

FIX Trading Community OTC Product Committee

Recommended Practices for Party Entitlement

OTC Markets: Foreign Exchange instruments

Editor(s): Etrading Software Limited
Date: Wednesday, 30 April 2014
Version: 1.0 Final
Paper designation: Analysis

DISCLAIMER

THE INFORMATION CONTAINED HEREIN AND THE FINANCIAL INFORMATION EXCHANGE PROTOCOL (COLLECTIVELY, THE "FIX PROTOCOL") ARE PROVIDED "AS IS" AND NO PERSON OR ENTITY ASSOCIATED WITH THE FIX PROTOCOL MAKES ANY REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, AS TO THE FIX PROTOCOL (OR THE RESULTS TO BE OBTAINED BY THE USE THEREOF) OR ANY OTHER MATTER AND EACH SUCH PERSON AND ENTITY SPECIFICALLY DISCLAIMS ANY WARRANTY OF ORIGINALITY, ACCURACY, COMPLETENESS, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SUCH PERSONS AND ENTITIES DO NOT WARRANT THAT THE FIX PROTOCOL WILL CONFORM TO ANY DESCRIPTION THEREOF OR BE FREE OF ERRORS. THE ENTIRE RISK OF ANY USE OF THE FIX PROTOCOL IS ASSUMED BY THE USER.

NO PERSON OR ENTITY ASSOCIATED WITH THE FIX PROTOCOL SHALL HAVE ANY LIABILITY FOR DAMAGES OF ANY KIND ARISING IN ANY MANNER OUT OF OR IN CONNECTION WITH ANY USER'S USE OF (OR ANY INABILITY TO USE) THE FIX PROTOCOL, WHETHER DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL (INCLUDING, WITHOUT LIMITATION, LOSS OF DATA, LOSS OF USE, CLAIMS OF THIRD PARTIES OR LOST PROFITS OR REVENUES OR OTHER ECONOMIC LOSS), WHETHER IN TORT (INCLUDING NEGLIGENCE AND STRICT LIABILITY), CONTRACT OR OTHERWISE, WHETHER OR NOT ANY SUCH PERSON OR ENTITY HAS BEEN ADVISED OF, OR OTHERWISE MIGHT HAVE ANTICIPATED THE POSSIBILITY OF, SUCH DAMAGES.

DRAFT OR NOT RATIFIED PROPOSALS (REFER TO PROPOSAL STATUS AND/OR SUBMISSION STATUS ON COVER PAGE) ARE PROVIDED "AS-IS" TO INTERESTED PARTIES FOR DISCUSSION ONLY. PARTIES THAT CHOOSE TO IMPLEMENT THIS DRAFT PROPOSAL DO SO AT THEIR OWN RISK. IT IS A DRAFT DOCUMENT AND MAY BE UPDATED, REPLACED, OR MADE OBSOLETE BY OTHER DOCUMENTS AT ANY TIME. THE FPL GLOBAL TECHNICAL COMMITTEE WILL NOT ALLOW EARLY IMPLEMENTATION TO CONSTRAIN ITS ABILITY TO MAKE CHANGES TO THIS DOCUMENT PRIOR TO FINAL RELEASE. IT IS INAPPROPRIATE TO USE FPL WORKING DRAFTS AS REFERENCE MATERIAL OR TO CITE THEM AS OTHER THAN "WORKS IN PROGRESS". THE FPL GLOBAL TECHNICAL COMMITTEE WILL ISSUE, UPON COMPLETION OF REVIEW AND RATIFICATION, AN OFFICIAL STATUS ("APPROVED") TO THE PROPOSAL AND A RELEASE NUMBER.

No proprietary or ownership interest of any kind is granted with respect to the FIX Protocol (or any rights therein).

Copyright 2003-2013 FIX Protocol Limited, all rights reserved

Table of Contents

DISCLAIMER	2
DOCUMENT HISTORY	5
EXECUTIVE SUMMARY	6
Background	6
Recommended Practices Guidelines	7
1 OBJECTIVE	9
2 SCOPE	10
2.1 In Scope	10
2.2 Out of Scope	10
3 TARGET AUDIENCE	11
4 AUTHORS	12
5 MARKET CONVENTIONS	13
5.1 Foreign Exchange Market Structure	13
5.2 Terminology	15
5.3 Entitlement convention	18
6 SYSTEM DIAGRAMS	20
6.1 Overview Diagram	20
6.2 Activity Diagram	21
6.3 Entitlements Data Model Diagram	22
7 D2C WORKFLOWS	23
7.1 Message Flows Summary	23
7.2 Scenario WF1: New buy-side request – accepted	25
7.3 Scenario WF2: New buy-side request – rejected or deferred	27
7.4 Scenario WF3: Sell-side setup a new buy-side	29
7.5 Scenario WF4: Amendment request (incl. new fund/account) - accepted	30
7.6 Scenario WF5: Amendment request (incl. new fund/account) – rejected or deferred	32
7.7 Scenario WF6: Amendment by sell-side - accepted	34
7.8 Scenario WF7: Amendment by sell-side – partially accepted and partially rejected	35
7.9 Scenario WF8: Amendment by sell-side – rejected	38
7.10 Scenario WF9: Reports snapshot	39
7.11 Scenario WF10: Reports update	41
7.12 Scenario WF11: Suspension of client by sell-side	43
7.13 Scenario WF12: Off-board a client by request	44
7.14 Scenario WF13: Off-board or Suspend a Firm, User or Account	45
8 PRIME BROKER WORKFLOWS	47
8.1 Message Flows Summary	47
8.2 Scenario WF.PB.14: Prime broker: New buy-side client request	49
8.3 Scenario WF.PB.15: Prime broker: New buy-side setup	50

8.4 Scenario WF.PB.16: Prime broker: New buy-side setup - rejected	51
8.5 Scenario WF.PB.17: Prime broker Amendment	52
8.6 Scenario WF.PB.18: Prime broker activate / deactivate user.....	53
8.7 Scenario WF.PB.19: Prime broker Off-boarding a client	54
9 STREAM ASSIGNMENT WORKFLOWS	55
9.1 Message Flows Summary	55
9.2 Scenario SA.1: Buy-side client requests to be assigned to a Pricing Stream – sell-side assigns	56
9.3 Scenario SA.2: Buy-side client requests to be assigned to a Pricing Stream – sell-side rejects.....	58
9.4 Scenario SA.3: Sell-side assigns a buy-side client to a Pricing Stream	60
9.5 Scenario SA.4: Sell-side terminates (un-assigns) a Pricing Stream from the buy-side client	61
10 MISCELLANEOUS WORKFLOWS	62
10.1 Message Flows Summary	62
10.2 Scenario WF.M.20: Sell-Side User Login	63
10.3 Scenario WF.M.21: Sell-Side User Logoff.....	64
10.4 Scenario WF.M.22: Sell-Side Requests Instrument types and sub-types	65
10.5 Scenario WF.M.23: Sell-Side Requests a Snapshot of Buy-Side Entities/Users	66
10.6 Scenario WF.M.24: Sell-Side Requests Snapshot + Updates of Buy-Side Entities/Users.....	68
10.7 Scenario WF.M.25: Sell side sends a message to the buy-side.....	70
11 MESSAGE DETAIL	71
11.1 UserRequest (35=BE)	71
11.2 UserResponse (35=BF).....	71
11.3 Email (35=C)	72
11.4 StreamAssignmentRequest (35=CC)	75
11.5 StreamAssignmentReport (35=CD).....	77
11.6 StreamAssignmentReportAck (35=CE).....	79
11.7 PartyDetailsGrp component Diagram.....	80
11.8 PartyDetailListRequest (35=CF).....	81
11.9 PartyDetailsListReport (35=CG)	83
11.10 PartyDetailsListUpdateReport (35=CK).....	87
11.11 PartyEntitlementsRequest (35=CU).....	91
11.12 PartyEntitlementsReport (35=CV).....	97
11.13 PartyEntitlementsUpdateReport (35=CZ).....	106
11.14 PartyEntitlementsDefinitionRequest (35=DA)	113
11.15 PartyEntitlementsDefinitionRequestAck (35=DB).....	121
APPENDIX I – ENTITLEMENT ATTRIBUTES	128
APPENDIX II – DOCUMENT FIGURES	129
APPENDIX III – DOCUMENT TABLES.....	130

Document History

Date	Change	Who	Version
20 September 2013	Initial Version	Yuval Cohen, Etrading Software	0.1
08 October 2013	Minor corrections	Sonali Tandon, Etrading Software	0.2
23 October 2013	Updated section 11.8.2 Party Entitlement Report and 11.10.2 with Party Entitlement Definition Request FIXML Example Updated section 5.3.1 Scope the Entitlement by trading feature with enumerations for new FPL tag Clean document post few reviewers comments	Sonali Tandon, Etrading Software	0.3
31 October 2013	Incorporate comments form Lisa Taikitsadaporn	Yuval Cohen, Etrading Software	0.4
5 November 2013	Incorporate further comments form Lisa Taikitsadaporn	Yuval Cohen, Sonali Tandon Etrading Software	0.5
26 November 2013	Incorporated corrections to references to FIX Trading Community	Sonali Tandon, Etrading Software	0.6
29 November 2013	Merge comments from J. Hurley Added <i>stream assignments workflows</i> section	Yuval Cohen, Etrading Software	0.7
21 February 2014	Merge comments from Lisa Taikitsadaporn	Yuval Cohen, Etrading Software	0.8
2 April 2014	Merge new tags and added numbers for EntitlementAttributes	Simon Wiltshire, Etrading Software	1.0

Executive Summary

Background

FIX Protocol

FIX Trading Community is the non-profit, industry-driven standards body at the heart of global trading. FIX Trading Community is independent and neutral, dedicated to addressing the real business and regulatory issues impacting multi-asset trading in global markets through standardisation, delivering operational efficiency, increased transparency, and reduced costs and risks for all market participants.

Central to FIX Trading Community's work is the continuous development and promotion of the FIX Protocol messaging language, which has revolutionised the electronic trading environment and has successfully become the way the world trades.

The FIX Protocol specification is a comprehensive description of a set of messages and fields that can be exchanged between trading parties in order to trade electronically. Much work has gone into the specification to ensure that it covers a wide set of scenarios, including specific work to cover foreign exchange as well as fixed income trading flows.

In the nearly 20 years of the FIX Protocol standard, the protocol has been synonymous with equities electronic trading and has become the de facto standard used by equities trading systems globally. FIX has evolved over those years to continually support the needs of the global user community across different user groups and several different asset classes.

Reasons for the Creation of the Recommended Practices Document

Currently, Foreign Exchange (FX) client enablement – the process by which sell-side firms enable clients to trade with them on execution venues (like ECNs) – is a manual and labour-intensive process. While a lot of resources and focus have gone into optimising and standardising trading workflows, the area of client enablement has attracted very little interest from market participants so far. As a result, each execution venue has been left to provide its own solution for client enablement resulting in dealers having to implement labour intensive, ad-hoc procedures to get this vital step right.

In January 2013, the Trading Enablement Standardisation Initiative (TESI) was launched by the global investment banking community to engage with execution venues and other stakeholders to promote the FIX Protocol for client and trader enablement on electronic trading platforms. This work resulted in the *Recommended Practices for Party Entitlement, OTC Markets: Fixed income instruments*.

In June 2013 the Trading Enablement Standardisation Initiative (TESI) decided to extend the work for the foreign exchange markets.

This Recommended Practices Document defines a standard set of guidelines based on Party Reference Data messages for exchanging enablement information for trading both Foreign Exchange as well as Fixed Income OTC instruments.

These guidelines, once implemented, are expected to bring the following benefits to the stakeholders:

- Faster client enablement
- Increased operational efficiency
- Reduction in STP failures
- Help meet more stringent operational regulatory requirements

As Dodd-Frank in the US and MiFID II in Europe push OTC derivatives trading onto Swap Execution Facilities (SEFs) and Organised Trading Facilities (OTFs) respectively, these Recommended Practices will help ensure that these trading venues can adopt the FIX Protocol for enablement in a consistent and standardised manner. This standardised entitlement protocol increases market efficiency by allowing market participants to communicate and complete the entitlement process quickly and cost efficiently.

Work of the FIX Trading Community - Global Cross Asset Committee

In September 2013, FIX Trading Community launched an initiative to create the Foreign Exchange recommended practices for party entitlement for the OTC markets in response to a request by a group of 7 global Foreign Exchange dealers (Trading Enablement Standardisation Initiative for Foreign Exchange [TESI-FX]).

The initial focus was to standardise the OTC Foreign Exchange instrument entitlement process by creating an industry agreed set of recommended FIX practices. These documents were compiled by FIX Trading Community – GCAC.

The document is currently (2014) maintained by the OTC Product Committee.

Recommended Practices Guidelines

Principles behind the Recommended Practices

The primary focus of the recommended practices is to provide detailed guidelines for the implementation of party entitlement messages for Foreign Exchange OTC electronic markets. The committee members, comprising the majority of the global Foreign Exchange dealers and Foreign Exchange trading venues, based their recommendations on real world enablement workflows in use today.



The starting point for each scenario was identifying the business workflow and then recommending the set of FIX messages and fields to use for that workflow.

The approach resulted in several gaps being identified in the FIX Protocol for Foreign Exchange trading. The committee submitted relevant gap analysis documents to the FIX Trading Community - Global Technical Committee (GTC) for enhancements to the FIX Protocol and these were all approved and incorporated into the FIX 5.0 SP2 specification. This approach has ensured that this Recommended Practices document is a practical FIX implementation guide that can be used immediately by all the major Foreign Exchange execution venues.

Overview of Recommendations

The recommended practices for party entitlement for Foreign Exchange OTC markets contain the following sections:

- Objective
- Scope (In scope and out of scope)
- Market Conventions
- Diagrams: overview diagram, activity diagram and data model diagram
- Workflows

-
- Message details
 - Entitlement attributes

1 Objective

The objective of this document is to define an agreed sub-set of permissible message implementations and, in doing so, provide a definitive statement as to how these messages should be implemented in practice to support Dealer to Clients (D2C) Foreign Exchange (FX) party entitlements workflows.

2 Scope

This section will set out what is in scope for the entitlement messaging requirements as well as highlighting what is not in scope (although this list will not be exhaustive).

2.1 In Scope

- Venue type: D2C Execution Venues in the Foreign Exchange domain
- Instrument scope: FX-Spot, FX-Forwards, FX-Swaps, FX-Option, NDF's and Precious Metal contracts which trade as currency pairs.
- Workflows: Required workflows for managing buy-side user entitlements by sell-side firms trading through electronic D2C execution venues
 - Ability to set entitlements for one or more buy-side users. Such entitlements pertain to either trading permissions or price viewing permissions, or both
- Communication parties: Electronic communication between execution venues to sell-side firms that acts either as liquidity providers or prime brokers

2.2 Out of Scope

- Communication between execution venues and buy-side users
- Entitlement or on-boarding of clearing or settlement firms
- Direct communication between sell-side and buy-side (i.e. not routed through the execution venue)
- The preliminary process by which a sell-side firm may initially view and identify a new buy-side firm or specific buy-side user
- Entitlements on D2D Execution Venues
- Communication with intermediate 3rd party firms or intermediate firms (e.g. aggregators, clearing house, clearing/settlement facilitators etc.)

3 Target Audience

Parties interested in implementing the FIX Protocol for exchanging foreign exchange parties reference data between the sell-side and a D2C Execution Venue.

4 Authors

Documents created during 2013 by the FIX Trading Community Global Cross Asset Committee with assistance from Etrading Software.

The document is maintained by the FIX Trading Community OTC product committee.

5 Market Conventions

5.1 Foreign Exchange Market Structure

Today, in the foreign exchange markets, brokers execute a proportion of transactions with their buy-side clients through electronic venues. These electronic venues are known as Dealer to Client (D2C) venues.

At each electronic trading venue, the broker has to enable trading permissions for all their counterparties and their respective traders. This process of entitlement management is currently conducted manually through graphical user interfaces provided by each of the venues, each different from the other, as shown in Figure 1. The arrows indicate the information flow required to setup, monitor and maintain trading permissions in the market:

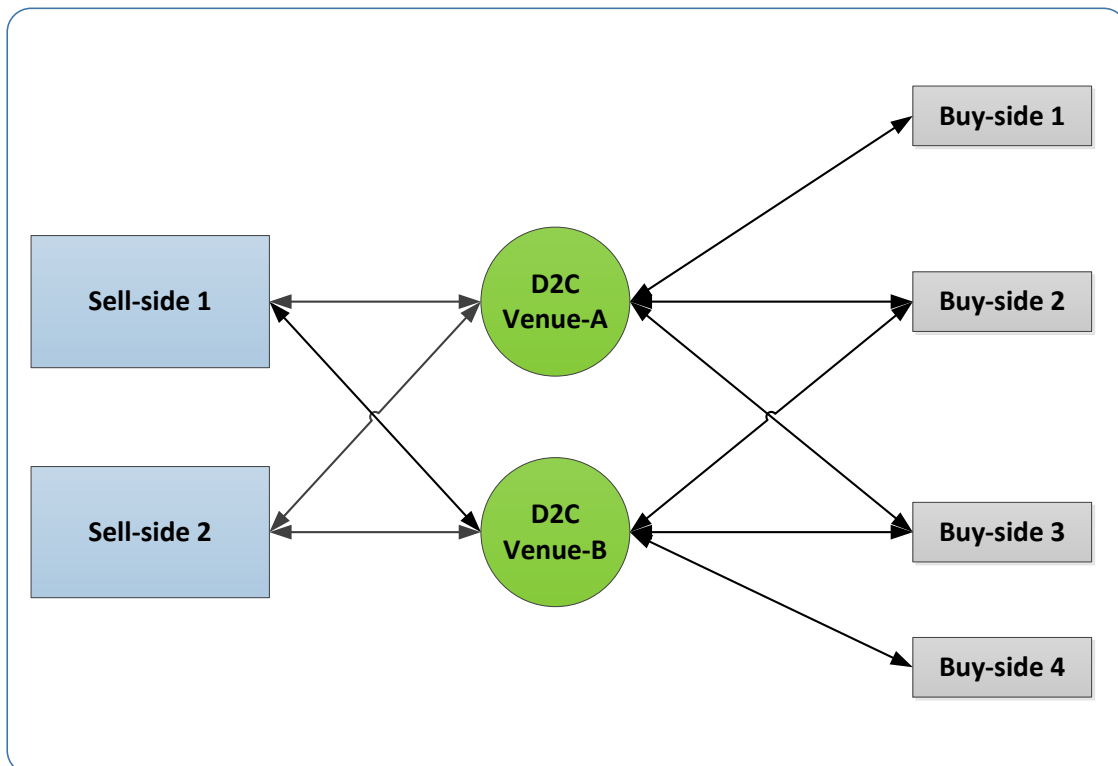


Figure 1: D2C Venues - Entitlement Information Flow

In some venues, all buy-side clients access the market through one or more prime broker. Figure 2 describes the entitlement communication flow in such venues:

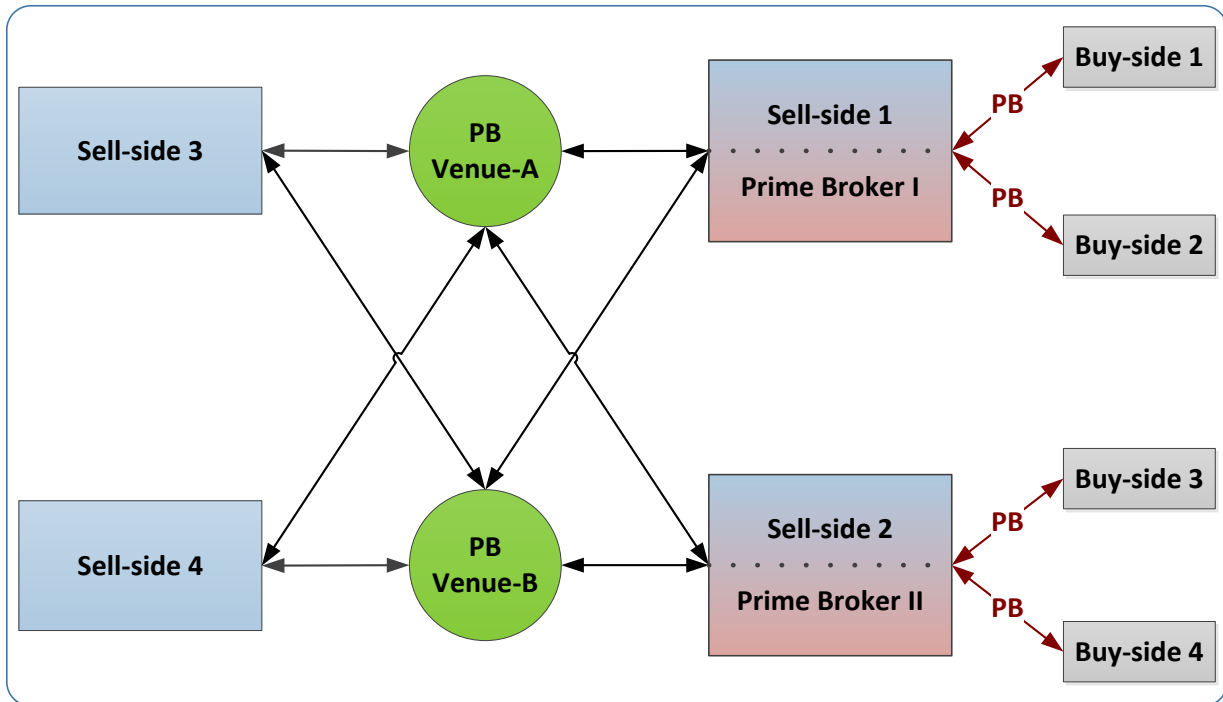


Figure 2: Prime Broker Venues - Entitlement Information Flow

In addition, to ensure that a dealer can service its clients' orders, it will execute transactions on Dealer to Dealer (D2D) venues between brokers to facilitate liquidity (sometimes referred to as inter-dealer activity).

