN, ACCOUNT_PIN, ACCOUNT_NUMBER (or other credential information) incorrect• Insufficient information to log in (e.g., new password must be provided on first log in)• Account not properly enabled for online access at the financial service <p>Often requires action by the end user to correct.</p>
Financial Service	The Financial Service is unavailable or responding in an abnormal way. Try the operation again later. If the error persists, contact Technical Support.
Internal Product	Infrequently occurring, these errors indicate an internal product error and should be reported to Technical Support.
Success	Successful result

APPENDIX D: SAMPLE CALL TO THE API

The following is a program fragment written in the Java programming language. It uses the capabilities available in the Java 2 programming environment (Java 1.3 Runtime). This fragment demonstrates the invocation of the DataConnect URL, posting of a request document to that URL, receipt of a response stream, and the writing of that compressed data (ZIP) to a file.

The compressed file produced by the fragment:

- Can be read using a tool capable of decompressing data in ZIP compression format (such as WinZip® or GNU gzip)
- Should be given a file extension of .zip
- Will contain a single XML file with the response data or an error response document
- Will be named **WpGetData mm dd hh mi .xml** where **mm dd hh mi** is a date and time (**mm** =month, **dd** =day, **hh** =hour, **mi** =minute)

...

```
import java.io.ByteArrayOutputStream;
```

```
import java.io.File;
```

```
import java.io.FileInputStream;
```

```
import java.io.FileOutputStream;
```

```
import java.io.FileReader;
```

```
import java.io.InputStream;
```

```
import java.net.HttpURLConnection;
```

```
import java.net.URL;
```

...

```
try {
```

```
    // Configure for use of https
```

```
        System.setProperty("java.protocol.handler.pkgs","com.sun.net.ssl.internal.www.protocol");
```

```
        java.security.Security.addProvider( new com.sun.net.ssl.internal.ssl.Provider());
```

```
// open the URL to DataConnect
//
String contactUrl =
"https://www.byallaccounts.net/dataconnect/WPServlet?RequestType=DataConnectV4";

String inputFileName = "myRequest.xml"; // my request document

URL url = new URL(contactUrl);
URLConnection urlc = (URLConnection)url.openConnection();

urlc.setDoOutput (true);
urlc.setUseCaches (false);
urlc.setRequestProperty ("Content-Type", "text/xml; charset=UTF-8");
urlc.setRequestProperty ("Accept-Encoding", "zip");
urlc.setRequestMethod ("POST");
ByteArrayOutputStream byteStream =
    (ByteArrayOutputStream)urlc.getOutputStream();
```

Appendix D: Sample Call to the API (continued):

// write the request document (POST it) to the DataConnect URL

```
FileReader fileReader = new FileReader(new File(inputFileName));  
int next=fileReader.read();  
while ( next != (-1))  
{  
    byteStream.write(next );  
    next=fileReader.read();  
}  
byteStream.flush();
```

// create the file to store the response - response data will be compressed in ZIP format

```
String outputFileName = "responseFile";
```

```
FileOutputStream reply = new FileOutputStream(File.createTempFile(outputFileName, ".zip"));  
InputStream input = urlc.getInputStream();
```

// read compressed bytes from the response write them to our file

```
int received = input.read();  
while (received != (-1))  
{  
    reply.write(received);  
    received=input.read();  
}  
input.close();  
reply.close();
```

```
} catch (Exception e) {
```

```
    // report exception
```

```
}
```

APPENDIX E: SAMPLE ERROR RESPONSE DOCUMENTS

This section presents three common error Response Documents and includes suggested actions for correcting them.

See [Appendix A: DataConnect Access and Document Type Definitions](#) for the Request and Response DTD files.

Example 1:

Error Message: The DOCTYPE is missing or incomplete.