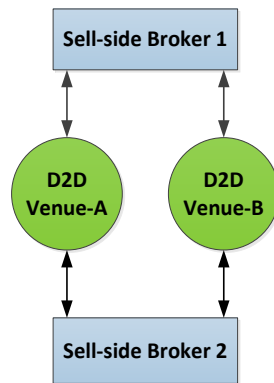


Figure 3: D2D venues

In practice there is more complex structure where sell-side can act as both the Liquidity Provider and prime broker for separate counterparties. Venue may connect to buy-side, Prime Broker and Liquidity Providers. The following diagram illustrates such complex Venues.

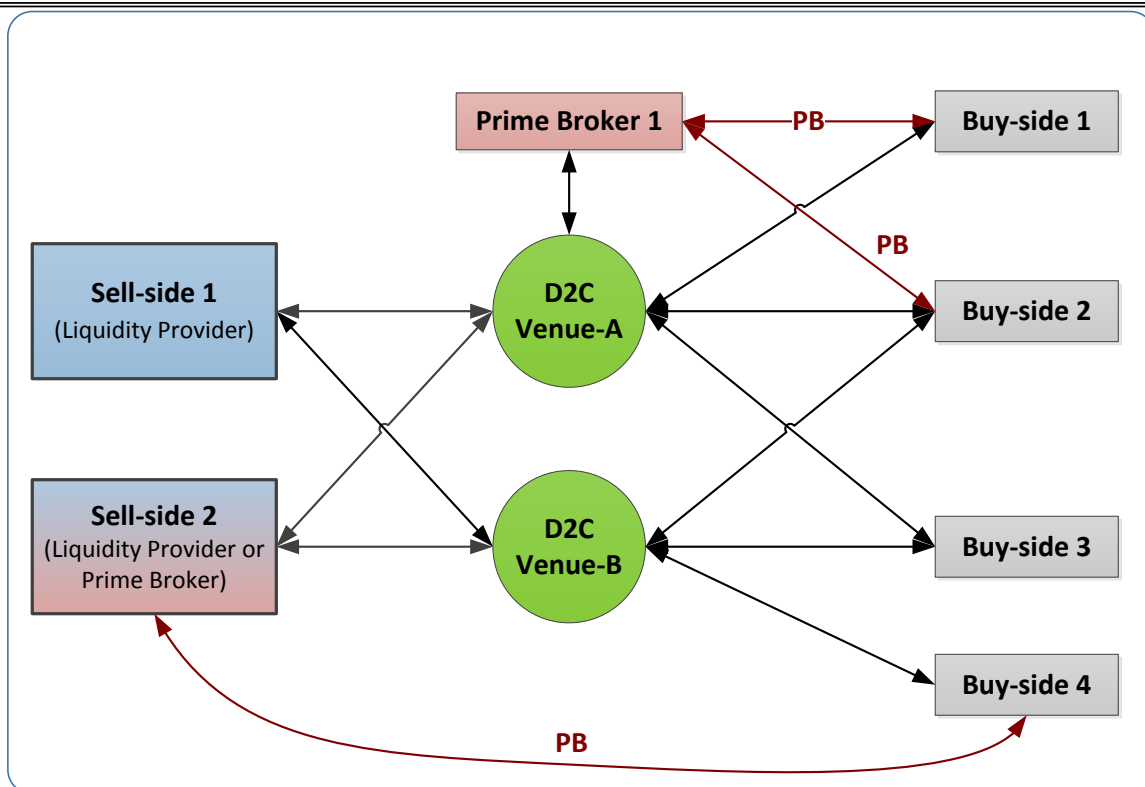


Figure 4: Complex venues combining buy-side, Prime Broker and Liquidity Provider

5.2 Terminology

The terminology used commonly in the industry to describe the entities and attributes associated with trader enablement differs from that used in the FIX messaging standard. This section provides the mapping between the two by indicating the FIX tags and values and how they relate to the various entities in the data model.

5.2.1 Sell-Side Terminology

This section describes how the terminology relating to the sell-side maps onto the FIX messaging standard. Each subsection defines a single attribute and indicates the FIX tags and values necessary to fully specify this sell-side attribute

5.2.1.1 Sell-Side Identifier

Sell-side firms usually operate as multiple entities. Such entities may represent different branches (e.g. divided by geographical location), legal entities, departments (e.g. investment bank, retail bank, liquidity provider, prime brokerage, etc.), and so on. During the entitlement process, it is necessary to define which sell-side entity is entitled for the relationship. Usually, one or more people at the sell-side may be authorized to retrieve and/or modify entitlements for multiple entities.

5.2.1.2 Sales people

Each entitlement may be associated with a primary sales person from the sell-side firm, and optionally a list of additional sales people. Each Sales person may have an ID at the Execution Venue as well as name and possibly contact number.

5.2.2 Buy-Side Terminology

This section describes how the terminology relating to the buy-side maps onto the FIX messaging standard. Each subsection defines a single attribute and indicates the FIX tags and values necessary to fully specify this buy-side attribute.

The FX entitlement may be assigned to a buy-side firm or a buy-side user. In both cases, they are uniquely identified by a single buy-side identifier

5.2.2.1 Buy-Side ID

The buy-side id is used to identify the buy-side firm or user within party reference data messages. Buy-side ID is set in the PartyDetailGrp or the RelatedPartyDetailGrp.

For simplicity the following sections assumes that buy-side attributes are populated in the PartyDetailGrp

5.2.2.2 Legal Entity Identifier

The legal entity identifier (LEI) is required by the CFTC regulations. In the FIX standard it is specified using the three tags PartyDetailID, PartyDetailIDSource and PartyDetailRole as follows:

Tag	Field Name	Value	Comment
1691	PartyDetailID	<LEI>	<i>Legal Entity Identifier</i>
1692	PartyDetailIDSource	N	Source of the identifier (N=LEI)
1693	PartyDetailRole	13	Identifies the type or role (13=Order Origination Firm)

5.2.2.3 Client Account

The client account is a trading account that is set by a sell-side firm in order to identify a buy-side account. Some buy-side entities may have multiple client accounts. In the FIX standard it is specified using the three tags PartyDetailID, PartyDetailIDSource and PartyDetailRole as follows:

Tag	Field Name	Value	Comment
1691	PartyDetailID	<Account ID>	<i>Account ID</i>
1692	PartyDetailIDSource	D	Source of the identifier (D=Proprietary)
1693	PartyDetailRole	24	Identifies the type or role (24=Customer Account)

5.2.2.4 Additional Buy-Side Attributes

Further attributes defining properties of the buy-side user may be defined using the FIX tags in the PartyDetailGrp component. Using these tags it is possible to provide values for the buy-side user name, firm, department, contact telephone number, email address and postal address, as shown below:

Tag	Field Name	Value	Comment
1691	PartyDetailID	<value>	Sub-identifier for the party
1692	PartyDetailIDSource	9=Contact name 1=Firm 24=Department 7=Phone number 8=Email address 6=Postal address	Type of PartyDetailSubID value

5.2.3 Product/Security Type Terminology

This section describes how the terminology relating to products and sectors maps onto the FIX messaging standard. Each subsection defines a single attribute and indicates the FIX tags and values necessary to fully specify this product/sector attribute.

5.2.3.1 Product

Foreign Exchange security may use the Product(460) field with value: 4(CURRENCY).

5.2.3.2 Security Type

Foreign Exchange security type are enumerated in SecurityType(167) field as:

- FXNDF: Non-deliverable forward
- FXSPOT: FX Spot
- FXFWD: FX Forward
- FXSWAP: FX Swap
- OPT: Option

5.2.3.3 Tier

The tier level is used to specify the pricing tiers that the sell-side firm offers to the buy-side users. The FIX tag [MDStreamID\(1500\)](#) is used to identify the price stream.

5.3 Entitlement convention

See [Appendix I](#) for the list of attributes used for entitlements:

5.3.1 Scope the Entitlement by trading feature

Some venues support the ability to scope entitlements by trading feature. A buy-side user may be entitled to all, part or none of the following trading features:

- Submit quote requests (RFQ)
 - Single dealer RFQ
 - Multi-dealer RFQ
- Submit requests for streaming quotes (RFS)
- Receive indicative prices - but not entitled to trade them
- Receive executable prices - but not entitled to trade them
- Receive streaming prices and trade them (Hit/Lift)
- Submit an order
- Submit allocation

The best practices recommends using **EntitlementSubType(2402)** in addition to **EntitlementType(1775)** in order to organise the entitlement types. **EntitlementSubType(2402)** enumerations used in OTC markets FX instruents are listed below:

Business Enumeration	EntitlementType(1775)	EntitlementSubType(2402)
Submit quote requests (RFQ)	7 = Submit quote requests	Single Quote
Submit requests for streaming quotes for limited period (RFS)		Streaming Quotes
Single Dealer RFQ		Single Broker quote request
Multi Dealer RFQ		Multi Brokers quote request
Receive indicative prices - but not entitled to trade them	5 = Subscribe to market data	View indicative prices
Receive executable prices - but not entitled to trade them		View executable prices
Receive streaming prices and trade them (Hit/Lift)	0 = Trade	Hit/Lift
Trade / Submit an order		Order entry

Table 1: EntitlementSubType(2402) proposed enumeration

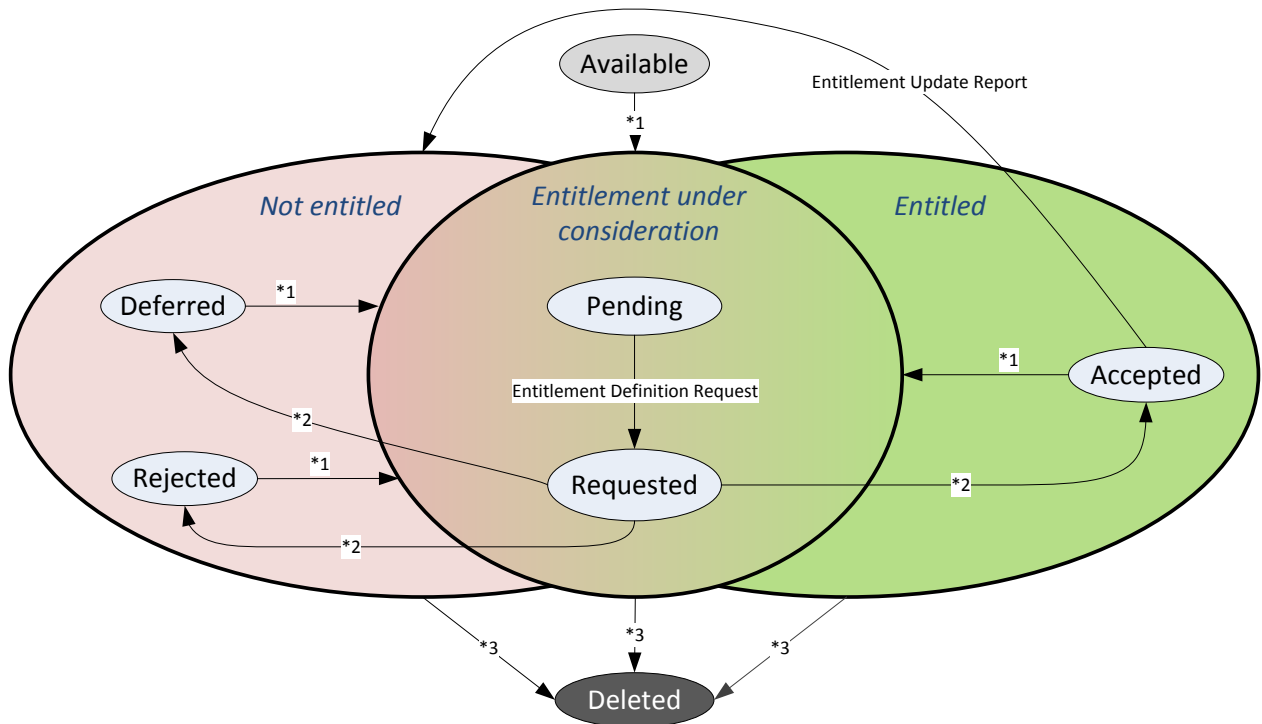
- Additional sub-type enumerations can be assigned by the execution venue
- EntitlementSubType(2402) is a sub-attribute of EntitlementType(1775) and appears in same messages and components as EntitlementType(1775)

5.3.2 Entitlement Status

The value of EntitlementStatus(1883) indicates the status of an entitlement. The values may be one of:

- Accepted(0)
- Accepted with changes(1)
- Rejected(2)
- **Deferred(5)**
- Pending(3) [Waiting for an entitlement definition request to be submitted]
- Requested(4) [Entitlement definition has been requested and is currently being processed]

The following diagram describes the states of entitlements:



Legend:

***1:** one of:

- | | | |
|-----------------------------------|--------------------|----------------------------------|
| ○ Entitlement Request; | venue → sell-side; | Destination state is 'Pending' |
| ○ Entitlement Update Report; | venue → sell-side | Destination state is 'Pending' |
| ○ Entitlement Definition Request; | sell-side → venue | Destination state is 'Requested' |

***2:** one of:

- | | |
|-------------------------------|-------------------|
| ○ Entitlement Definition Ack; | venue → sell-side |
| ○ Entitlement Update Report; | venue → sell-side |

***3:**

- | | |
|------------------------------|-------------------|
| ○ Entitlement Update Report; | venue → sell-side |
|------------------------------|-------------------|

Figure 5: Entitlement Status transition

6 System Diagrams

6.1 Overview Diagram

The following diagram illustrates the FIX messages and the Workflows described in later sections.

D2C Party Entitlement Flows

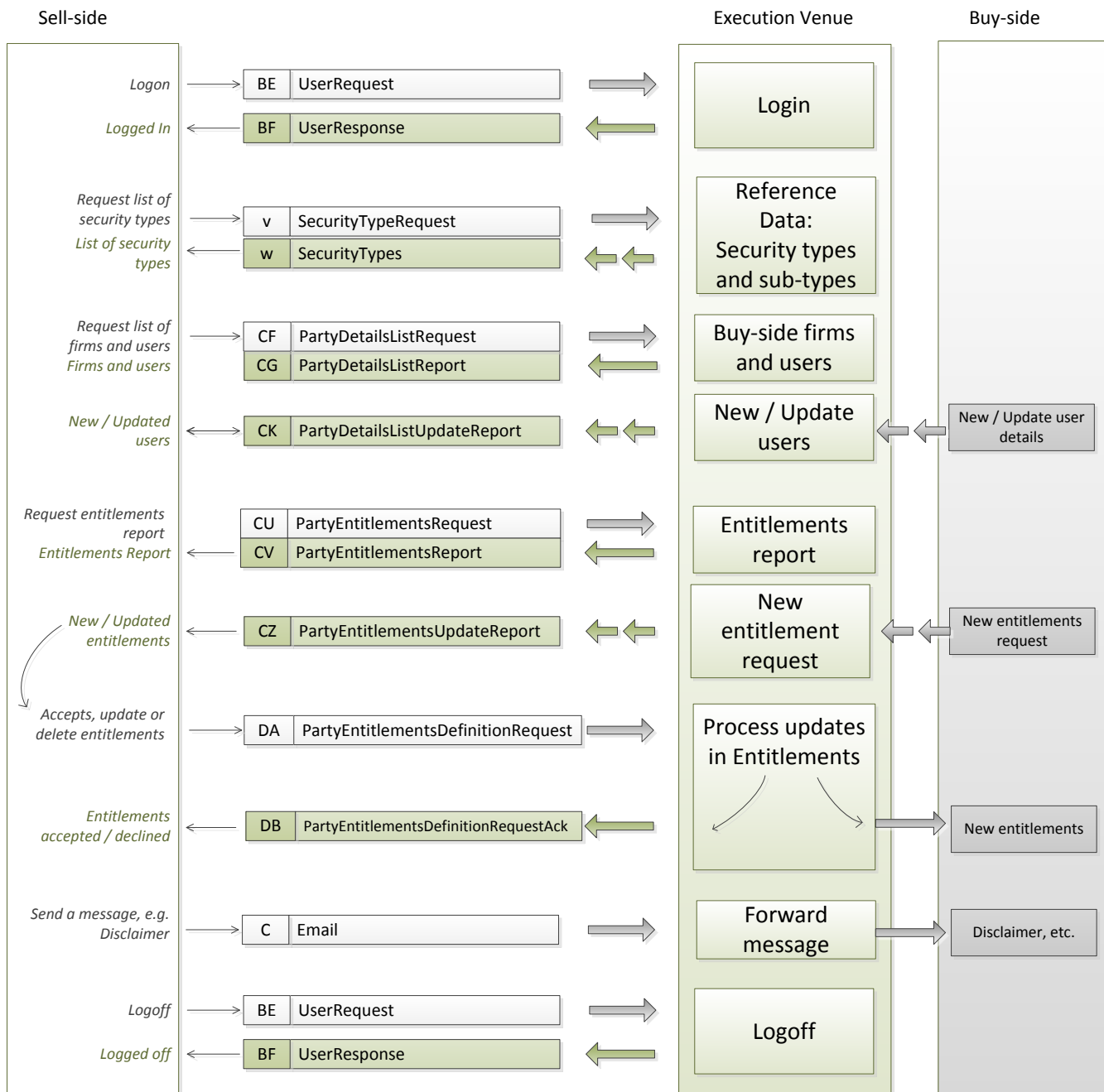


Figure 6: Overview Diagram

6.2 Activity Diagram

The activity diagram illustrates the FIX messages that will be used to manage Party Entitlements. The use of each message is explained in the context of the workflows later in this document.

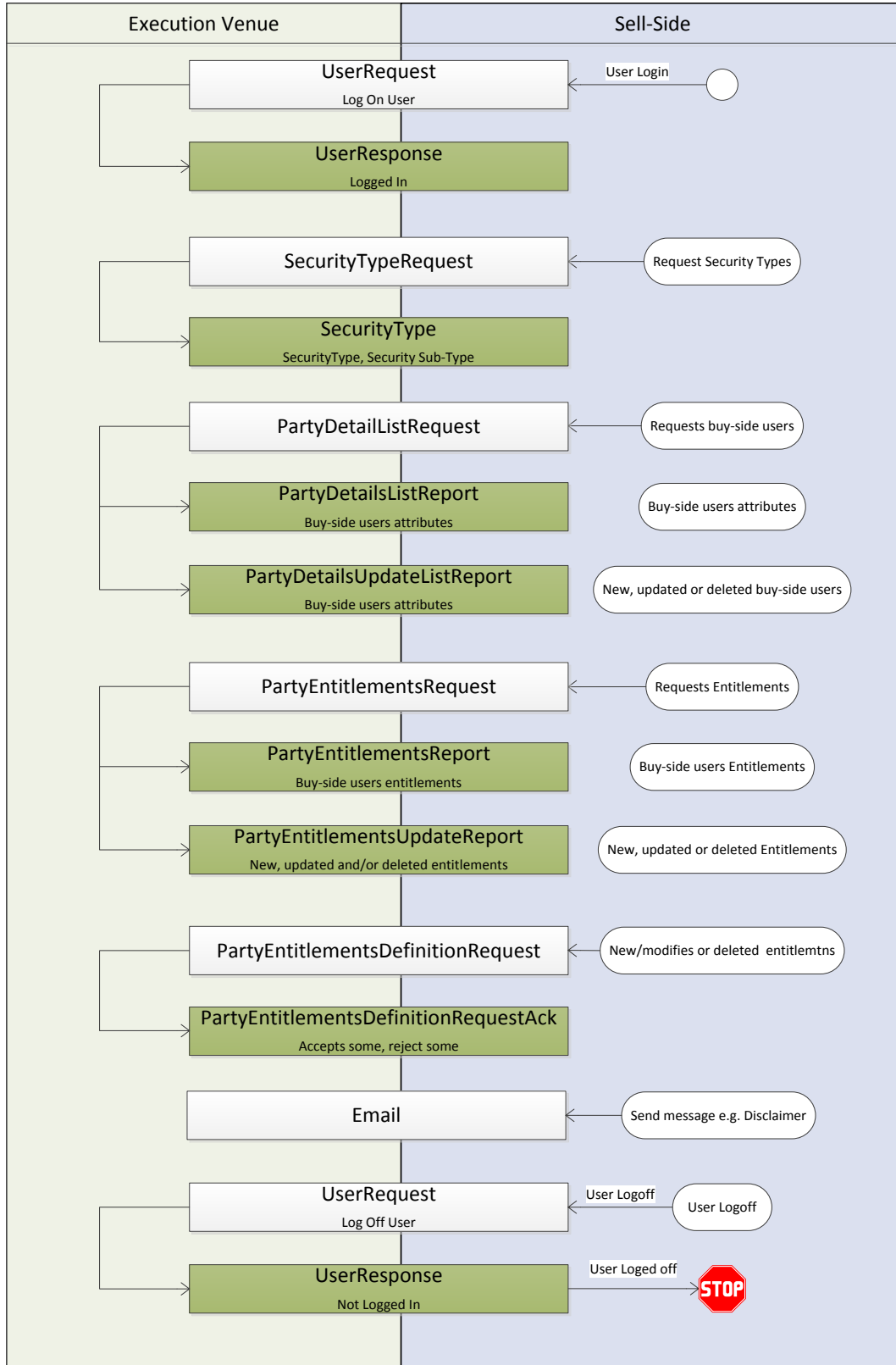


Figure 7: Activity Diagram

6.3 Entitlements Data Model Diagram

Chapter 3, Market Conventions, described the various components used to describe Party Entitlements. The following diagram illustrates how they are organised within the data model comprising:

1. Sell-side Firm, with sell-side entity, its sales people and the private client account codes it uses to describe its clients
2. Execution venue and the instruments it trades described as products and sectors
3. Buy-side firm and the individual users that the sell-side trades with
4. Parties entitlement describing the relationship between the sell-side, buy-side and venue on which the transaction takes place

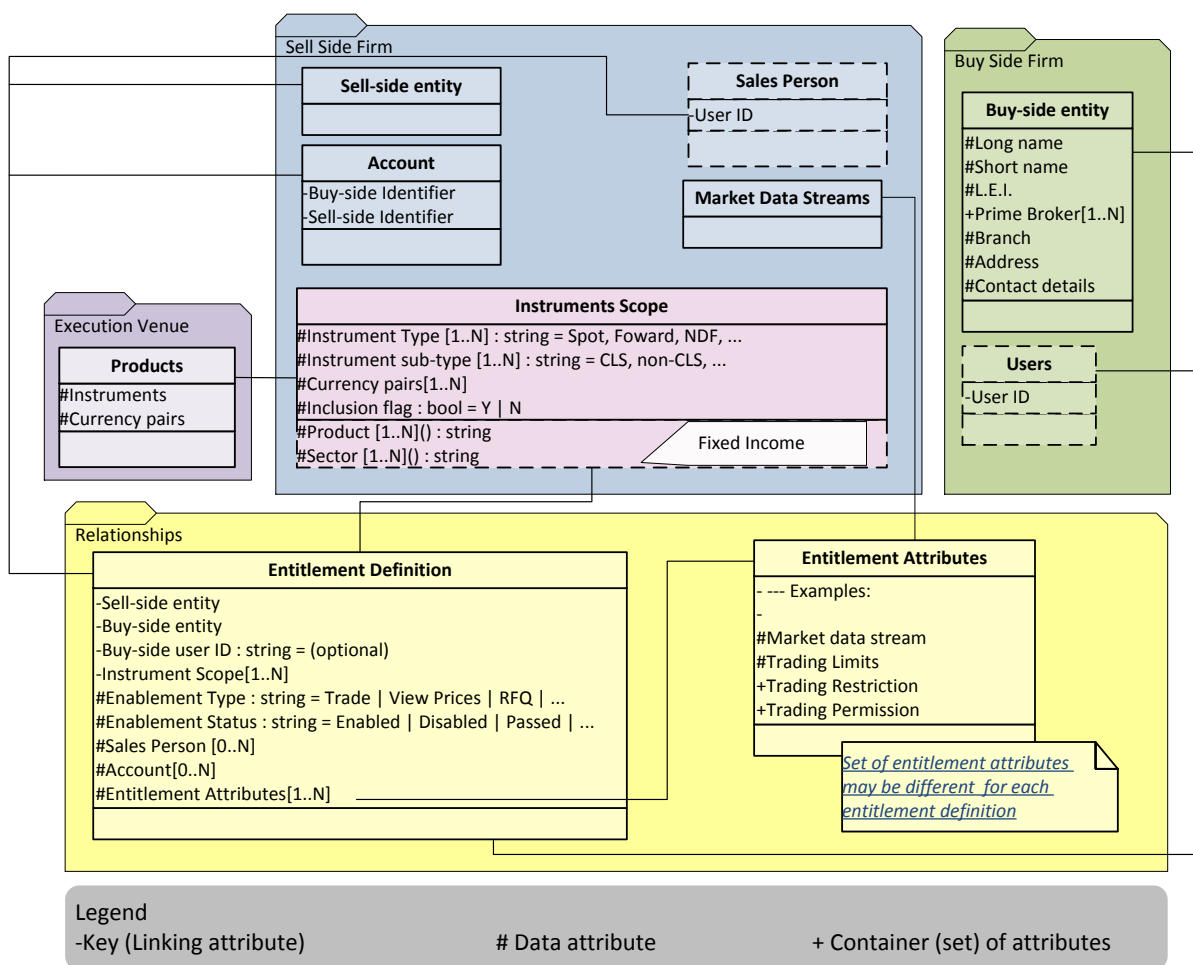


Figure 8: Entitlements Data Model Diagram

7 D2C Workflows

7.1 Message Flows Summary

This section details the various scenarios identified as requirements to be supported in communication between execution venues and sell-side liquidity providers. Each scenario is described at a high level followed by the FIX messages used to support the scenario.

For a given implementation of these scenarios, Execution Venues need to take into consideration having the proper controls in place to safeguard the information and determining whether the requester is permitted to see or retrieve the information. Further discussions on this topic are out of scope of this document.

Scenario summary by category:

Category	Workflow initiated by venue	Workflow initiated by sell-side
New buy-side client setup	<ul style="list-style-type: none"> • WF.01: New buy-side request – accepted • WF.02: New buy-side request – rejected or deferred 	<ul style="list-style-type: none"> • WF.03: Sell-side setup a new buy-side
Relationship amendments (single or multiple)	<ul style="list-style-type: none"> • WF.04: Amendment request (incl. new fund/account) – accepted • WF.05: Amendment request (incl. new fund/account) – rejected or deferred 	<ul style="list-style-type: none"> • WF.06: Amendment by sell-side – accepted • WF.07: Amendment by sell-side – partially accepted and partially rejected • WF.08: Amendment by sell-side – rejected
Disable entitlement and off-board	<ul style="list-style-type: none"> • WF13: Off-board or Suspend a Firm, User or Account 	<ul style="list-style-type: none"> • WF.11: Suspension of client by sell-side • WF12: Off-board a client by request
Report	<ul style="list-style-type: none"> • WF.10: Reports update 	<ul style="list-style-type: none"> • WF.09: Reports snapshot

Scenario summary by section:

Scenarios

7.2 Scenario WF1: New buy-side request – accepted 25

7.3 Scenario WF2: New buy-side request – rejected or deferred..... 27

7.4 Scenario WF3: Sell-side setup a new buy-side 29

7.5 Scenario WF4: Amendment request (incl. new fund/account) - accepted 30

7.6 Scenario WF5: Amendment request (incl. new fund/account) – rejected or deferred 32

7.7 Scenario WF6: Amendment by sell-side - accepted 34

7.8 Scenario WF7: Amendment by sell-side – partially accepted and partially rejected 35

7.9 Scenario WF8: Amendment by sell-side – rejected 38

7.10 Scenario WF9: Reports snapshot 39

7.11 Scenario WF10: Reports update 41

7.12 Scenario WF11: Suspension of client by sell-side 43

7.13 Scenario WF12: Off-board a client by request..... 44

7.14 Scenario WF13: Off-board or Suspend a Firm, User or Account 45

7.2 Scenario WF1: New buy-side request – accepted

This scenario illustrates the case where the Execution Venue sends a request to the sell-side on behalf of one or more buy-side users for entitlement which is subsequently approved.

Such a request may contain:

- Buy-side client information including venue’s identifiers for the buy-side client
- Type of entitlement requested (e.g. RFS, Streaming, Trade)
- Users information and venue identifiers
- Fund/account information and venue identifier

Sell-side performs internal checks. Sell-side respond back to the venue:

- Agreeing to trade with the buy-side client
- The type of entitlement and restrictions
- Any additional information which is supported by the venue (e.g. attributes used by the sell-side to identify the buy-side firm, buy-side users and/or buy-side fund/account)

Venue sets the entitlement definition on the platform and acknowledges the sell-side

- Accordingly notifies the buy-side client they can start trading with sell-side

A preliminary condition is that the sell-side subscribes to receive the requests or ‘actively asks’ (i.e. requests) to receive buy-side requests for entitlements.

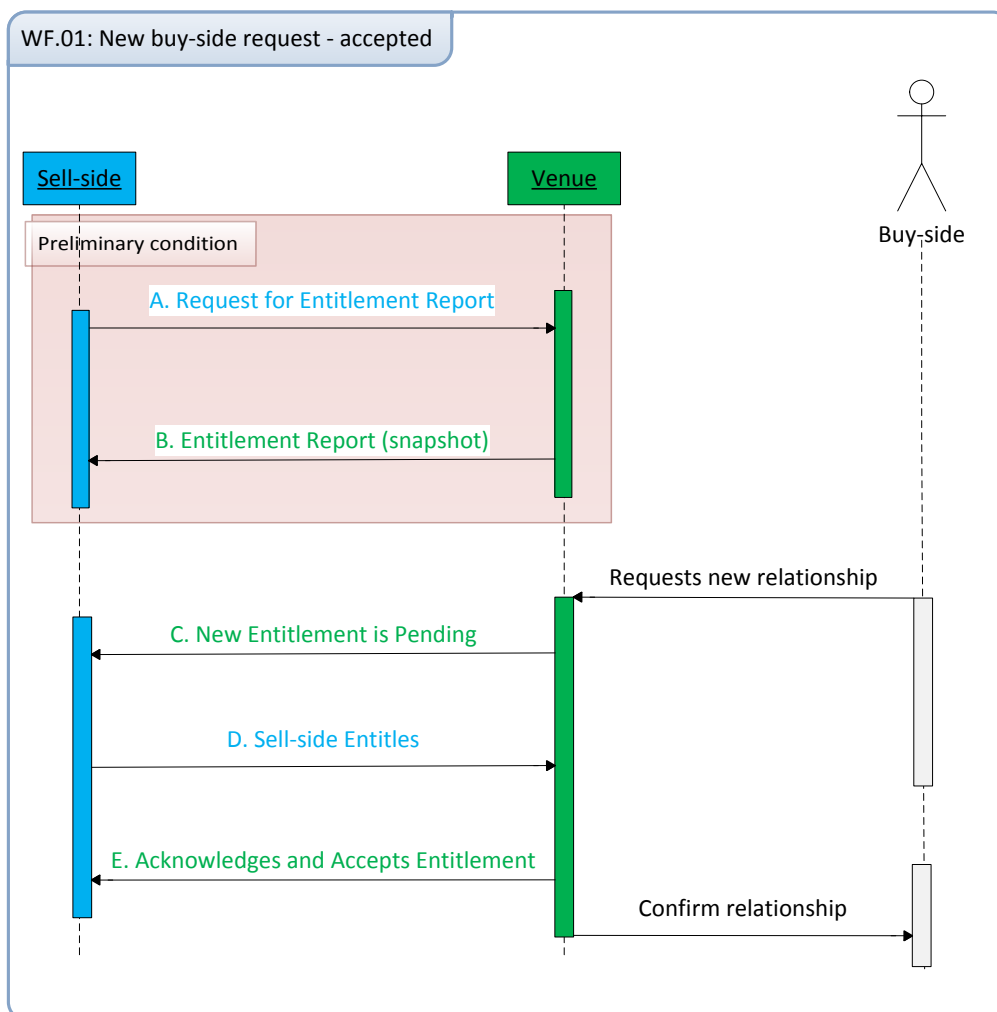


Figure 9: Scenario WF1: New buy-side request – accepted

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0		
(A) Entitlement Report Request	→	CU – PartyEntitlementsRequest EntitlementRequestID(1770)=❶ SubscriptionRequestType(263)=Snapshot + Updates(1) PartyDetailStatus(1672) <optional> MarketSegmentScopeGrp <optional> NoEntitlementType(2345) <optional> -> EntitlementType(1775)
(B) Entitlement Report (Snapshot)	←	CV – PartyEntitlementsReport EntitlementRequestID(1770)=❶ EntitlementReportID(1771)= <required>
(C) New Entitlement is Pending	←	CZ – PartyEntitlementsUpdateReport EntitlementRequestID(1770)=❶ EntitlementReportID(1771)= <required> EntitlementStatus(1883)=Pending(3) NoPartyEntitlements(1772)=1 -> ListUpdateAction(1324)=Add(A) -> PartyDetailsGrp -> NoEntitlements(1773)=M -> -> EntitlementID(1776)=❷
(D) Sell-side Entitles	→	DA – PartyEntitlementsDefinitionRequest EntitlementRequestID(1770)=❸ RequestingPartyGrp NoPartyEntitlements(1772)=1 -> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp -> NoEntitlements(1773)=M -> -> EntitlementIndicator(1774)= Y (i.e. enabled) -> -> EntitlementID(1776)= ❹ -> EntitlementStatus(1883)=Accepted(0)
(E) Execution Venue Accepts and Acknowledges	←	DB – PartyEntitlementsDefinitionRequestAck EntitlementRequestID(1770)=❸ PartyDetailStatus(1672)=Active(0) EntitlementRequestStatus(1882)=Accepted(0) EntitlementRequestResult(1881)=successful(0) (default) -> NoEntitlements(1773)=M -> EntitlementStatus(1883)=Accepted(0)

Table 2: Scenario WF1: New buy-side request – accepted

Notes

- The sell-side owns the entitlement information to be set for a given counterparty, whether at the firm level or individual level
- The EntitlementStatus(1883)=Pending(3) is set by the ExecutionVenue; The PartyEntitlementsDefinitionRequest(35=DA) message would be used to request the execution venue to take an action that would result in the EntitlementStatus(1883) being set appropriately (Accepted(0) or Rejected(1)).
- During the period where the sell side sent the PartyEntitlementDefinitionRequest (35=DA) and until the execution venue process and set the entitlement accordingly, the entitlement status may be reported as: EntitlementStatus(1883)=Requested(4).

7.3 Scenario WF2: New buy-side request – rejected or deferred

This scenario illustrates the case where the Execution Venue sends a request to the sell-side on behalf of one or more buy-side users for entitlement which is subsequently declined.

Such a request may contain:

- Buy-side client information including venue identifiers for the buy-side client
- Type of entitlement requested (e.g. RFS, Streaming, Trade)
- Users information and venue identifiers
- Fund/account information and venue identifier

Sell-side perform internal checks, and if relationship with the buy-side client is not agreed to or to be processed at a later stage, sell-side rejects or defers the request.

Venue sets the entitlement definition on the platform (connection and entitlement now exists) and acknowledges the sell-side

- Accordingly notifies the buy-side client

A preliminary condition is that the sell-side subscribes to receive the requests or ‘actively asks’ (i.e. requests) to receive buy-side requests for entitlements.

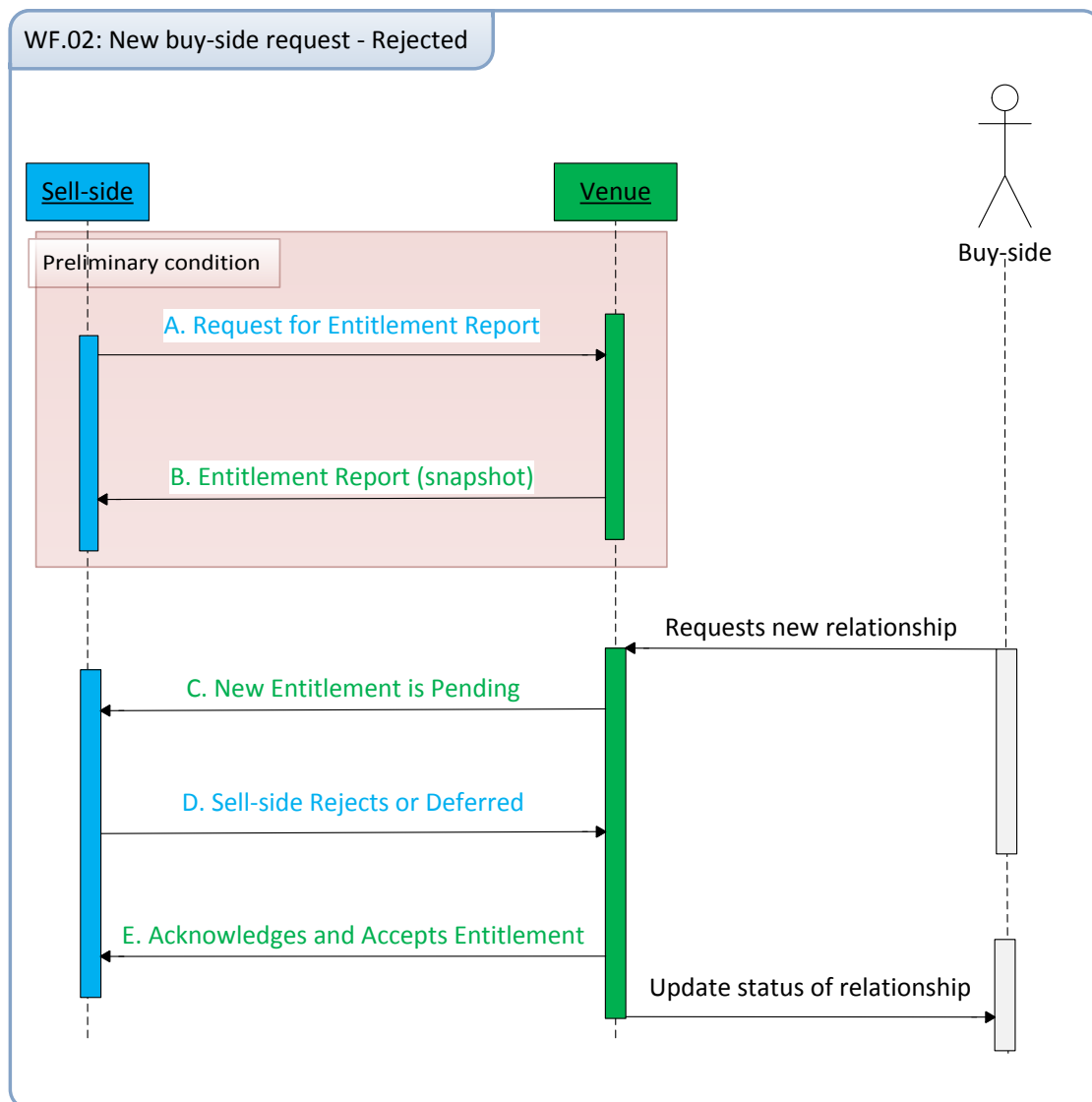


Figure 10: Scenario WF2: New buy-side request – rejected or deferred

Model Flow – significant attributes

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0				
(A) Entitlement report request	Sell-Side	→	CU – PartyEntitlementsRequest EntitlementRequestID(1770)=❶ SubscriptionRequestType(263)=Snapshot + Updates(1) PartyDetailStatus(1672) <optional> MarketSegmentScopeGrp <optional> NoEntitlementType(2345) <optional> -> EntitlementType(1775)	Execution Venue
(B) Entitlement Report (Snapshot)		←	CV – PartyEntitlementsReport EntitlementRequestID(1770)=❶ EntitlementReportID(1771)= <required>	
(C) New Entitlement(s) (Update report)		←	CZ – PartyEntitlementsUpdateReport EntitlementRequestID(1770)=❶ EntitlementReportID(1771)= <required> EntitlementStatus(1883)=Pending(3) NoPartyEntitlements(1772)=1 -> ListUpdateAction(1324)=Add(A) -> PartyDetailsGrp -> NoEntitlements(1773)=M -> -> EntitlementID(1776)=❷	
(D) Sell-side Rejects or Defers entitlement		→	DA – PartyEntitlementsDefinitionRequest EntitlementRequestID(1770)=❸ RequestingPartyGrp NoPartyEntitlements(1772)=1 -> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp -> NoEntitlements(1773)=M -> -> EntitlementIndicator(1774)= N (i.e. rejected) -> -> EntitlementID(1776)= ❹ -> EntitlementStatus(1883)=Rejected(2) or Deferred (5)	
(E) Execution Venue acknowledges		←	DB – PartyEntitlementsDefinitionRequestAck EntitlementRequestID(1770)=❸ PartyDetailStatus(1672)=Active(0) EntitlementRequestStatus(1882)=Accepted(0) EntitlementRequestResult(1881)=successful(0) (default) -> NoEntitlements(1773)=M -> EntitlementStatus(1883)=Rejected(2) or Deferred (5)	

Table 3: Scenario WF2: New buy-side request – Rejected or deferred

Notes

- See [WF1 notes](#)
- EntitlementStatus(1883)=Deferred may later be amended to accepted(0) or Rejected(2) by the sell-side. See [WF6](#) for further example

7.4 Scenario WF3: Sell-side setup a new buy-side

This scenario describes a sell-side setting up (i.e. entitles) a new buy-side user on the venue. The scenario usually follows communication (i.e. over the phone or email) between the sell-side and the buy-side.