Possible Error: No DTD definition provided for DATACONNECTRQ in Request Document

Error Received:

You receive the following error in response to a Request Document:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DATACONNECTRS PUBLIC '-//DataConnect DTD//DataConnect//EN'
'http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/DataConnectRS.dtd'>

<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <GENERALRS>
    <STATUS>
      <ERRCODE>65803</ERRCODE>
      <ERRMSG>The DOCTYPE is missing or incomplete</ERRMSG>
    </STATUS>
  </GENERALRS>
</DATACONNECTRS>
```

Appendix E: Sample Error Response Documents (continued):

Example 1 (continued)

Your Input:

You provided a Request Document to DataConnect similar to the following:

```
<?xml version="1.0" encoding="UTF-8"?>

<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>mylogin</LOGIN_NAME>
    <LOGIN_PW>mypassword1234</LOGIN_PW>
  </LOGINRQ>
  <DATAGETRQ>
    <GET_DATA_QUERY>
      <USER_QUERY>
        <USER_IDENT>
          <PERSON_FIRM_TAG1>000111222</PERSON_FIRM_TAG1>
        </USER_IDENT>
        <USER_IDENT>
          <PERSON_FIRM_TAG1>435242</PERSON_FIRM_TAG1>
        </USER_IDENT>
      </USER_QUERY>
    </GET_DATA_QUERY>
    <INCFI/>
    <INCPORFOLIO/>
    <INCAccount/>
  </DATAGETRQ>
</DATACONNECTRQ>
```

Appendix E: Sample Error Response Documents (continued):

Example 1 (continued)

Suggested Action:

The error indicates that in attempting to parse the Request Document, the parser had no DTD definition for the first element, **DATACONNECTRQ**. The reason for the error is that the Request did not provide a **DOCTYPE** reference to identify the DTD to use to validate the document. Correct the Request Document to include the **DOCTYPE**.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DATACONNECTRQ PUBLIC "-//DataConnect DTD//DataConnect//EN"
'http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/DataConnectRQ.dtd'>

<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>mylogin</LOGIN_NAME>
    <LOGIN_PW>mypassword1234</LOGIN_PW>
  </LOGINRQ>
  <DATAGETRQ>
<GET_DATA_QUERY>
  <USER_QUERY>
  <USER_IDENT>
    <PERSON_FIRM_TAG1>000111222</PERSON_FIRM_TAG1>
  </USER_IDENT>
  <USER_IDENT>
    <PERSON_FIRM_TAG1>435242</PERSON_FIRM_TAG1>
  </USER_IDENT>
  </USER_QUERY>
</GET_DATA_QUERY>
<INCFI/>
<INCPORFOLIO/>
<INCAccount/>
  </DATAGETRQ>
</DATACONNECTRQ>
```

Appendix E: Sample Error Response Documents (continued):

Example 2:

Error Message: Element "DATAGETRQ" does not allow "get_data_query" here.

Possible Error: Lower case lettering used in element name

Error Received:

You receive the following error in response to a Request Document:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DATACONNECTRS PUBLIC "-//DataConnect DTD//DataConnect//EN"
'http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/DataConnectRS.dtd'>

<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <GENERALRS>
    <STATUS>
      <ERRCODE>65699</ERRCODE>
      <ERRMSG>Element "DATAGETRQ" does not allow "get_data_query" here.</ERRMSG>
    </STATUS>
  </GENERALRS>
</DATACONNECTRS>
```

Appendix E: Sample Error Response Documents (continued):

Example 2 (continued)

Your Input:

You provided a Request Document to DataConnect similar to the following:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DATACONNECTRQ PUBLIC "-//DataConnect DTD//DataConnect//EN"
'http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/DataConnectRQ.dtd'>
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>mylogin</LOGIN_NAME>
    <LOGIN_PW>mypassword1234</LOGIN_PW>
  </LOGINRQ>
  <DATAGETRQ>
    <get_data_query>
      <USER_IDENT>
        <PERSON_LOGIN_NAME>marysmith</PERSON_LOGIN_NAME>
      </USER_IDENT>
    </get_data_query >
    <INCPORFOLIO/>
  </DATAGETRQ>
</DATACONNECTRQ>
```

Appendix E: Sample Error Response Documents (continued):

Suggested Action:

The error indicates that the element **get_data_query** is not allowed within **DATAGETRQ**. The reason for the error is that element names are case-sensitive. Always use upper case element names for DataConnect. Correct the error in the Request Document by using **GET_DATA_QUERY** instead of **get_data_query**.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DATACONNECTRQ PUBLIC "-//DataConnect DTD//DataConnect//EN"
'http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/DataConnectRQ.dtd'>
<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>mylogin</LOGIN_NAME>
    <LOGIN_PW>mypassword1234</LOGIN_PW>
  </LOGINRQ>
  <DATAGETRQ>
    <GET_DATA_QUERY>
      <USER_IDENT>
        <PERSON_LOGIN_NAME>marysmith</PERSON_LOGIN_NAME>
      </USER_IDENT>
    </GET_DATA_QUERY>
    <INCPORFOLIO/>
  </DATAGETRQ>
</DATACONNECTRQ>
```

Appendix E: Sample Error Response Documents (continued):

Example 3:

Error Message: The requested user was not found

Possible Error: Error in spelling the user's name

Error Received:

You receive the following error in response to a Request Document:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DATACONNECTRS PUBLIC '-//DataConnect DTD//DataConnect//EN'
'http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/DataConnectRS.dtd'>

<DATACONNECTRS>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRS>
    <STATUS>
      <ERRCODE>0</ERRCODE>
      <ERRMSG>Success</ERRMSG>
    </STATUS>
  </LOGINRS>
  <DATAGETRS>
    <STATUS>
      <ERRCODE>66107</ERRCODE>
      <ERRMSG>The requested user was not found</ERRMSG>
    </STATUS>
  </DATAGETRS>
</DATACONNECTRS>
```

Appendix E: Sample Error Response Documents (continued):

Example 3 (continued)

Your Input:

You provided a Request Document to DataConnect to retrieve the user MARYSMITH, but misspelled the user's Login name:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DATACONNECTRQ PUBLIC "-//DataConnect DTD//DataConnect//EN"
'http://www.byallaccounts.net/WebPortfolio/com/baa/dtd/v4/DataConnectRQ.dtd'>