Sell-side initiates entitlement definition request to venue

- Such a request may contain
 - Request to trade with the buy-side client
 - The type of entitlement and restrictions
 - Any additional information which is supported by the venue (e.g. Bank identifiers for the buy-side client)

Venue sets the entitlement definition on the platform (system)

- For 'reject' case see [WF8](#)

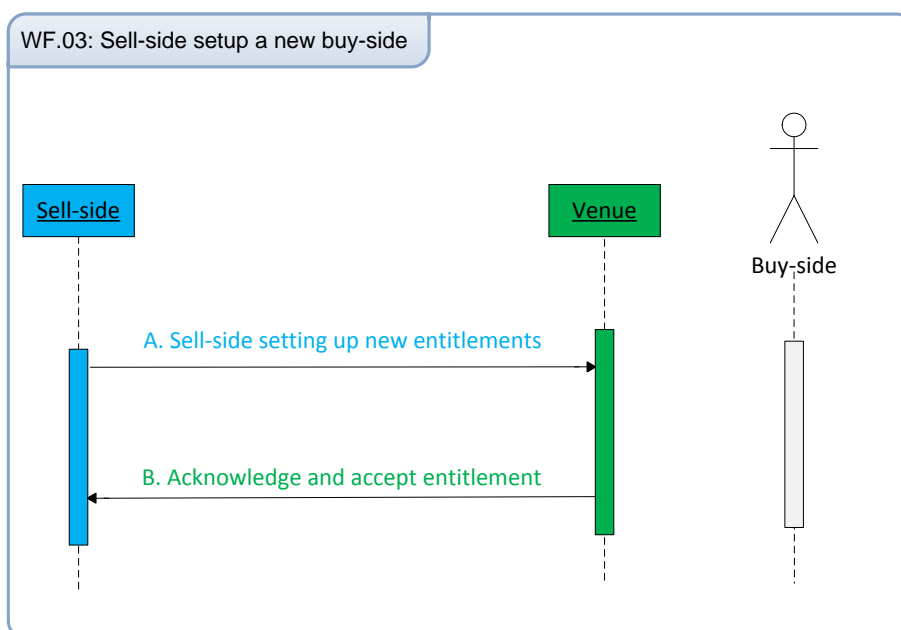


Figure 11: Scenario WF3: Sell-side setup a new buy-side

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0				
(A) Sell-side setting up new entitlements	→	Sell-Side	DA – PartyEntitlementsDefinitionRequest EntitlementRequestID(1770)=❶ RequestingPartyGrp NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Add(A) -> PartyDetailGrp -> NoEntitlements(1773)=M ->-> EntitlementIndicator(1774)=Y N (i.e.: enabled disabled) ->-> EntitlementID(1776)=<Identifying this entitlement>	Execution Venue
(B) Execution Venue acknowledges and accepts	←		DB – PartyEntitlementsDefinitionRequestAck EntitlementRequestID(1770)=❶ EntitlementRequestStatus(1882)=Accepted(0) EntitlementRequestResult(1881)=successful(0) (default) -> NoEntitlements(1773)=M -> EntitlementStatus(1883)=Accepted(0)	

Table 4: Scenario WF3: Sell-side setup a new buy-side

7.5 Scenario WF4: Amendment request (incl. new fund/account) - accepted

This scenario illustrates the case where the Execution Venue sends a request to the sell-side on behalf of one or more buy-side users for entitlement amendment which is subsequently approved.

- See [WF1](#) for more details

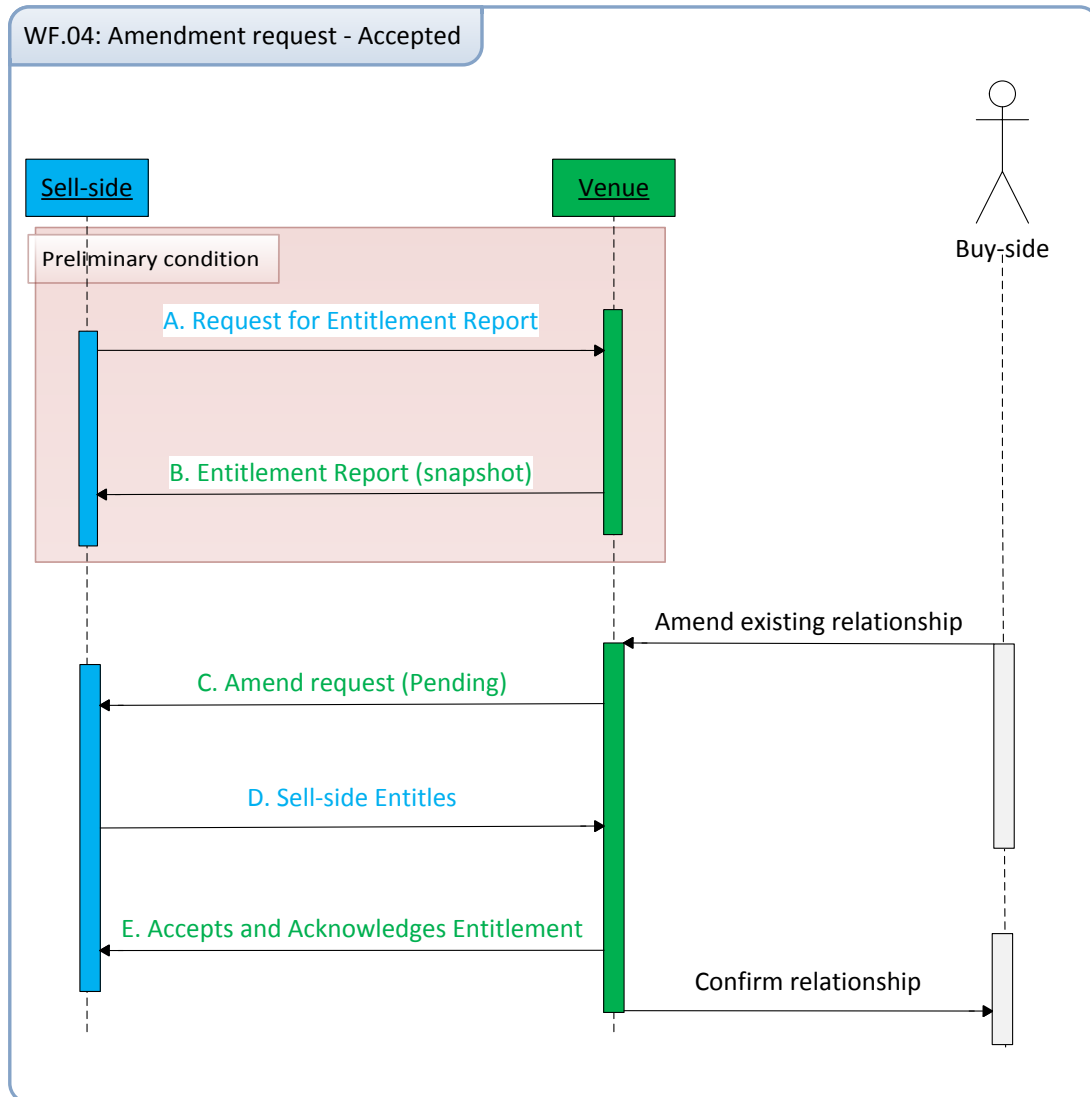


Figure 12: Scenario WF4: Amendment request (incl. new fund/account) - accepted

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0		
(A) Entitlement report request	→	CU – PartyEntitlementsRequest EntitlementRequestID(1770)=❶ SubscriptionRequestType(263)=Snapshot + Updates(1) PartyDetailStatus(1672) <optional> MarketSegmentScopeGrp <optional> NoEntitlementType(2345) <optional> -> EntitlementType(1775)
(B) Entitlement Report (Snapshot)	←	CV – PartyEntitlementsReport EntitlementRequestID(1770)=❶ EntitlementReportID(1771)= <required>
(C) New Entitlement(s) (Update report)	←	CZ – PartyEntitlementsUpdateReport EntitlementRequestID(1770)=❶ EntitlementReportID(1771)= <required> EntitlementStatus(1883)=Pending(3) NoPartyEntitlements(1772)=1 -> ListUpdateAction(1324)=Modify(M) -> PartyDetailsGrp -> NoEntitlements(1773)=M -> -> EntitlementID(1776)=❷
(D) Sell-side Entitles	→	DA – PartyEntitlementsDefinitionRequest EntitlementRequestID(1770)=❸ RequestingPartyGrp NoPartyEntitlements(1772)=1 -> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp -> NoEntitlements(1773)=M -> -> EntitlementIndicator(1774)= Y (i.e. enabled) -> -> EntitlementID(1776)=❹
(E) Execution Venue acknowledges and accepts	←	DB – PartyEntitlementsDefinitionRequestAck EntitlementRequestID(1770)=❸ PartyDetailStatus(1672)=Active(0) EntitlementRequestStatus(1882)=Accepted(0) EntitlementRequestResult(1881)=successful(0) (default) -> NoEntitlements(1773)=M -> EntitlementStatus(1883)=Accepted(0)

Table 5: Scenario WF4: Amendment request (incl. new fund/account) – accepted

Notes

- See [WF1 notes](#)

7.6 Scenario WF5: Amendment request (incl. new fund/account) – rejected or deferred

This scenario illustrates the case where the Execution Venue sends a request to the sell-side on behalf of one or more buy-side users for entitlement amendment which is subsequently either declined or deferred.

- See [WF1](#) for more details

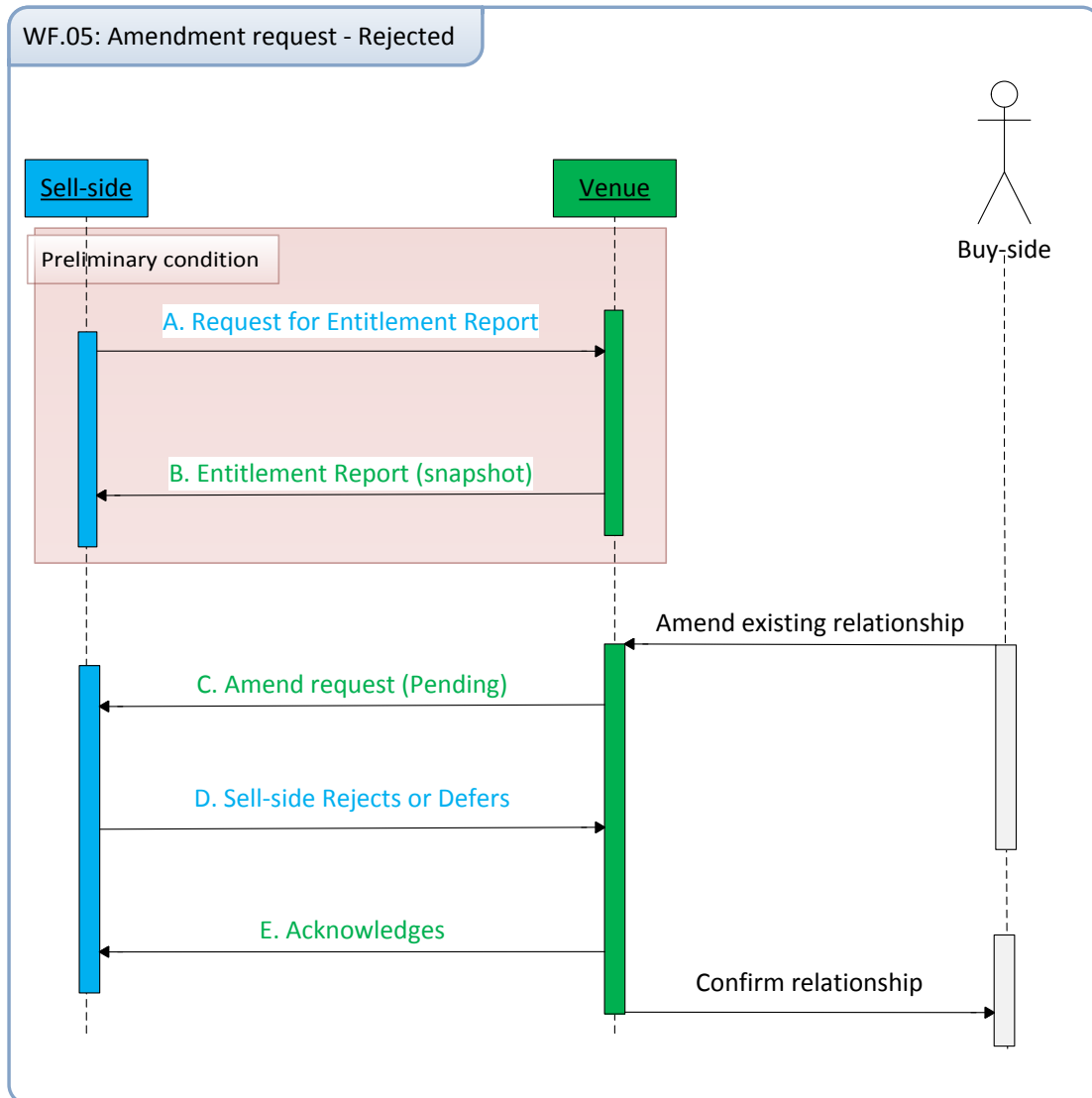


Figure 13: Scenario WF5: Amendment request (incl. new fund/account) – rejected or deferred

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0		
(A) Entitlement report request	→	CU – PartyEntitlementsRequest EntitlementRequestID(1770)=❶ SubscriptionRequestType(263)=Snapshot + Updates(1) PartyDetailStatus(1672) <optional> MarketSegmentScopeGrp <optional> NoEntitlementType(2345) <optional> -> EntitlementType(1775)
(B) Entitlement Report (Snapshot)	←	CV – PartyEntitlementsReport EntitlementRequestID(1770)=❶ EntitlementReportID(1771)= <required>
(C) New Entitlement(s) (Update report)	←	CZ – PartyEntitlementsUpdateReport EntitlementRequestID(1770)=❶ EntitlementReportID(1771)= <required> EntitlementStatus(1883)=Pending(3) NoPartyEntitments(1772)=1 -> ListUpdateAction(1324)=Modify(M) -> PartyDetailsGrp -> NoEntitlements(1773)=M -> -> EntitlementID(1776)=❷
(D) Sell-side Rejects or Defers entitlement	→	DA – PartyEntitlementsDefinitionRequest EntitlementRequestID(1770)=❸ RequestingPartyGrp NoPartyEntitlements(1772)=1 -> ListUpdateAction(1324)=Modify(M) <existing entitlement attributes values> -> -> EntitlementID(1776)=❷ -> EntitlementStatus(1883)=Rejected(2) or Deferred (5)
(E) Execution Venue acknowledges	←	DB – PartyEntitlementsDefinitionRequestAck EntitlementRequestID(1770)=❸ PartyDetailStatus(1672)=Active(0) EntitlementRequestStatus(1882)=Accepted(0) EntitlementRequestResult(1881)=successful(0) (default) -> NoEntitlements(1773)=M -> EntitlementStatus(1883)=Rejected(2) or Deferred (5)

Table 6: Scenario WF5: Amendment request (incl. new fund/account) – rejected or deferred

Notes

- See [WF1 notes](#)
- EntitlementStatus(1883)=Deferred may later be amended to accepted(0) or Rejected(2) by the sell-side. See [WF6](#) for further example

7.7 Scenario WF6: Amendment by sell-side - accepted

This scenario illustrates the case where the sell-side modifies one or more entitlements.

- Some entitlements are enabled
- Other entitlements are disabled

All modifications may be sent in one or multiple messages. The Execution Venue accepts all the modification.

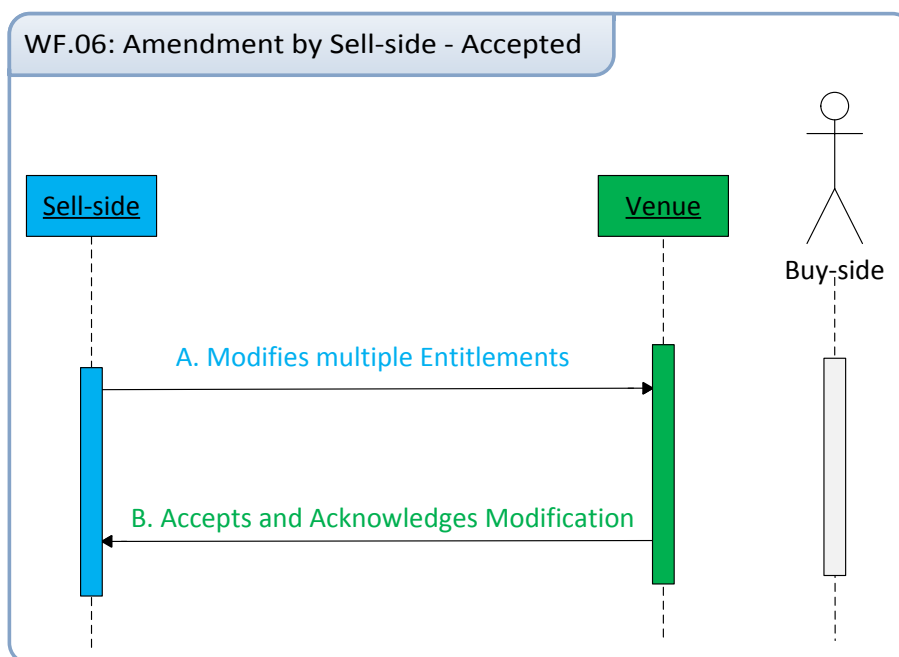


Figure 14: Scenario WF6: Amendment by sell-side - accepted

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0				
(A) Sell-side Modifies multiple entitlements	Sell-Side	→	DA – PartyEntitlementsDefinitionRequest EntitlementRequestID(1770)=❶ RequestingPartyGrp NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp -> NoEntitlements(1773)=M ->-> EntitlementIndicator(1774)=Y N (i.e.: enabled disabled) ->-> EntitlementID(1776)=<Identifying this entitlement>	Execution Venue
(B) Execution Venue accepts and acknowledges		←	DB – PartyEntitlementsDefinitionRequestAck EntitlementRequestID(1770)=❶ EntitlementRequestStatus(1882)=Accepted(0) EntitlementRequestResult(1881)=successful(0) (default) -> NoEntitlements(1773)=M -> EntitlementStatus(1883) <required>	

Table 7: Scenario WF6: Amendment by sell-side – accepted

Note

- Each PartyEntitlementDefinitionRequest may be used to modify multiple entitlements
- PartyEntitlementDefinitionRequest and PartyEntitlementDefinitionRequestAck messages can be fragmented i.e. split across multiple FIX messages

7.8 Scenario WF7: Amendment by sell-side – partially accepted and partially rejected

This scenario illustrates the case where the sell-side modifies multiple entitlements:

- Some entitlements are enabled
- Other entitlements are disabled

All modifications may be sent in one or multiple messages.

The Execution Venue

- Accepts some of the modifications
Rejects the remainder of the modifications

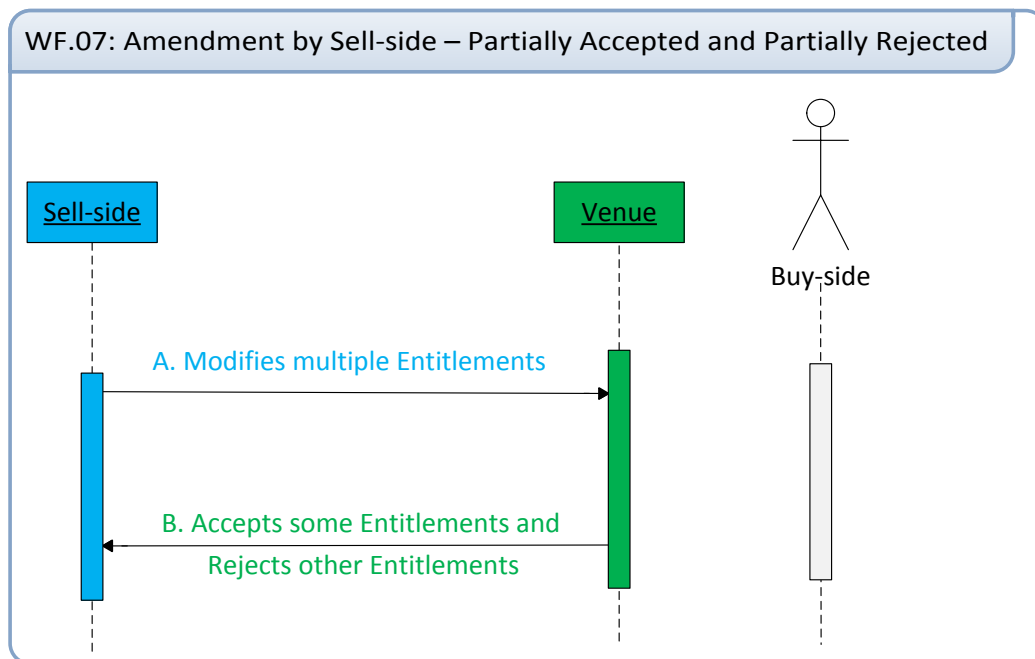


Figure 15: Scenario WF7: Amendment by sell-side – partially accepted and partially rejected

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2. In this example, the sell side modifies three entitlements for two different buy-side users. Two entitlements are accepted and the other entitlement is rejected:

Model FIX 5.0			
(A) Sell-side Modifies multiple entitlements	→	<p>DA – PartyEntitlementsDefinitionRequest</p> <p>EntitlementRequestID(1770)=❶ RequestingPartyGrp NoPartyEntitlements(1772)=3</p> <p>-> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp -> -> NoPartyDetailID(1562) -> -> -> PartyDetailID(1563) = User-1 -> NoEntitlements(1773)=1 -> -> EntitlementIndicator(1774)=Y (i.e. enabled) -> -> EntitlementType(1775)=Subscribe to Market Data(5) -> -> EntitlementID(1776)=❷</p> <p>-> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp -> -> NoPartyDetailID(1562) -> -> -> PartyDetailID(1563) = User-1 -> NoEntitlements(1773)=1 -> -> EntitlementIndicator(1774)=Y (i.e. enabled) -> -> EntitlementType(1775)=Trade(0) -> -> EntitlementID(1776)=❸</p> <p>-> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp -> -> NoPartyDetailID(1562) -> -> -> PartyDetailID(1563) = User-2 -> NoEntitlements(1773)=1 -> -> EntitlementIndicator(1774)=Y (i.e. enabled) -> -> EntitlementType(1775)= Trade(0) -> -> EntitlementID(1776)=❹</p>	Execution Venue

Model FIX 5.0	
<p>(B) Execution Venue accepts some entitlements and rejects the rest</p>	<p>DB – PartyEntitlementsDefinitionRequestAck</p> <p>EntitlementRequestID(1770)=1 EntitlementRequestStatus(1882)=Accepted with changes(1) EntitlementRequestResult(1881)=successful(0) (default) RequestingPartyGrp NoPartyEntitlements(1772)=3</p> <p>-> ListUpdateAction(1324)=Modify(M) -> EntitlementStatus(1883)= Accepted (0) -> EntitlementResult(1884)= Successful (0) -> PartyDetailGrp ->-> NoPartyDetailID(1562) ->->-> PartyDetailID(1563) = User-1 -> NoEntitlements(1773)=1 ->-> EntitlementIndicator(1774)=Y (i.e. enabled) ->-> EntitlementType(1775)= Subscribe to Market Data(5) ->-> EntitlementID(1776)=2</p> <p>← -> ListUpdateAction(1324)=Modify(M) -> EntitlementStatus(1883)=Rejected(2) -> EntitlementResult(1884)=<Required and non-zero> -> PartyDetailGrp ->-> NoPartyDetailID(1562) ->->-> PartyDetailID(1563) = User-1 -> NoEntitlements(1773)=1 ->-> EntitlementIndicator(1774)=Y (i.e. enabled) ->-> EntitlementType(1775)=Trade(0) ->-> EntitlementID(1776)=3</p> <p>-> ListUpdateAction(1324)=Modify(M) -> EntitlementStatus(1883)=Accepted(0) -> EntitlementResult(1884)= Successful (0) -> PartyDetailGrp ->-> NoPartyDetailID(1562) ->->-> PartyDetailID(1563) = User-2 -> NoEntitlements(1773)=1 ->-> EntitlementIndicator(1774)=Y (i.e. enabled) ->-> EntitlementType(1775)= Trade (0) ->-> EntitlementID(1776)=4</p>

Table 8: Scenario WF7: Amendment by sell-side – partially accepted and partially rejected

Notes

- Each PartyEntitlementDefinitionRequest may be used to modify multiple entitlements
- PartyEntitlementDefinitionRequest and PartyEntitlementDefinitionRequestAck messages can be fragmented

7.9 Scenario WF8: Amendment by sell-side – rejected

This scenario illustrates the case where the sell-side modifies multiple entitlements:

- Some entitlements are enabled
- Other entitlements are disabled

All modifications may be sent in one or multiple messages. The Execution Venue rejects all of these modifications.

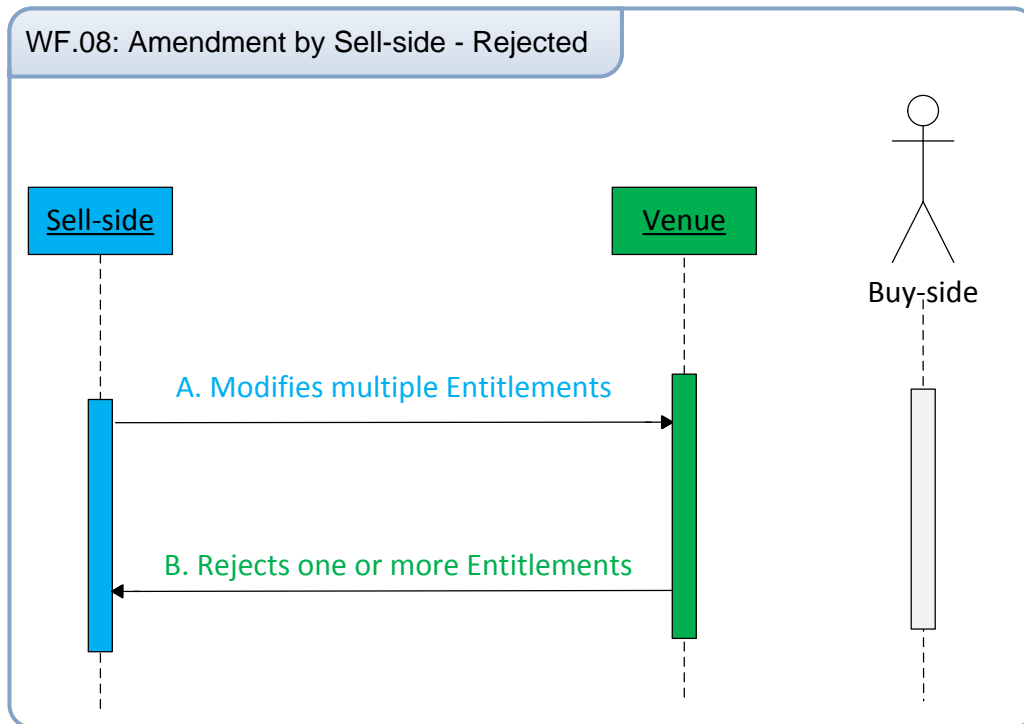


Figure 16: Scenario WF8: Amendment by sell-side – rejected

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0				
(A) Sell-side Modifies multiple entitlements	Sell-Side	→	DA – PartyEntitlementsDefinitionRequest EntitlementRequestID(1770)=❶ RequestingPartyGrp NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp -> NoEntitlements(1773)=M -> -> EntitlementIndicator(1774)=Y N (i.e.: enabled disabled) -> -> EntitlementID(1776)=<Identifying this entitlement>	Execution Venue
(B) Execution Venue rejects		←	DB – PartyEntitlementsDefinitionRequestAck EntitlementRequestID(1770)=❶ EntitlementRequestStatus(1882)=Rejected(2) EntitlementRequestResult(1881)= <required> -> NoEntitlements(1773)=M -> EntitlementStatus(1883)= <required>	

Table 9: Scenario WF8: Amendment by sell-side – rejected

Notes

- See scenario [WF6](#)

7.10 Scenario WF9: Reports snapshot

This scenario illustrates the case where the sell-side sends a request for the list of buy-side users and their entitlement status in the form of party entitlement request. The request message may include filtering criteria to retrieve entitlement data, for example, for all buy-side users, a single buy-side company/branch, a specified product or sector, or a specified entitlement status. The Execution Venue responds by returning a report message containing the result set that met the request criteria.

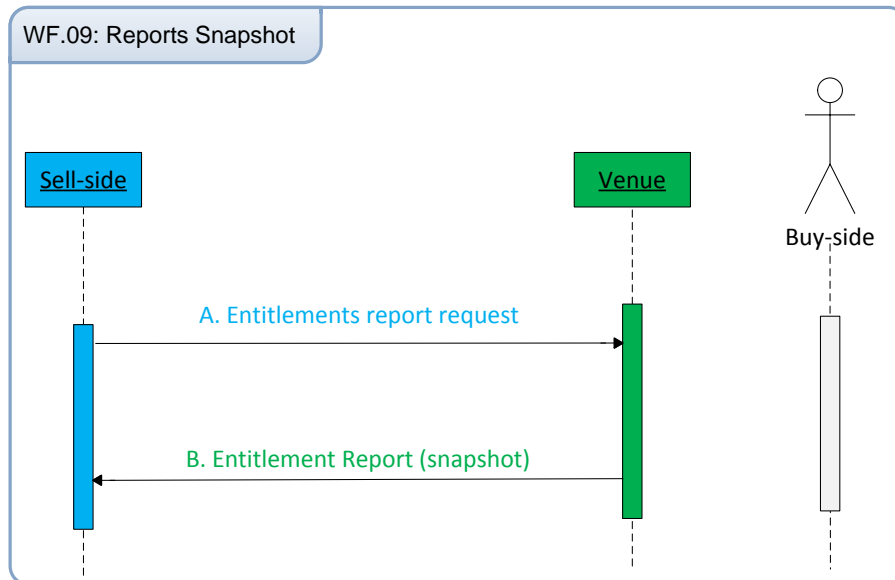


Figure 17: Scenario WF9: Reports snapshot

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0				
(A) Entitlements report request	Sell-Side	→	CU – PartyEntitlementsRequest EntitlementRequestID(1770)= ❶ SubscriptionRequestType(263)=Snapshot(0) PartyDetailStatus(1672) <optional> MarketSegmentScopeGrp <optional> NoEntitlementTypes(2345) <optional> -> EntitlementType(1775)	Execution Venue
(B) Entitlements Report (Snapshot)		←	CV – PartyEntitlementsReport EntitlementRequestID(1770)= ❶ EntitlementReportID(1771)= <required> RequestResult(1511)= Valid Request(0) NoPartyEntitlements(1772)=N -> NoPartyDetails(1671)=M -> PartyDetailGrp -> EntitlementGrp	

Table 10: Scenario WF9: Reports snapshot

Notes

- The result set may be fragmented across multiple PartyEntitlementsReport messages by setting the TotNoParties(1512) and LastFragment(893) fields
- When a sell side sends a a PartyEntitlementsRequest he may wish to add a criteria to filter the results set by:

-
- the value of PartyDetailStatus; e.g. set criteria to filter the results set to only active (or pending) parties
 - Instrument Scope (see instrument scope FIX component)
 - one or more values of EntitlementType; e.g. retrieve the entitlements to Trade and to Subscribe to market data; In such a case, the PartyEntitlementReport(35=CU) message contains NoEntitlementType(2345)=2 followed by EntitlementType(1775)=Trade(0) and EntitlementType(1775)=Subscribe to market data(5)
 - the market segment (i.e. product/sector); e.g. set criteria to filter the results set to only IRS
-
- See examples request messages in: [PartyEntitlementRequest \(35=CU\)](#)
 - See an example report message in: [PartyEntitlementReport \(35=CV\)](#)

7.11 Scenario WF10: Reports update

This scenario illustrates the case where the sell-side subscribes for the list of buy-side users and their entitlements status in the form of a party entitlement request. The Execution Venue responds by returning a PartyEntitlementReport containing the required parties, their current entitlements. The Venue follows with messages on any addition, deletion or modification of entitlements. Addition and deletion are usually a result of buy-side requests whilst the modifications which usually made by the sell-side (by sending PartyEntitlementsDefinitionRequests message) are sent back to the sell-side in PartyEntitlementsReport message (having ListUpdateAction(1324)=Modify(M)).;

It is possible to have new entitlements, modified entitlements and deleted entitlements in the same update message.

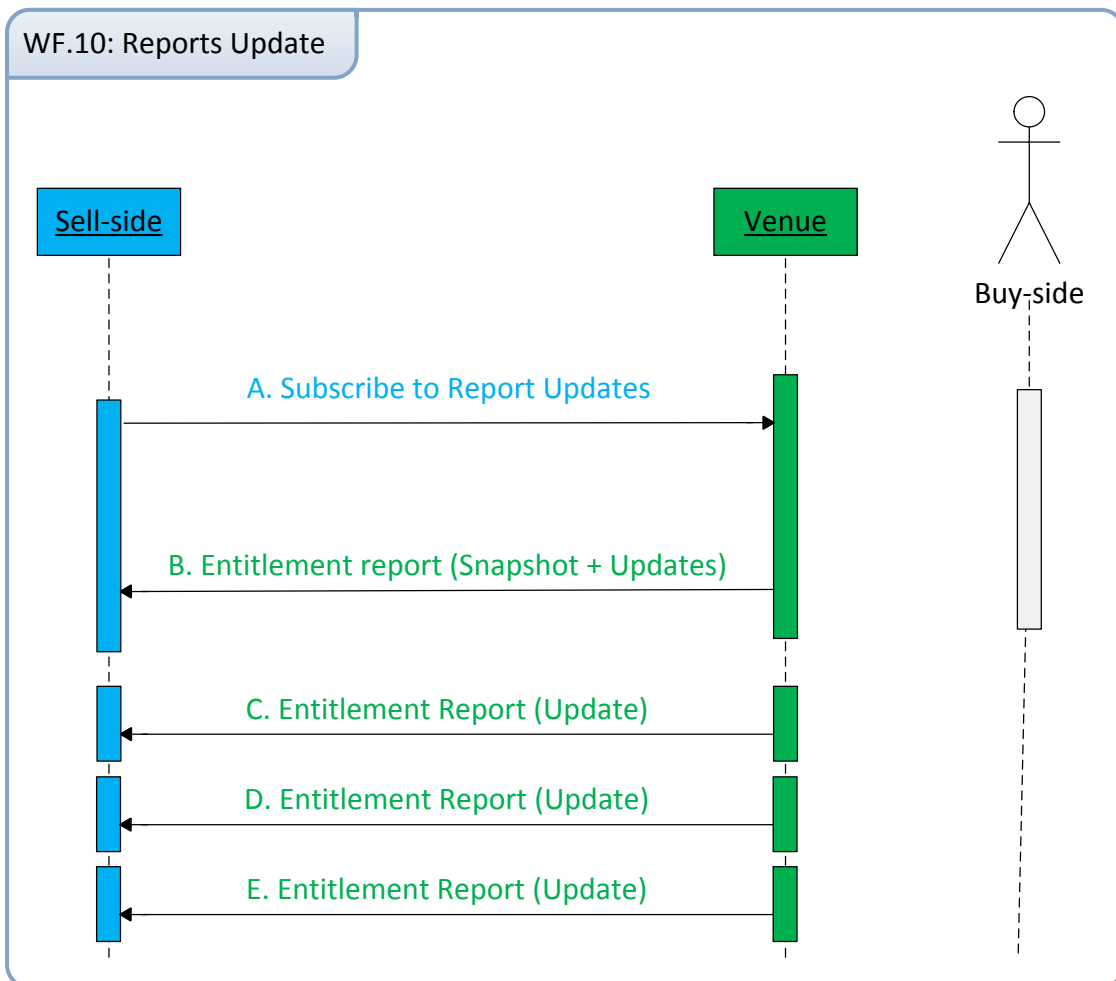


Figure 18: Scenario WF10: Reports update

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0			
(A) Entitlements report request	→	CU – PartyEntitlementsRequest EntitlementRequestID(1770)= ❶ SubscriptionRequestType(263)=Snapshot + Updates(1) PartyDetailStatus(1672) <optional> MarketSegmentScopeGrp <optional> NoEntitlementType(2345) <optional> -> EntitlementType(1775)	Execution Venue
(B) Entitlements Report (Snapshot)	←	CV – PartyEntitlementsReport EntitlementRequestID(1770)= ❶ EntitlementReportID(1771)= ❷ RequestResult(1511)= Valid Request(0) NoPartyEntitlements(1772)=N -> NoPartyDetails(1671)=M -> PartyDetailGrp -> EntitlementGrp	
(C) New Entitlement(s) (Update report)	↑ ↑ ↑	CZ – PartyEntitlementsUpdateReport EntitlementRequestID(1770)= ❶ EntitlementReportID(1771)= ❸ NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Add(A) -> PartyDetailGrp -> EntitlementGrp	
(D) Modified Entitlement(s) (Update report)	↑ ↑ ↑	CZ – PartyEntitlementsUpdateReport EntitlementRequestID(1770)= ❶ EntitlementReportID(1771)= ❹ NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp -> EntitlementGrp	
(E) Delete Entitlement(s) (Update report)	↑ ↑ ↑	CZ – PartyEntitlementsUpdateReport EntitlementRequestID(1770)= ❶ EntitlementReportID(1771)= ❺ NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Delete(D) -> PartyDetailGrp -> EntitlementGrp	

Table 11: Scenario WF10: Reports update

Notes

- See scenario [WF9 Notes](#)
- Each PartyEntitlementReport and PartyEntitlementUpdateReport message may contain more than a single entitlement
- A single PartyEntitlementUpdateReport message may contain updates to add some new entitlements, modify some entitlements and delete some entitlements
- PartyEntitlementReport and PartyEntitlementUpdateReport messages can be fragmented

7.12 Scenario WF11: Suspension of client by sell-side

This scenario illustrates the case where the Sell-side suspends buy-side client (single or multiple); Suspension may be for the firm, one or more users, one or more accounts etc. The Venue amends the entitlement definition on the platform (system) to be suspended

- For 'reject' case see [Scenario WF8](#)

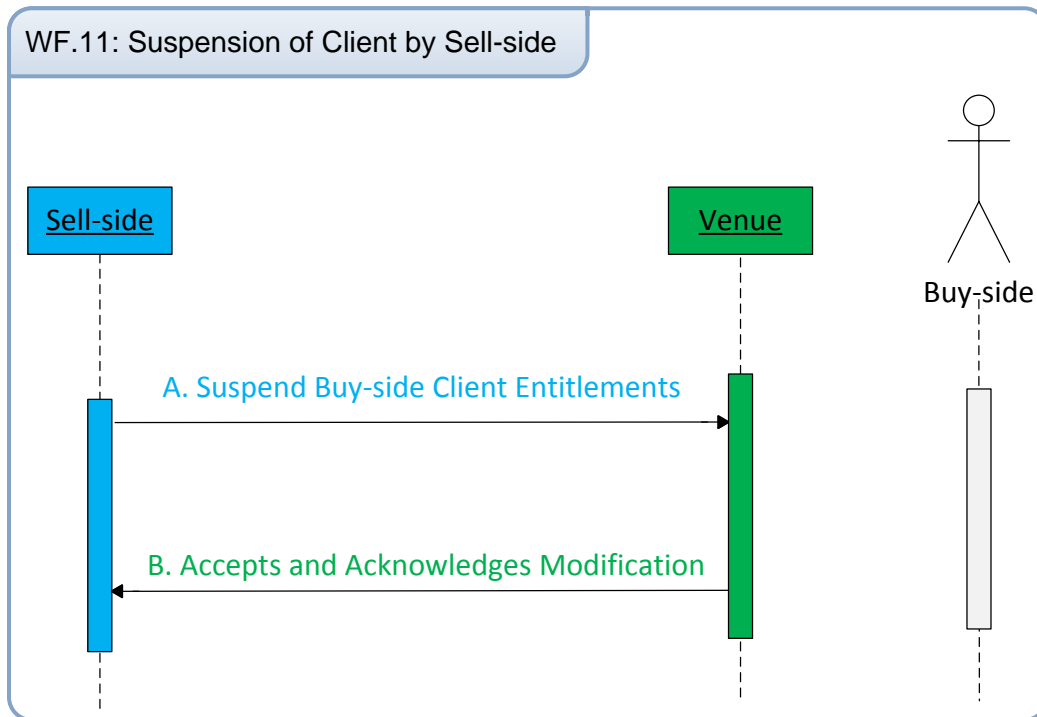


Figure 19: Scenario WF11: Suspension of client by sell-side

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0				
(A) Sell-side Suspend buy-side client	Sell-Side	→	DA – PartyEntitlementsDefinitionRequest EntitlementRequestID(1770)=❶ RequestingPartyGrp NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp ->->PartyDetailStatus(1672) = Suspended(1)	Execution Venue
(B) Execution Venue Accepts and acknowledges Modification		←	DB – PartyEntitlementsDefinitionRequestAck EntitlementRequestID(1770)=❶ EntitlementRequestStatus(1882)=Accepted(0) EntitlementRequestResult(1881)=successful(0) (default) -> PartyDetailGrp ->->PartyDetailStatus(1672) = Suspended(1)	

Table 12: Scenario WF11: Suspension of client by sell-side

7.13 Scenario WF12: Off-board a client by request

This scenario illustrates the case where the Sell-side requests Execution Venue to remove all the entitlements of a client. Venue removes all the entitlements which are associated with the client from the platform (system)

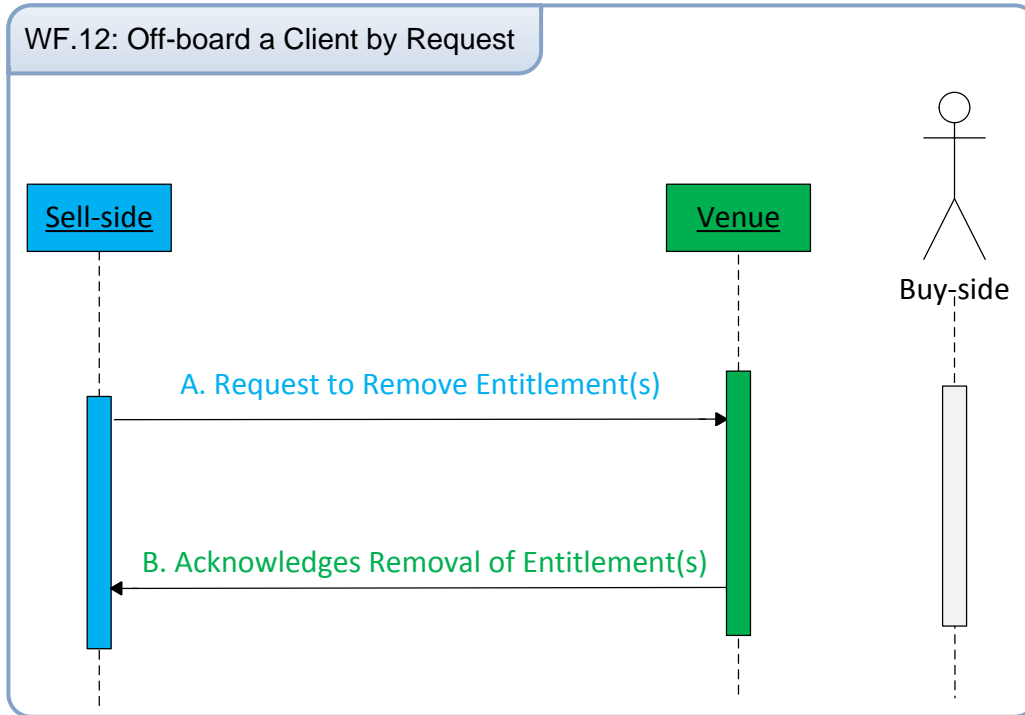


Figure 20 Scenario WF12: Off-board a client by request

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0				
(A) Sell-side disable (Delete) entitlement(s)	Sell-Side	→	DA – PartyEntitlementsDefinitionRequest EntitlementRequestID(1770)=❶ RequestingPartyGrp NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Delete(D) -> PartyDetailGrp	Execution Venue
(B) Execution Venue acknowledges and accepts		←	DB – PartyEntitlementsDefinitionRequestAck EntitlementRequestID(1770)=❶ EntitlementRequestStatus(1882)=Accepted(0) EntitlementRequestResult(1881)=successful(0) (default)	

Table 13: Scenario WF12: Off-board a client by request

7.14 Scenario WF13: Off-board or Suspend a Firm, User or Account

This scenario illustrates the case where the Venue notifies sell-side of buy-side client to be removed or suspended. A preliminary condition is that the sell-side subscribes to receive the requests or ‘actively asks’ (i.e. requests) to receive buy-side requests for entitlements.