<DATACONNECTRQ>
  <VERSION>VERSION4.0</VERSION>
  <LOGINRQ>
    <LOGIN_NAME>mylogin</LOGIN_NAME>
    <LOGIN_PW>mypassword1234</LOGIN_PW>
  </LOGINRQ>
  <DATAGETRQ>
    <GET_DATA_QUERY>
      <USER_IDENT>
        <PERSON_LOGIN_NAME>marysmit</PERSON_LOGIN_NAME>
      </USER_IDENT>
    </GET_DATA_QUERY>
    <INCPORFOLIO/>
  </DATAGETRQ>
</DATACONNECTRQ>
```

Suggested Action:

Check and correct the spelling of the user's Login name in the Request Document (i.e., marysmith in this example).

APPENDIX F: TAX LOT FILES

This appendix describes the possible Tax Lot files.

Tax Lot File Names

The Tax Lot files are named with the following format:

- BAAOPENLOTS_<FI_ID>_<date>_<DeliveryStyle>.csv
- BAACLOSEDLOTS_<FI_ID>_<date>_< DeliveryStyle >.csv

Part	Description
prefix	"BAAOPENLOTS" or "BAACLOSEDLOTS", identifying the type of lots.
FI_ID	The BAA financial institution id.
date	Dates are in the format YYYYMMDD where YYYY is a 4-digit year (e.g., 2003), MM is a 2-digit month code from 01 (January) through 12 (December), and DD is a 2-digit day code from 01 to 31.
DeliveryStyle	<p>The delivery style of the file. Possible values are as follows:</p> <ul style="list-style-type: none"> ▪ Open Lot Files <ul style="list-style-type: none"> ▪ FULL - Contains all open lots. ▪ Closed Lot Files: <ul style="list-style-type: none"> ▪ COMPLETE - The file is the complete universe of closed lots from the first date seen to the last date seen in the file. ▪ NEW_ONLY - Only contains new records. ▪ KEYED_DELTA - Only records that have been updated or are new. They are keyed based on broker lot ID.

Open Lot Tax File

This table describes the BAAOPENLOTS_<FI_ID>_<date>_<DeliveryStyle>.csv file.

If the custodian data does not fit within the defined field, it will be truncated.

Column Header	Required?	Data Type	Description
ACCOUNT_NUMBER	√	CHAR128	The number of the account that contains this lot.
SECURITY_TYPE		CHAR20	This is the security type value provided by the custodian, if available.
FI_SUPPLIED_CUSIP		CHAR20	The CUSIP for the security corresponding to this lot as provided by the FI.

Column Header	Required?	Data Type	Description
FI_SUPPLIED_TICKER		CHAR32	The ticker symbol for the security corresponding to this lot as provided by the FI.
FI_SUPPLIED_SEDOL		CHAR7	The SEDOL for the security corresponding to this lot as provided by the FI.
FI_SUPPLIED_ISIN		CHAR12	The ISIN for the security corresponding to this lot as provided by the FI.
NAME		CHAR512	Name of the security associated with this lot record.
UNITS		NUMBER	Number of units in this lot. Maximum of 6 decimal places.
COST_BASIS		NUMBER (CURRENCY)	The fully-adjusted cost basis of the tax lot, incorporating all applicable adjustments such as commissions, fees, taxes, stock splits, wash sales, return of capital events, amortization, etc. If the financial institution provides this value, it will be passed through. If not supplied, it will be calculated as: $COST_BASIS = COST_PER_SHARE \times UNITS$. Minimum of 2 decimal places and maximum of 4.
ORG_COST_BASIS		NUMBER (CURRENCY)	The gross amount of the opening transaction. If the financial institution provides this value, it will be passed through. If not supplied, it will be calculated as: $ORG_COST_BASIS = ORG_COST_PER_SHARE \times UNITS$. Minimum of 2 decimal places and maximum of 4.
COST_PER_SHARE		NUMBER (CURRENCY)	Represents the per-unit cost of the tax lot, fully adjusted for commissions, fees, taxes, stock splits, wash sales, return of capital events, amortization, etc. If the financial institution provides this value, it will be passed through. If not supplied, it will be calculated as: $COST_PER_SHARE = COST_BASIS / UNITS$. Minimum of 2 decimal places and maximum of 4.

Column Header	Required?	Data Type	Description
UNIT_PRICE		NUMBER (CURRENCY)	Current market price. Minimum of 2 decimal places and maximum of 4.
TOTAL_AMOUNT		NUMBER (CURRENCY)	Current market value. Minimum of 2 decimal places and maximum of 4.