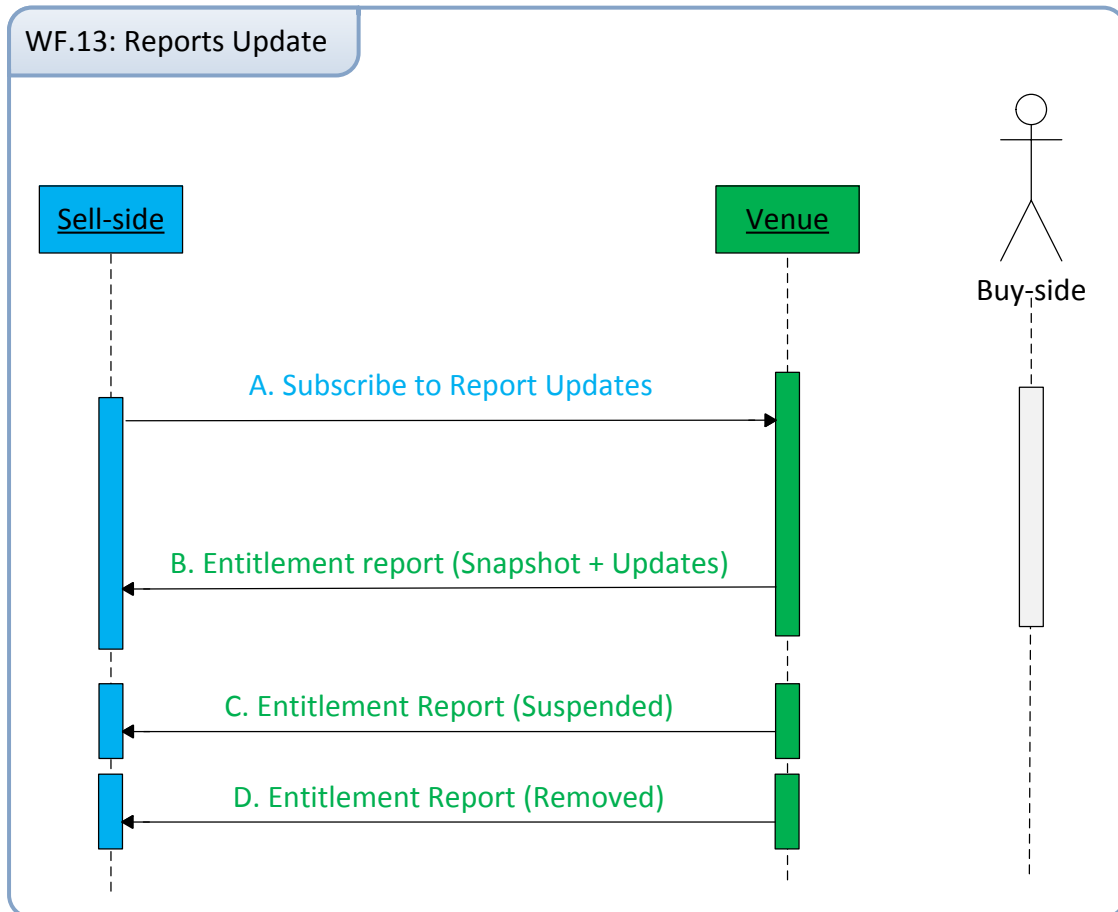


Figure 21 Scenario WF13: Off-board or Suspend a Firm, User or Account

Model Flow

The following table highlights the significant attributes which are relevant to this workflow.

Model FIX 5.0				
(A) Entitlements report request	Sell-Side	→	CU – PartyEntitlementsRequest EntitlementRequestID(1770)= ❶ SubscriptionRequestType(263)=Snapshot + Updates(1) PartyDetailStatus(1672) <optional> MarketSegmentScopeGrp <optional> NoEntitlementType(2345) <optional> -> EntitlementType(1775)	Execution Venue
(B) Entitlements Report (Snapshot)		←	CV – PartyEntitlementsReport EntitlementRequestID(1770)= ❶ EntitlementReportID(1771)= ❷ RequestResult(1511)= Valid Request(0) NoPartyEntitlements(1772)=N -> NoPartyDetails(1671)=M -> PartyDetailGrp -> EntitlementGrp	
(C) Modified Entitlement(s) (Suspended report)		← ← ←	CZ – PartyEntitlementsUpdateReport EntitlementRequestID(1770)= ❶ EntitlementReportID(1771)= ❸ NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Modify(M) -> PartyDetailGrp ->> PartyDetailStatus(1672) = Suspended(1) -> EntitlementGrp	
(D) Delete Entitlement(s) (Update report)		← ← ←	CZ – PartyEntitlementsUpdateReport EntitlementRequestID(1770)= ❶ EntitlementReportID(1771)= ❹ NoPartyEntitlements(1772)=N -> ListUpdateAction(1324)=Delete(D) -> PartyDetailGrp -> EntitlementGrp	

Table 14: Scenario WF13: Off-board or Suspend a Firm, User or Account

8 Prime Broker Workflows

8.1 Message Flows Summary

This section details the various scenarios identified as requirements to be supported in communication between execution venues and prime brokers. Each scenario is described at a high level followed by the FIX messages used to support the scenario.

For a given implementation of these scenarios, Execution Venues need to take into consideration having the proper controls in place to safeguard the information and determining whether the requester is permitted to see or retrieve the information. Further discussions on this topic are out of scope of this document.

Scenario summary by category:

Category	Workflow initiated by venue	Workflow initiated by sell-side
New buy-side client setup	<ul style="list-style-type: none"> • WF.PB.14: Prime broker: New buy-side client request 	<ul style="list-style-type: none"> • WF.PB.15: Prime broker: New buy-side client setup • WF.PB.16: Prime broker: New buy-side client setup - rejected
Relationship amendments (single or multiple)		<ul style="list-style-type: none"> • WF.PB.17: Prime broker Amendment • WF.PB.18: Prime broker active / deactivate user
Relationship removal		<ul style="list-style-type: none"> • WF.PB.19: Prime broker Off-boarding a client

Scenario summary by section:

Scenarios

8.2 Scenario WF.PB.14: Prime broker: New buy-side client request	49
8.3 Scenario WF.PB.15: Prime broker: New buy-side setup	50
8.4 Scenario WF.PB.16: Prime broker: New buy-side setup - rejected	51
8.5 Scenario WF.PB.17: Prime broker Amendment	52
8.6 Scenario WF.PB.18: Prime broker activate / deactivate user.....	53
8.7 Scenario WF.PB.19: Prime broker Off-boarding a client	54

8.2 Scenario WF.PB.14: Prime broker: New buy-side client request

This scenario illustrates the case where the Execution Venue sends a request to the prime broker on behalf of a buy-side user for entitlement which is subsequently approved.

- Such a request may contain:
 - Buy-side client information including venue identifiers for the buy-side client
 - Type of entitlement requested (e.g. Trade)
- Prime broker agrees and sends entitlement definition
 - Agree to buy-side client to trade
 - The type of entitlement and restrictions (e.g. credit limit, currency pairs)
- Venue sets the entitlement definition on the platform (system) and acknowledges the Prime broker
 - Accordingly notifies the buy-side client they can start trading

This scenario's workflow is identical to [WF1](#) where the sell-side acts as a prime broker.
See: [WF1](#) for further details

8.3 Scenario WF.PB.15: Prime broker: New buy-side setup

This scenario illustrates the case where the prime broker initiates entitlement definition request to venue, Venue sets the entitlement definition on the platform (system) and acknowledges the Prime broker

- Such a request may contain:
 - Agree to buy-side client to trade
 - The type of entitlement and restrictions (e.g. credit limit, currency pairs)
 - Define buy-side client and user identifier

This scenario's workflow is identical to [WF3](#) where the sell-side acts as a prime broker.

See: [WF3](#) for further details

8.4 Scenario WF.PB.16: Prime broker: New buy-side setup - rejected

This scenario illustrates the case where the Prime broker initiates entitlement definition request to venue, which is rejected

- Such a request may contain:
 - Agree to buy-side client to trade
 - The type of entitlement and restrictions (e.g. credit limit, currency pairs)
 - Define buy-side client and user identifier

This scenario's workflow is identical to [WF8](#) where the sell-side acts as a prime broker.
See: [WF8](#) for further details

8.5 Scenario WF.PB.17: Prime broker Amendment

This scenario illustrates the case where the prime broker modifies existing entitlement definition (single or multiple)

- Such a request may contain:
 - Type of entitlement requested (e.g. Trade)
 - Instrument scope (e.g. currency pairs)
 - Attributes (e.g. credit limit)

Venue amends the entitlement definition on the platform (system)

This scenario's workflow is identical to [WF6](#) where the sell-side acts as a prime broker.
See: [WF6](#) for further details

8.6 Scenario WF.PB.18: Prime broker activate / deactivate user

This scenario illustrates the case where the prime broker modifies existing entitlement definition i.e. user trading status

- Such a request may contain:
 - Activate or suspend individual user within a buy-side client

Venue amends the entitlement definition on the platform (system)

This scenario's workflow is identical to [WF11](#) where the sell-side acts as a prime broker. See [WF11](#) for further details.

8.7 Scenario WF.PB.19: Prime broker Off-boarding a client

This scenario illustrates the case where the prime broker disables buy-side client and the Venue removes the entitlement from the platform (system)

This scenario's workflow is identical to [WF12](#) where the sell-side acts as a prime broker.
See [WF12](#) for further details.

9 Stream Assignment Workflows

9.1 Message Flows Summary

This section details the various scenarios identified as requirements to be supported in communication pricing streams assignments between execution venues and sell-side liquidity providers. Each scenario is described at a high level followed by the FIX messages used to support the scenario.

Sell-side may assign the buy-side clients to certain pricing streams that the sell-side publishes via the Execution Venue. In the FX markets, clients are often assigned to different pricing streams based on the volume bands and currency pairs without changing their other entitlements.

Scenarios

9.2 Scenario SA.1: Buy-side client requests to be assigned to a Pricing Stream – sell-side assigns	56
9.3 Scenario SA.2: Buy-side client requests to be assigned to a Pricing Stream – sell-side rejects.....	58
9.4 Scenario SA.3: Sell-side assigns a buy-side client to a Pricing Stream	60
9.5 Scenario SA.4: Sell-side terminates (un-assigns) a Pricing Stream from the buy-side client	61

9.2 Scenario SA.1: Buy-side client requests to be assigned to a Pricing Stream – sell-side assigns

This scenario illustrates the case where a buy-side client requests via the execution venue for one or more buy-side users to be assigned to a pricing stream. The sell-side assigns. The venue sets the new pricing stream assignment and acknowledges the sell-side.

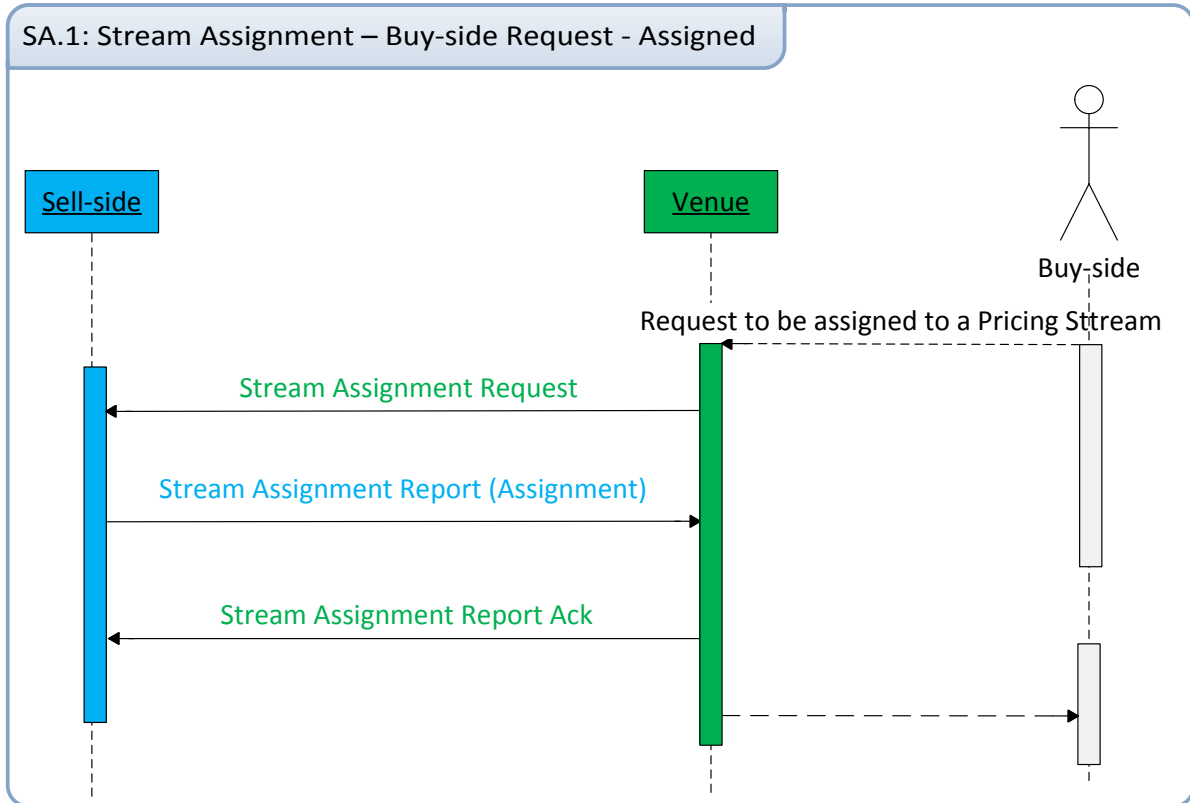


Figure 22: Scenario SA.1: Buy-side client requests to be assigned to a Pricing Stream – sell-side assigns

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0				
(A) Buy-side requests to assign one or more users to a pricing stream	Sell-Side	←	CC – StreamAssignmentRequest StreamAsgnReqID(1497)=❶ StreamAsgnReqType(1498)=<required> NoAsgnReqs(1499)=<required> -> Parties <required> -> NoRelatedSym (146)=<required> ->> Symbol(55)=<required> // format: "CCY1/CCY2" ->> MDentrySize(271) <optional> ->> MDStreamID(1500) // market data stream identifier	Execution Venue
(B) sell-side assigns one or more users to a pricing stream		→	CD – StreamAssignmentReport StreamAsgnRptID (1501)=❷ StreamAsgnReqType(1498)=<required> (Must be the same value as received in the StreamAssignmentRequest) StreamAsgnReqID(1497)=❶ NoAsgnReqs(1499)=<required> -> Parties <required> -> NoRelatedSym (146)=<required> ->> Symbol(55)=<required> // format: "CCY1/CCY2" ->> StreamAsgnType(1617)=Assignment(1) ->> MDStreamID(1500) < pricing stream identifier >	
(C) Acknowledgement		←	CE – StreamAssignmentReportAck StreamAsgnAckType(1503) = Assignment Accepted(0) StreamAsgnRptID (1501)=❷	

Table 15: Scenario SA.1: Buy-side client requests to be assigned to a Pricing Stream – sell-side assigns

Notes

- StreamAsgnReqType(1498) is used to distinct between an existing client (StreamAsgnReqType(1498)=2) to a new client (StreamAsgnReqType(1498)=1).
- Parties component may contain one or more users to be assigned to the price stream

9.3 Scenario SA.2: Buy-side client requests to be assigned to a Pricing Stream – sell-side rejects

This scenario illustrates the case where a buy-side client requests for one or more users to be assigned to a pricing stream. Subsequently, sell-side rejects the request.

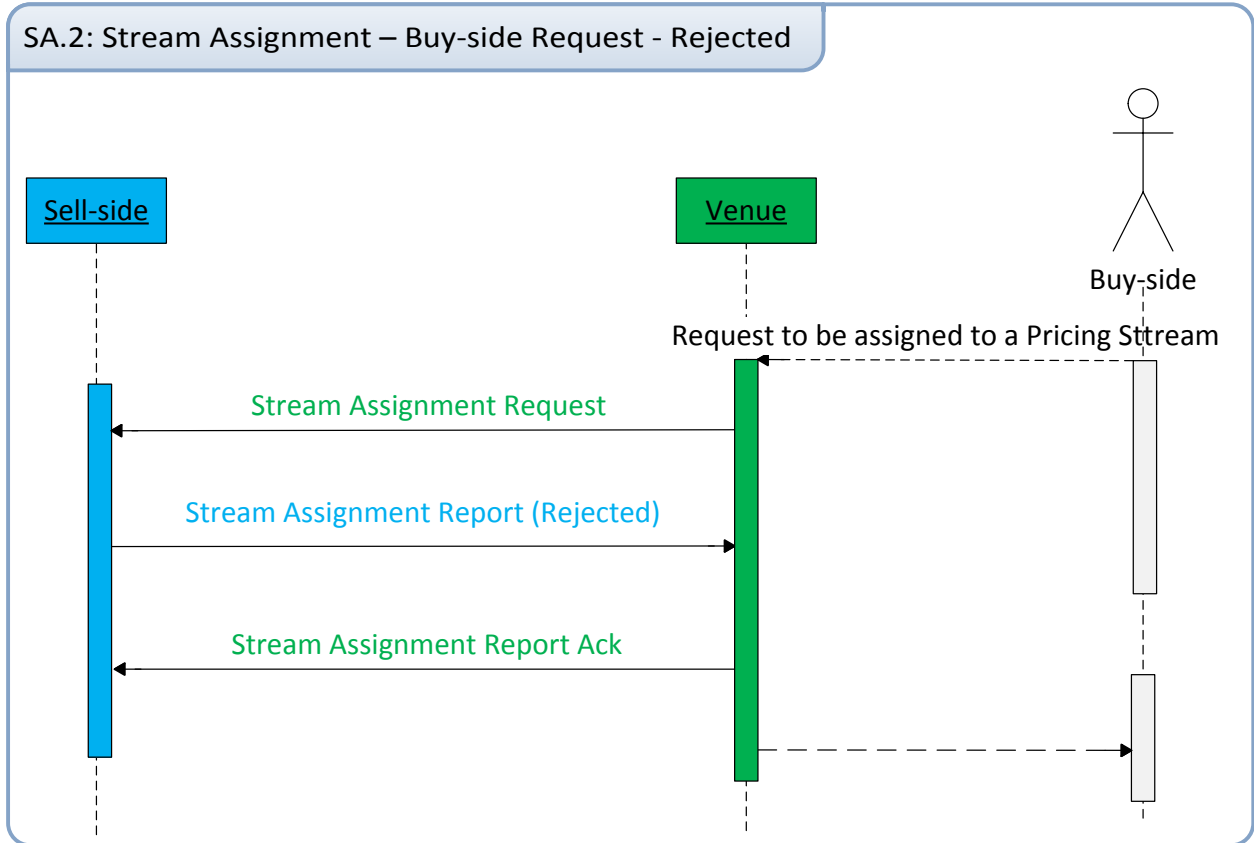


Figure 23: Scenario SA.2: Buy-side client requests to be assigned to a Pricing Stream – sell-side rejects

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0				
(A) Buy-side requests to assign one or more users to a pricing stream	Sell-Side	←	CC – StreamAssignmentRequest StreamAsgnReqID(1497)=❶ StreamAsgnReqType(1498)=<required> NoAsgnReqs(1499)=<required> -> Parties <required> -> NoRelatedSym (146)=<required> ->> Symbol(55)=<required> // format: "CCY1/CCY2" ->> MDentrySize(271) <optional> ->> MDStreamID(1500) // market data stream identifier	Execution Venue
(B) sell-side rejects the assignment request		→	CD – StreamAssignmentReport StreamAsgnRptID (1501)=❷ StreamAsgnReqType(1498)=<required> (Must be the same value as received in the StreamAssignmentRequest) StreamAsgnReqID(1497)=❶ NoAsgnReqs(1499)=<required> -> Parties <required> -> NoRelatedSym (146)=<required> ->> Symbol(55)=<required> // format: "CCY1/CCY2" ->> StreamAsgnType(1617)=Rejected(2) ->> MDStreamID(1500) // market data stream identifier ->> StreamAsgnRejReason(1502) = <Required>	
(C) Acknowledgement		←	CE – StreamAssignmentReportAck StreamAsgnAckType(1503) = Assignment Accepted(0) StreamAsgnRptID (1501)=❷	

Table 16: Scenario SA.2: Buy-side client requests to be assigned to a Pricing Stream – sell-side rejects

9.4 Scenario SA.3: Sell-side assigns a buy-side client to a Pricing Stream

This scenario illustrates the case where a sell-side assigns one or more buy-side clients to a pricing stream in a unilateral manner (i.e. the request may have occurred out-of-band). Execution venue sets the assignments and acknowledges the sell-side.