PURCHASE_DATE		DATE	The date the lot was originally purchased.
PRIOR_BUSINESS_DAY_DATE		DATE	The effective business day date for the file contents.
COVERED		CHAR20	The FI Supplied designation of whether the lot is covered or uncovered.
ADJUSTMENT_INDICATOR		CHAR64	Represents the value the FI provided indicating if this lot has been adjusted. The adjusted cost basis value is populated in the COST_BASIS field.
LOT_ID		CHAR60	FI-supplied lot identifier.
ORIGINAL_FACE		NUMBER	The original face or par value for a security that amortizes or accretes (e.g., a mortgage). For such securities, the “current face” is available in the UNITS field. Minimum of 2 decimal places and maximum of 4.
UNREALIZED_GAIN_LOSS		NUMBER (CURRENCY)	The unrealized gain or loss of the tax lot. If the financial institution provides this value, it will be passed through as-is. If not supplied, the value is derived based on the following priority order: <ul style="list-style-type: none"> ▪ TOTAL_AMOUNT COST_BASIS ▪ TOTAL_AMOUNT – UNADJ_COST_BASIS (if COST_BASIS is unavailable) ▪ TOTAL_AMOUNT – ORIG_COST_BASIS (if neither COST_BASIS nor UNADJ_COST_BASIS is available) Minimum of 2 decimal places and maximum of 4.
UNADJ_COST_BASIS		NUMBER (CURRENCY)	The net cost of the opening transaction, adjusted for commissions, fees, and taxes. This value reflects the initial net amount paid for the tax lot at the time of purchase. Minimum of 2 decimal places and maximum of 4.

Column Header	Required?	Data Type	Description
ORG_COST_PER_SHARE			The per-unit purchase price of the original transaction. If the financial institution provides this value, it will be passed through. If not supplied, it will be calculated as: $ORG_COST_PER_SHARE = ORG_COST_BASIS / UNITS$ Minimum of 2 decimal places and maximum of 4.
DAYS_HELD		NUMBER	Represents the number of calendar days the tax lot has been held since its purchase. If the financial institution provides this value, it is passed through as-is.
FI_SUPPLIED_LOT_HOLDING_PERIOD		CHAR64	The financial institution supplied holding period classification of the tax lot.
LOT_HOLDING_PERIOD		CHAR20	The normalized holding period classification of the tax lot, based on the value supplied by the financial institution. Value is either SHORT, LONG, or OTHER.
ACQUISITION_DATE		DATE	The date the tax lot was acquired, either through an original purchase or via another method (e.g., transfer, corporate action). If the FI provides a purchase date, it is used as the ACQUISITION_DATE.

Closed Lot Tax File

This table describes the BAACLOSEDLOTS_<FI_ID>_<date>_<DeliveryStyle>.csv file.

If the custodian data does not fit within the defined field, it will be truncated.

Column Header	Required?	Data Type	Description
ACCOUNT_NUMBER	√	CHAR128	The number of the account that contains this lot.
SECURITY_TYPE		CHAR20	This is the security type value provided by the custodian, if available.
FI_SUPPLIED_CUSIP		CHAR20	The CUSIP for the security corresponding to this lot as provided by the FI.
FI_SUPPLIED_TICKER		CHAR32	The ticker symbol for the security corresponding to this lot as provided by the FI.

Column Header	Required?	Data Type	Description
FI_SUPPLIED_SEDOL		CHAR7	The SEDOL for the security corresponding to this lot as provided by the FI.
FI_SUPPLIED_ISIN		CHAR12	The ISIN for the security corresponding to this lot as provided by the FI.
NAME		CHAR512	Name of the security associated with this lot record.
UNITS	√	NUMBER	Number of units represented by the lot. Maximum of 6 decimal places.
COST_BASIS		NUMBER (CURRENCY)	The fully-adjusted cost basis of the tax lot, incorporating all applicable adjustments such as commissions, fees, taxes, stock splits, wash sales, return of capital events, amortization, etc. If the financial institution provides this value, it will be passed through. If not supplied, it will be calculated as: $COST_BASIS = COST_PER_SHARE \times UNITS$ Minimum of 2 decimal places and maximum of 4.
ORG_COST_BASIS		NUMBER (CURRENCY)	The gross amount of the opening transaction. If the financial institution provides this value, it will be passed through. If not supplied, it will be calculated as: $ORG_COST_BASIS = ORG_COST_PER_SHARE \times UNITS$ Minimum of 2 decimal places and maximum of 4.
REALIZED_GAIN_LOSS	√	NUMBER (CURRENCY)	Gain or loss on the lot. Minimum of 2 decimal places and maximum of 4.
COST_PER_SHARE		NUMBER (CURRENCY)	Represents the per-unit cost of the tax lot, fully adjusted for commissions, fees, taxes, stock splits, wash sales, return of capital events, amortization, etc. If the financial institution provides this value, it will be passed through. If not supplied, it will be calculated as: $COST_PER_SHARE = COST_BASIS / UNITS$ Minimum of 2 decimal places and maximum of 4.

Column Header	Required?	Data Type	Description
CLOSING_PRICE	√	NUMBER (CURRENCY)	The price used to dispose of the lot. Minimum of 2 decimal places and maximum of 4.
TOTAL_AMOUNT	√	NUMBER (CURRENCY)	The net proceeds from closing the lot. Minimum of 2 decimal places and maximum of 4.
PURCHASE_DATE		DATE	The date the lot was originally purchased.
CLOSE_DATE	√	DATE	The date the lot was closed.
LOT_HOLDING_PERIOD		CHAR20	The normalized holding period classification of the tax lot, based on the financial institution supplied value. Value is either SHORT, LONG, or OTHER.
PRIOR_BUSINESS_DAY_DATE		DATE	The effective business day date for the file contents.
FI_SUPPLIED_TX_TYPE		CHAR64	The FI supplied transaction type.
LOSS_DISALLOWED	√	NUMBER (CURRENCY)	Any disallowed loss due to wash sales (or other reasons). Minimum of 2 decimal places and maximum of 4.
CLOSE_ACCOUNTING_METHOD		CHAR26	Method to be used when closing the lot as provided by the FI.
UNKNOWN_COST_BASIS	√	BOOLEAN	A flag indicating that the source did not have the cost basis when the lot was closed. Value is either 1 (indicating TRUE) or 0 (indicating FALSE).
LOT_ID		CHAR60	FI-supplied lot identifier.
ORIGINAL_FACE		NUMBER	The original face or par value for a security that amortizes or accretes (e.g., a mortgage). For such securities, the “current face” is available in the UNITS field. Minimum of 2 decimal places and maximum of 4.

Column Header	Required?	Data Type	Description
UNREALIZED_GAIN_LOSS		NUMBER (CURRENCY)	The unrealized gain or loss of the tax lot. If the financial institution provides this value, it will be passed through as-is. If not supplied, the value is derived based on the following priority order: <ul style="list-style-type: none"> ▪ TOTAL_AMOUNT - COST_BASIS ▪ TOTAL_AMOUNT - UNADJ_COST_BASIS (if COST_BASIS is unavailable) ▪ TOTAL_AMOUNT - ORIG_COST_BASIS (if neither COST_BASIS nor UNADJ_COST_BASIS is available) Minimum of 2 decimal places and maximum of 4.
UNADJ_COST_BASIS		NUMBER (CURRENCY)	The net cost of the opening transaction, adjusted for commissions, fees, and taxes. This value reflects the initial net amount paid for the tax lot at the time of purchase. <p>Minimum of 2 decimal places and maximum of 4.</p>
ORG_COST_PER_SHARE		NUMBER (CURRENCY)	The per-unit purchase price of the original transaction. If the financial institution provides this value, it will be passed through. If not supplied, it will be calculated as: $\text{ORG_COST_PER_SHARE} = \text{ORG_COST_BASIS} / \text{UNITS}$ <p>Minimum of 2 decimal places and maximum of 4.</p>
DAYS_HELD		NUMBER	Represents the number of calendar days the tax lot has been held since its purchase. If the financial institution provides this value, it is passed through as-is.
FI_SUPPLIED_LOT_HOLDING_PERIOD		CHAR64	The financial institution supplied holding period classification of the tax lot.
ACQUISITION_DATE		DATE	The date the tax lot was acquired, either through an original purchase or via another method (e.g., transfer, corporate action). If the FI provides a purchase date, it is used as the ACQUISITION_DATE.