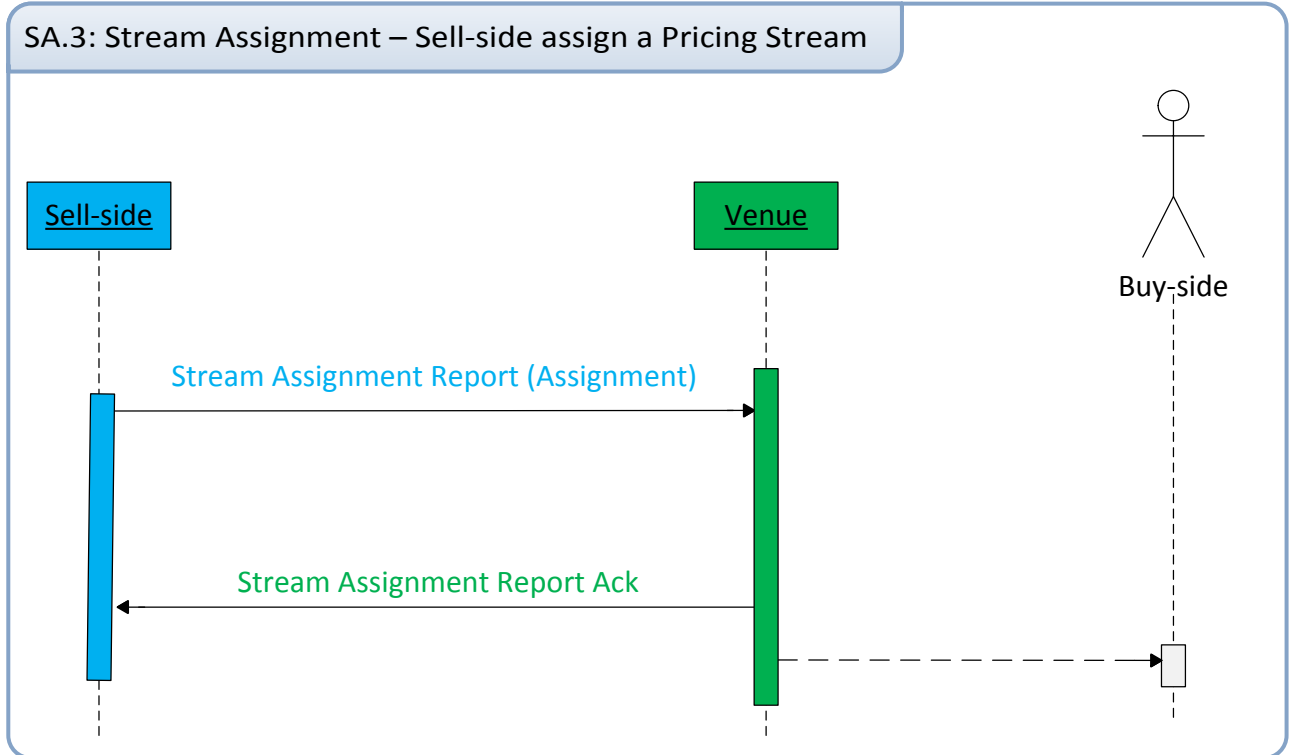


Figure 24: Scenario SA.3: Sell-side assigns a buy-side client to a Pricing Stream

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0				
(A) Sell-side assigns one or more users to a pricing stream	Sell-Side	→	CD – StreamAssignmentReport StreamAsgnRptID (1501)= ❶ NoAsgnReqs(1499)=<required> -> Parties <required> -> NoRelatedSym (146)=<required> ->> Symbol(55)=<required> // format: "CCY1/CCY2" ->> StreamAsgnType(1617)=Assignment(1) ->> MDStreamID(1500) // market data stream identifier	Execution Venue
(B) Acknowledgement	Sell-Side	←	CE – StreamAssignmentReportAck StreamAsgnAckType(1503) = Assignment Accepted(0) StreamAsgnRptID (1501)= ❷	Execution Venue

Table 17: Scenario SA.2: Sell-side assigns a buy-side client to a Pricing Stream

9.5 Scenario SA.4: Sell-side terminates (un-assigns) a Pricing Stream from the buy-side client

This scenario illustrates the case where a sell-side terminates (or unassign) a pricing stream from one or more buy-side clients.

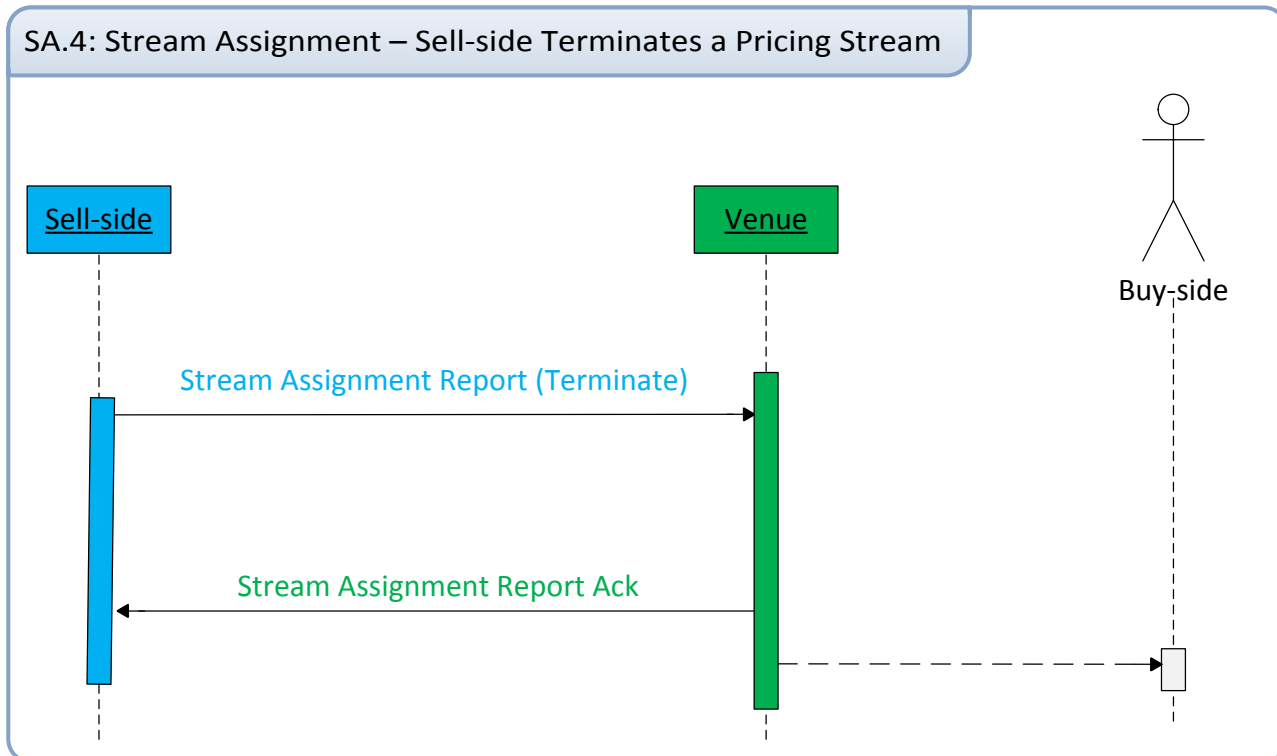


Figure 25: Scenario SA.4: Sell-side terminates (unassigns) a Pricing Stream from the buy-side client

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0				
(A) Sell-side Terminates a Pricing Stream to a Buy-side Client	Sell-Side	→	CD – StreamAssignmentReport StreamAsgnRptID (1501)= ❶ -> Parties <required> -> NoRelatedSym (146)=<required> ->> Symbol(55)=<required> // format: "CCY1/CCY2" ->> StreamAsgnType(1617)=Terminate / Unassign(3) ->> MDStreamID(1500) // market data stream identifier	Execution Venue
(B) Acknowledgement		←	CE – StreamAssignmentReportAck StreamAsgnAckType(1503) = Assignment Accepted(0) StreamAsgnRptID (1501)= ❷	

Table 18: Scenario SA.4: Sell-side terminates (unassigns) a Pricing Stream from the buy-side client

10 Miscellaneous Workflows

10.1 Message Flows Summary

This section details the various scenarios identified as requirements to be supported in communication between execution venues and sell-side. Each scenario is described at a high level followed by the FIX messages used to support the scenario.

For a given implementation of these scenarios, Execution Venues need to take into consideration having the proper controls in place to safeguard the information and determining whether the requester is permitted to see or retrieve the information. Further discussions on this topic are out of scope of this document.

Scenarios

10.2 Scenario WF.M.20: Sell-Side User Login	63
10.3 Scenario WF.M.21: Sell-Side User Logoff.....	64
10.4 Scenario WF.M.22: Sell-Side Requests Instrument types and sub-types	65
10.5 Scenario WF.M.23: Sell-Side Requests a Snapshot of Buy-Side Entities/Users	66
10.6 Scenario WF.M.24: Sell-Side Requests Snapshot + Updates of Buy-Side Entities/Users.....	68
10.7 Scenario WF.M.25: Sell side sends a message to the buy-side.....	70

10.2 Scenario WF.M.20: Sell-Side User Login

This scenario illustrates the case where an authorized sell-side user sends a login message to authenticate with the Execution Venue system and gains permission for further entitlement activities.

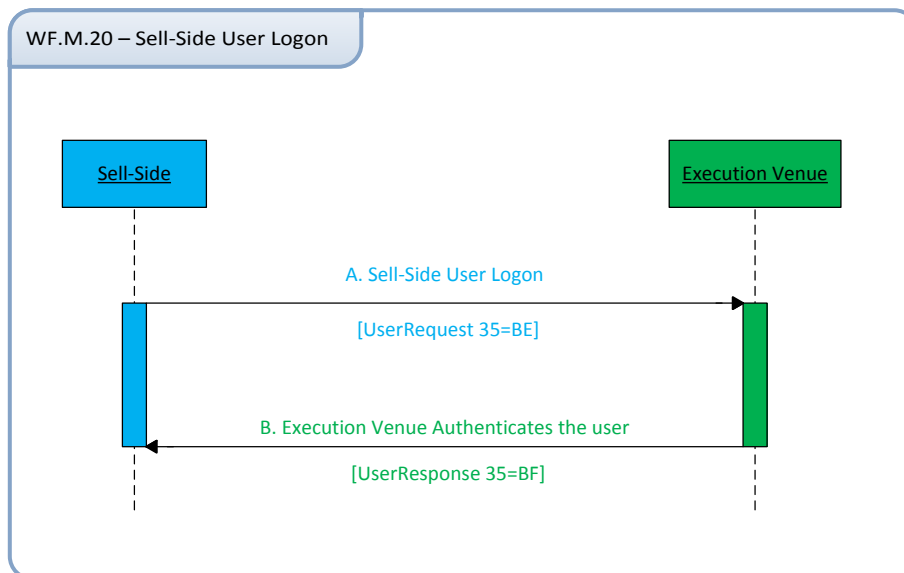


Figure 26: Scenario WF.M.20: A Sell-Side User Login

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0				
(A) Sell-Side User Logon	Sell-Side	→	BE – UserRequest UserReqID(923)=❶ UserReqType(924)=Log On User(1) Username(553)=<required> Password(554)=<optional>	Execution Venue
(B) Execution Venue Authenticates the user		←	BF – UserResponse UserReqID(923)=❶ Username(553)=<required> UserStatus(926)=Logged In(1)	

Table 19: Scenario WF.M.20: A Sell-Side User Login

Notes

- In subsequent messages the Username will be contained in the field: RequestingPartyID(1658), for example in:
 - PartyDetailsListRequest(35=CF)
 - PartyEntitlementsDefinitionRequest(35=DA)
- Workflows to change user password and request individual user status are implemented using the same messages: UserRequest(BE) followed by UserResponse(BF).

10.3 Scenario WF.M.21: Sell-Side User Logoff

This scenario illustrates the case where a sell-side user sends a logoff message to the Execution Venue system in order to terminate the session.

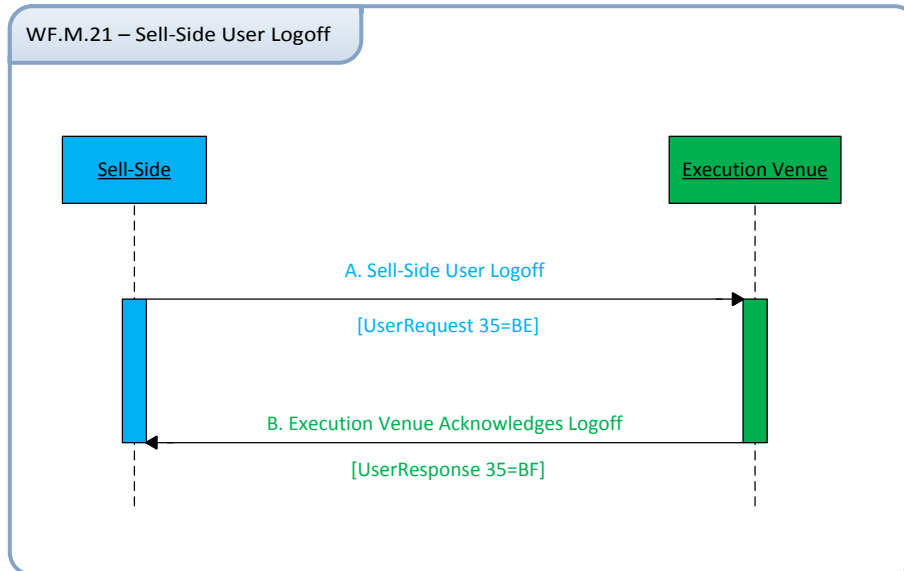


Figure 27: Scenario WF.M.21: Sell-Side User Logoff

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0				
(A) Sell-Side User Logoff	Sell-Side	→	BE – UserRequest UserReqID(923)=❶ UserReqType(924)=Log Off User(2) Username(553)=<required> Password(554)=<optional>	Execution Venue
(B) Execution Venue Acknowledges Logoff		←	BF – UserResponse UserReqID(923)=❶ Username(553)=<required> UserStatus(926)=Not Logged In(2)	

Table 20: Scenario WF.M.21: Sell-Side User Logoff

Notes

- During Logoff, the UserName is the same value of a user who has previously logged-in.
- After a user has logged-off, subsequent request messages PartyDetailsListRequest(35=CF) and PartyEntitlementsDefinitionRequest(35=DA), having the same UserName in field RequestingPartyID(1658) should be rejected by the Execution Venue.

10.4 Scenario WF.M.22: Sell-Side Requests Instrument types and sub-types

This scenario illustrates the case where a sell-side requests market segmentation information from the Execution Venue. The sell-side requests for the list of instrument types and sub-types by sending a request message. The Execution Venue replies with a list of instrument-types and sub-types. The request may be filtered to receive sub-types of a single Instrument type.

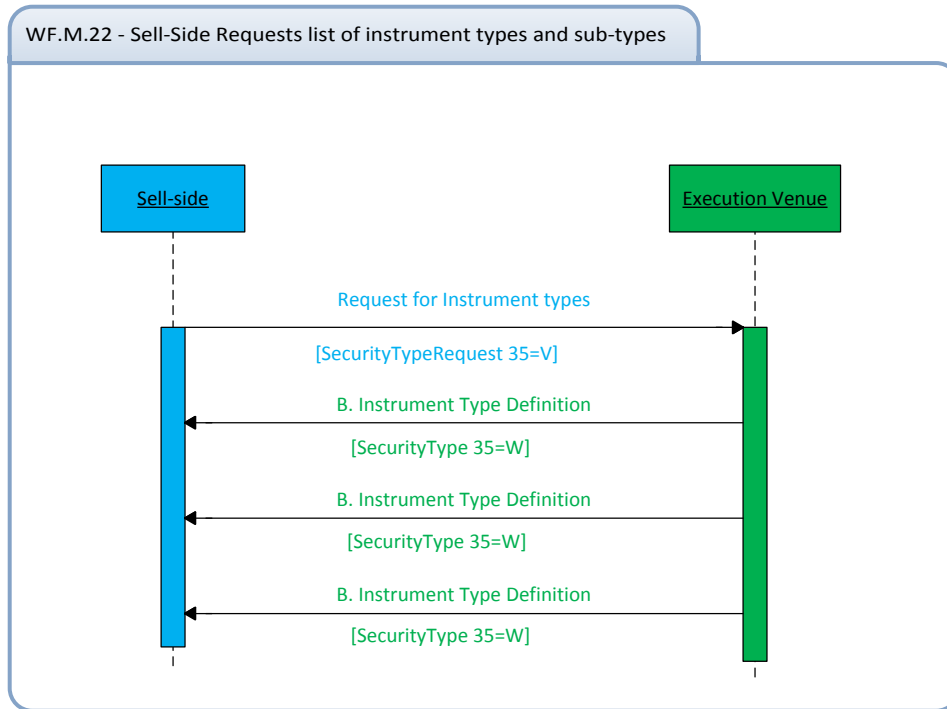


Figure 28: Scenario WF.M.22: Sell-Side Requests Instrument types and sub-types

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0				
(A) Sell-Side Requests for Instrument Type		→	v – SecurityTypeRequest SecurityReqID(320)=❶	Execution Venue
(B) Instrument Type List	Dealer	←	w – SecurityTypes SecurityReqID(320)=❶ SecurityResponseID(320)= <required> SecurityResponseType(323)=List Of Securities Returned per request(4) Tot NoSecurityTypes(557)=N NoSecurityTypes(558)=N -> SecurityType(167) -> Product(460) -> SecuritySubType(762)	

Table 21: Scenario WF.M.22: Sell-Side Requests Instrument types and sub-types

10.5 Scenario WF.M.23: Sell-Side Requests a Snapshot of Buy-Side Entities/Users

This scenario illustrates the case where the sell-side sends a request for buy-side user information to the Execution Venue. A request message may contain filtered attributes to retrieve data for a single buy-side company, a single branch within a buy-side company or a single instrument type. The Execution Venue replies by sending a PartyDetailListReport message containing the details of the requested buy-side users. The requesting message returned by the Execution Venue may be fragmented across multiple messages if the result set is large.

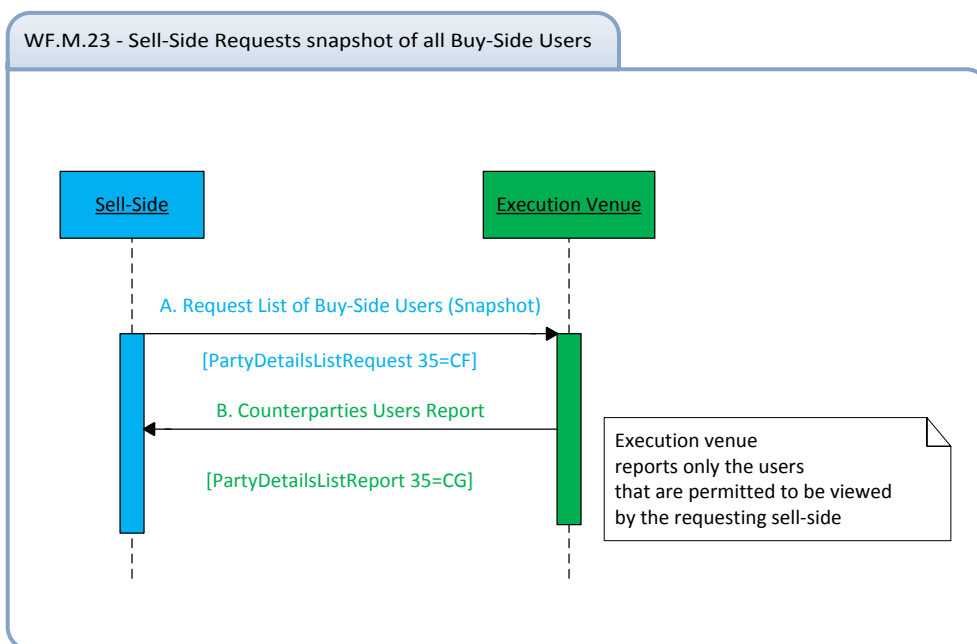


Figure 29: Scenario WF.M.23: Sell-Side Requests a Snapshot of Buy-Side Entities/Users

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0				
(A) Sell-Side Requests for Buy-side Users (Snapshot)	Sell-Side	→	CF – PartyDetailsListRequest PartyDetailsListRequestID(1505) = ① SubscriptionRequestType(263)=Snapshot(0)	Execution Venue
(B) Execution Venue sends results set		←	CG – PartyDetailsListReport PartyDetailsListRequestID(1505) = ① PartyDetailsListReportID (1510)= <required> RequestResult(1511)=Valid Request(0) NoPartyDetails(1671)=N (number of users in this message) -> PartyDetailID(1691) = <required> -> PartyDetailIDSource(1692) -> PartyDetailRole(1693) -> PartyDetailSubGrp	

Table 22: Scenario WF.M.23: Sell-Side Requests a Snapshot of Buy-Side Entities/Users

Notes

- The following enumerations were recently added to FIX protocol:
 - PartyDetailRole(1693) = Sales Person(117)
 - PartyDetailRole(1693) = Sales Operator(118)
- The result set may be fragmented across multiple PartyDetailsListReport messages by setting the TotNoParties(1512) and LastFragment(893) fields.

-
- Each buy-side user should have a PartyDetailGrp containing all the attributes that are associated with this user
 - It is currently impossible to filter the request by products or sectors (i.e. request all users that are associated with a single security type)
 - Requests may include attributes to filter the response by The requested party role (e.g. a request may be sent for all users of a single buy-side firm or a single branch within the buy-side firm)

10.6 Scenario WF.M.24: Sell-Side Requests Snapshot + Updates of Buy-Side Entities/Users

This scenario illustrates the case where the sell-side sends a request for buy-side user information with updates to the Execution Venue. A request message is sent, which may contain data filtering attributes to retrieve data for all users or a single buy-side company or a single branch within a buy-side company or a single product. The Execution Venue responds with a report message containing the details of the requested buy-side users. The result set returned by the Execution Venue may be fragmented across multiple messages. The Execution Venue continues to send updates when:

- A new buy-side user is added to their system
- There is a change to an existing buy-side user
- A buy-side user is deleted or removed from their system (i.e. “trader leaves”)

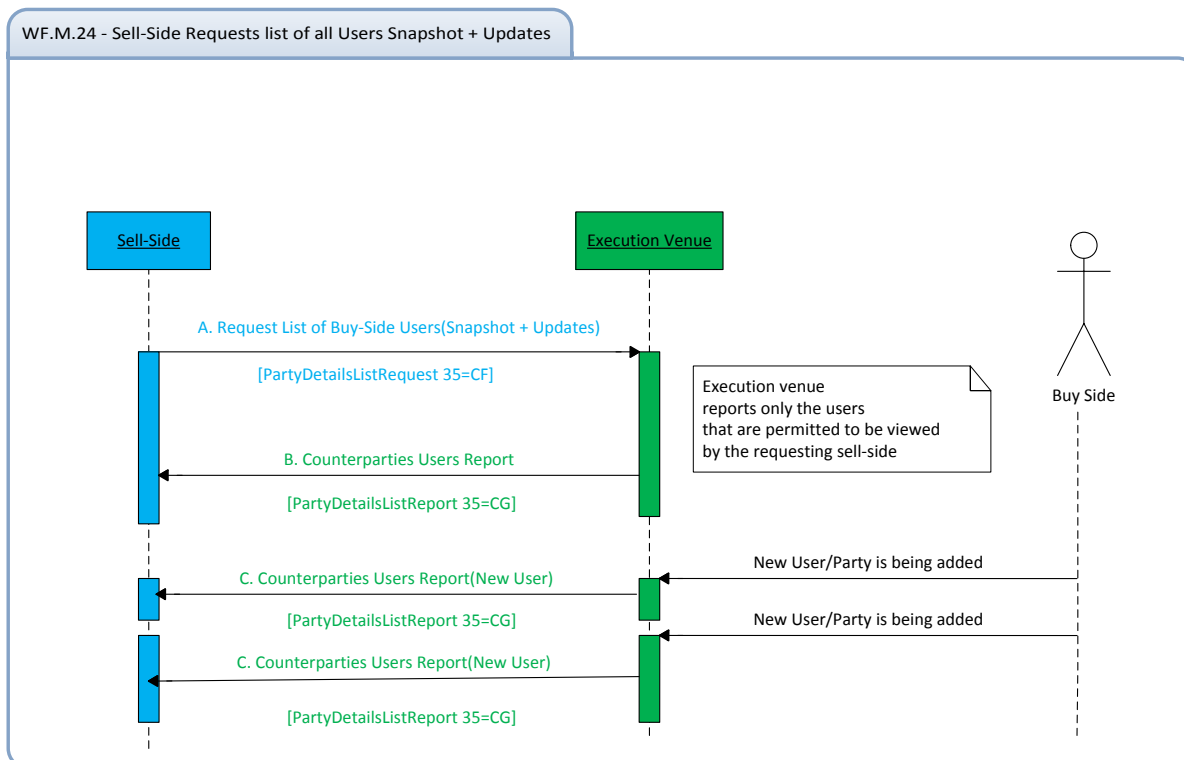


Figure 30: Scenario WF.M.24: Sell-Side Requests Snapshot + Updates of Buy-Side Entities/Users

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0				
(A) Sell-Side Requests for Buy-side Users	Sell-Side	→	CF – PartyDetailsListRequest PartyDetailsListRequestID(1505) = ❶ SubscriptionRequestType(263)=Snapshot+Updates(1)	Execution Venue
(B) Counterparties User Report (Snapshot)		←	CG – PartyDetailsListReport PartyDetailsListRequestID(1505) = ❶ PartyDetailsListReportID (1510)=❷ RequestResult(1511)=Valid Request(0) NoPartyDetails(1671)=N (number of users in this message) -> PartyDetailID(1691) = ❸ -> PartyDetailIDSource(1692) -> PartyDetailRole(1693) -> PartyDetailSubGrp	
(C) Counterparties New User Report		←	CK – PartyDetailsListUpdateReport PartyDetailsListRequestID(1505) = ❶ PartyDetailsListReportID (1510)=❷ NoPartyUpdates (1676)=N (number of changes) -> ListUpdateAction = Add/Delete/Modify -> NoPartyDetails(1671)=N (number of users in this message) -> -> PartyDetailID(1691) = ❹ -> -> PartyDetailIDSource(1692) -> -> PartyDetailRole(1693) -> -> PartyDetailSubGrp	

Table 23: Scenario WF.M.24: Sell-Side Requests Snapshot + Updates of Buy-Side Entities/Users

10.7 Scenario WF.M.25: Sell side sends a message to the buy-side

This scenario illustrates the case where the sell-side sends a message to the buy-side. This is sometimes important in order to send the sell-side disclaimer to the buy-side.

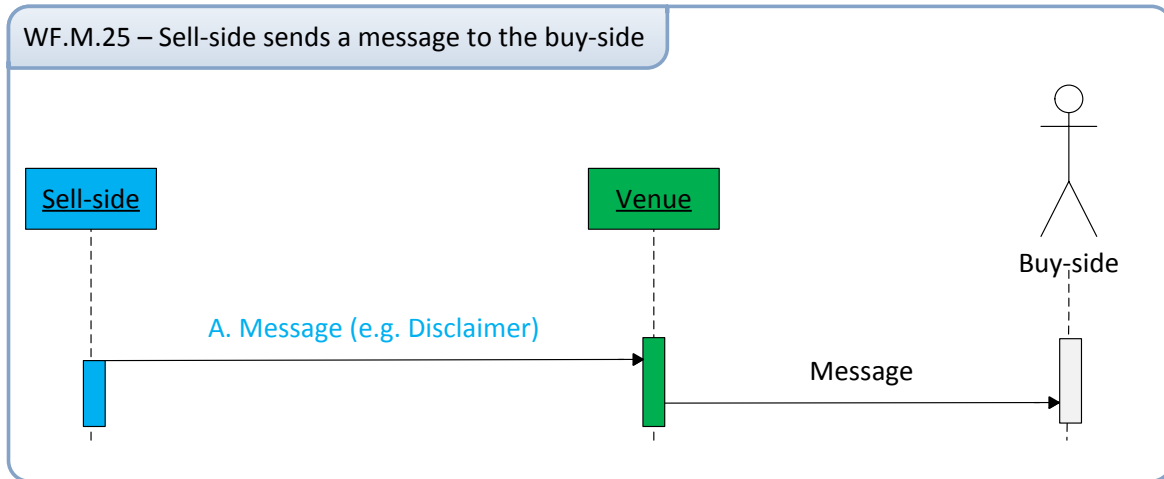


Figure 31: Scenario WF.M.25: Sell side sends a message to the buy-side

Model Flow

The following table illustrates the flows expected when communicating with an Execution Venue implementing FIX 5.0 SP2.

Model FIX 5.0		
(A) Sell-Side send a message	Sell-Side →	<p>C – Email</p> <p>EmailThreadID(164) = 1</p> <p>EmailType(94)=New(0)</p> <p>Subject(147) <required></p> <p>NoRoutingIDs = N</p> <p>-> RoutingType(216)=Target Firm(1)</p> <p>-> RoutingID <required> i.e. destination</p> <p>NoLinesOfText=M</p> <p>->Text <required></p> <p>-> AttachmentGrp <Required for attachments e.g. disclaimer></p>
		Execution Venue

Table 24: Scenario WF.M.25: Sell side sends a message to the buy-side

Note: Non-text portion, or attachments (i.e. PDF, word processor documents or bitmaps) may be added to the message using component: AttachmentGrp

11 Message Detail

This section describes in detail all FIX application messages used in this volume. A summary of all the messages described in this volume is provided below.

11.1 UserRequest (35=BE)

UserRequest (35=BE)			Dealer -> Execution Venue	
<i>This message is used to initiate a user action, logon, logout or password change. It can also be used to request a report on a user's status.</i>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType = "BE"	
923	UserRequestID	Y	Unique identifier for a User Request.	
924	UserRequestType	Y	Indicates the action required by a User Request Message	
553	Username	Y	Userid or username.	
554	Password	N	Password or passphrase.	
925	NewPassword	N	New Password or passphrase	
1400	EncryptedPasswordMethod	N	Enumeration defining the encryption method used to encrypt password fields. At this time there are no encryption methods defined by FPL.	
1401	EncryptedPasswordLen	N	Length of the EncryptedPassword(1402) field	
1402	EncryptedPassword	N	Encrypted password - encrypted via the method specified in the field EncryptedPasswordMethod(1400)	
1403	EncryptedNewPasswordLen	N	Length of the EncryptedNewPassword(1404) field	
1404	EncryptedNewPassword	N	Encrypted new password - encrypted via the method specified in the field EncryptedPasswordMethod(1400)	
95	RawDataLength	N	Number of bytes in raw data field.	
96	RawData	N	Can be used to hand structures etc to other API's etc	
	StandardTrailer	Y	The standard FIX message trailer	

11.2 UserResponse (35=BF)

UserResponse (35=BF)			Execution Venue -> Dealer	
<i>This message is used to respond to a user request message, it reports the status of the user after the completion of any action requested in the user request message.</i>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType = "BF"	
923	UserRequestID	Y	Unique identifier for a User Request.	
553	Username	Y	Userid or username.	
926	UserStatus	N	Indicates the status of a user	
	ThrottleParamsGrp	N		
927	UserStatusText	N	Reason a request was not carried out	
	StandardTrailer	Y	The standard FIX message trailer	

11.3 Email (35=C)

Email (35=C)

The email message is similar to the format and purpose of the News message, however, it is intended for private use between two parties.

Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType = C	
164	EmailThreadID	Y	Unique identifier for the email message thread	
94	EmailType	Y	Email message type.	
42	OrigTime	N	Time of message origination (always expressed in UTC (Universal Time Coordinated, also known as "GMT"))	
147	Subject	Y	Specifies the Subject text	
356	EncodedSubjectLen	N	Must be set if EncodedSubject field is specified and must immediately precede it.	
357	EncodedSubject	N	Encoded (non-ASCII characters) representation of the Subject field in the encoded format specified via the MessageEncoding field.	
	RoutingGrp	N	Required if any RoutingType and RoutingIDs are specified. Indicates the number within repeating group.	
215	NoRoutingIDs	N	Required if any RoutingType and RoutingIDs are specified. Indicates the number within repeating group.	
-> 216	RoutingType	N	Indicates type of RoutingID. Required if NoRoutingIDs is > 0.	
-> 217	RoutingID	N	Identifies routing destination. Required if NoRoutingIDs is > 0.	
	InstrmtGrp	N	Specifies the number of repeating symbols (instruments) specified	
146	NoRelatedSym	N	Specifies the number of repeating symbols (instruments) specified	
->	Instrument	N	Insert here the set of "Instrument" (symbology) fields defined in "Common Components of Application Messages"	
->-> 55	Symbol	N	Common, "human understood" representation of the security. SecurityID value can be specified if no symbol exists (e.g. non-exchange traded Collective Investment Vehicles) Use "[N/A]" for products which do not have a symbol.	
...				
	UndInstrmtGrp	N	Number of underlyings	
...				
	InstrmtLegGrp	N	Number of legs Identifies a Multi-leg Execution if present and non-zero.	
...				
37	OrderID	N	Unique identifier for Order as assigned by sell-side (broker, exchange, ECN). Uniqueness must be guaranteed within a single trading day. Firms which accept multi-day orders should consider embedding a date within the OrderID field to assure uniqueness across days.	

11	ClOrdID	N	Unique identifier for Order as assigned by the buy-side (institution, broker, intermediary etc.) (identified by SenderCompID (49) or OnBehalfOfCompID (5) as appropriate). Uniqueness must be guaranteed within a single trading day. Firms, particularly those which electronically submit multi-day orders, trade globally or throughout market close periods, should ensure uniqueness across days, for example by embedding a date within the ClOrdID field.	
	LinesOfTextGrp	Y	Specifies the number of repeating lines of text specified	
33	NoLinesOfText	Y	Specifies the number of repeating lines of text specified	
-> 58	Text	Y	Repeating field, number of instances defined in LinesOfText	
-> 354	EncodedTextLen	N	Must be set if EncodedText field is specified and must immediately precede it.	
-> 355	EncodedText	N	Encoded (non-ASCII characters) representation of the Text field in the encoded format specified via the MessageEncoding field.	
95	RawDataLength	N	Number of bytes in raw data field.	
	AttachmentGrp	N	Required if any attachments files.	
2104	NoAttachments	N	Indicates the number of attached files	
-> 2105	AttachmentName	N	Specifies the file name of the attachment	
-> 2106	AttachmentMediaType	N	The MIME media type (and optional subtype) of the attachment. The values used are those assigned, listed and maintained by IANA (www.iana.org) [RFC2046]. See http://www.iana.org/assignments/media-types/index.html for available types. Examples values (RFC number provided for reference here only): "application/pdf" (see [RFC3778]) "application/msword" (for .doc files) "multipart/signed" (see [RFC1847]) "application/vnd.openxmlformats-officedocument.wordprocessingml.document" (for .docx files)	
-> 2107	AttachmentClassification	N	Specifies semantically the type of the attached document from a business perspective. The default classification scheme reuses the FIX standard classification scheme of a high level section (pretrade, trade, posttrade, etc.) and a category, then a specific application or document type. The expression follows {"section/category/application type"}. The goal here is to map the attachment into the sections and categories of the FIX business messages if possible. The classification scheme can be expanded or replaced by	

			counterparty agreement. This approach permits the introduction and reference to other business ontologies. Example: posttrade/confirmation/confirm pretrade//termsheet	
-> 2108	AttachmentExternalURL	N	Used to specify an external URL where the attachment can be obtained.	
-> 2109	AttachmentEncodingType	N	The encoding type of the content provided in EncodedAttachment(2112). Valid values: 0 = Base64 1 = Raw binary (Elaboration: Unencoded binary content.) Reserved100+	
-> 2110	UnencodedAttachmentLen	N	Unencoded content length in bytes. Can be used to validate successful unencoding.	
-> 2111	EncodedAttachmentLen	N	Byte length of encoded the EncodedAttachment(2112) field.	
-> 2112	EncodedAttachment	N	The content of the attachment in the encoding format specified in the AttachmentEncodingType(2109) field.	
->	AttachmentKeywordGrp	N	Specifies the number of repeating lines of attachment keywords	
-> > 2113	NoAttachmentKeywords	N	Indicates the number of attachedment keywords	
-> > 2114	AttachmentKeyword	N	Can be used to provide data or keyword tagging of the content of the attachment.	
96	RawData	N	Unformatted raw data, can include bitmaps, word processor documents, etc.	
	StandardTrailer	Y	The standard FIX message trailer	

11.4 StreamAssignmentRequest (35=CC)

StreamAssignmentRequest (35=CC)		Execution Venue -> Dealer		
<p><i>In certain markets where market data aggregators fan out to end clients the pricing streams provided by the price makers, the price maker may assign the clients to certain pricing streams that the price maker publishes via the aggregator. An example of this use is in the FX markets where clients may be assigned to different pricing streams based on volume bands and currency pairs.</i></p>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType = CC	
1497	StreamAsgnReqID	Y	Unique identifier of the request.	
1498	StreamAsgnReqType	Y	Type of assignment being requested.	
	StrmAsgnReqGrp	Y	Assignment requests	
1499	NoAsgnReqs	N	Stream Assignment Requests.	
->	Parties	N	The Parties component block is used to identify and convey information on the entities both central and peripheral to the financial transaction represented by the FIX message containing the Parties Block. The Parties block allows many different types of entites to be expressed through use of the PartyRole field and identifies the source of the PartyID through the the PartyIDSource.	
->	StrmAsgnReqInstrmtGrp	N		
->-> 146	NoRelatedSym	N	Specifies the number of repeating symbols specified.	
->->->	Instrument	N	The Instrument component block contains all the fields commonly used to describe a security or instrument. Typically the data elements in this component block are considered the static data of a security, data that may be commonly found in a security master database. The Instrument component block can be used to describe any asset type supported by FIX.	
->->-> 63	SettlType	N	Indicates order settlement period. If present, SettlDate (64) overrides this field. If both SettlType (63) and SettlDate (64) are omitted, the default for SettlType (63) is 0 (Regular) Regular is defined as the default settlement period for the particular security on the exchange of execution. In Fixed Income the contents of this field may influence the instrument definition if the SecurityID (48) is ambiguous. In the US an active Treasury offering may be re-opened, and for a time one CUSIP	

			<p>will apply to both the current and "when-issued" securities. Supplying a value of "7" clarifies the instrument description; any other value or the absence of this field should cause the respondent to default to the active issue. Additionally the following patterns may be uses as well as enum values Dx = FX tenor expression for "days", e.g. "D5", where "x" is any integer > 0 Mx = FX tenor expression for "months", e.g. "M3", where "x" is any integer > 0 Wx = FX tenor expression for "weeks", e.g. "W13", where "x" is any integer > 0 Yx = FX tenor expression for "years", e.g. "Y1", where "x" is any integer > 0 Noted that for FX the tenors expressed using Dx, Mx, Wx, and Yx values do not denote business days, but calendar days.</p>	
->->-> 271	MDEntrySize	N	Quantity or volume represented by the Market Data Entry.	
->->-> 1500	MDStreamID	N	The identifier or name of the price stream.	

11.5 StreamAssignmentReport (35=CD)

StreamAssignmentReport (35=CD)		Dealer -> Execution Venue		
<p>The StreamAssignmentReport message is in response to the StreamAssignmentRequest message. It provides information back to the aggregator as to which clients to assign to receive which price stream based on requested CCY pair. This message can be sent unsolicited to the Aggregator from the Price Maker.</p>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType = CD	
1501	StreamAsgnRptID	Y	Unique identifier of the Stream Assignment Report.	
1498	StreamAsgnReqType	N	Required if report is being sent in response to a StreamAssignmentRequest. The value should be the same as the value in the corresponding request.	
1497	StreamAsgnReqID	N	Conditionally required if Stream Assignment Report is being sent in response to a StreamAssignmentRequest(MsgType=CC). Not required for unsolicited stream assignments.	
	StrmAsgnRptGrp	N	Stream assignments	
1499	NoAsgnReqs	N	Stream Assignment Reports.	
->	Parties	N	The Parties component block is used to identify and convey information on the entities both central and peripheral to the financial transaction represented by the FIX message containing the Parties Block. The Parties block allows many different types of entites to be expressed through use of the PartyRole field and identifies the source of the PartyID through the the PartyIDSource.	
->	StrmAsgnRptInstrmtGrp	N		
->-> 146	NoRelatedSym	N	Specifies the number of repeating symbols specified.	
->->->	Instrument	N	The Instrument component block contains all the fields commonly used to describe a security or instrument. Typically the data elements in this component block are considered the static data of a security, data that may be commonly found in a security master database. The Instrument component block can be used to describe any asset type supported by FIX.	
->->-> 63	SettlType	N	Indicates order settlement period. If present, SettlDate (64) overrides this field. If both SettlType (63) and SettlDate (64) are omitted, the default for SettlType (63) is 0 (Regular) Regular is defined as the default settlement period for the particular security on the exchange of execution. In Fixed Income the contents of this field may influence the instrument definition if the SecurityID (48) is ambiguous. In the US an active Treasury offering may be re-opened, and for a time one CUSIP will apply to both the current and "when-issued" securities. Supplying a value of "7"	

			clarifies the instrument description; any other value or the absence of this field should cause the respondent to default to the active issue. Additionally the following patterns may be used as well as enum values Dx = FX tenor expression for "days", e.g. "D5", where "x" is any integer > 0 Mx = FX tenor expression for "months", e.g. "M3", where "x" is any integer > 0 Wx = FX tenor expression for "weeks", e.g. "W13", where "x" is any integer > 0 Yx = FX tenor expression for "years", e.g. "Y1", where "x" is any integer > 0 Noted that for FX the tenors expressed using Dx, Mx, Wx, and Yx values do not denote business days, but calendar days.	
->->- >1617	<u>StreamAsgnType</u>	N	The type of assignment being affected in the Stream Assignment Report.	
->->- >1500	<u>MStreamID</u>	N	The identifier or name of the price stream.	
->->- >1502	<u>StreamAsgnRejReason</u>	N	Reason code for stream assignment request reject.	
->->- >58	<u>Text</u>	N	Free format text string (Note: this field does not have a specified maximum length)	
->->- >354	<u>EncodedTextLen</u>	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.	
->->- >355	<u>EncodedText</u>	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.	

11.6 StreamAssignmentReportAck (35=CE)

StreamAssignmentReportACK (35=CE)		Execution Venue -> Dealer		
<i>This message is used to respond to the StreamAssignmentReport, to either accept or reject an unsolicited assignment.</i>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType = CE	
1503	StreamAsgnAckType	Y	Type of acknowledgement.	
1501	StreamAsgnRptID	Y	Unique identifier of the stream assignment report provided by the respondent.	
1502	StreamAsgnRejReason	N	Reason code for stream assignment request reject.	
58	Text	N	Can be used to provide additional information regarding the assignment report, such as reject description.	
354	EncodedTextLen	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.	
355	EncodedText	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.	

11.7 PartyDetailsGrp component Diagram

The PartyDetailsGrp component is contained in the following messages:

- PartyDetailsListReport (35=CG)
- PartyDetailsListUpdateReport (35=CK)
- PartyEntitlementsReport (35=CV)
- PartyEntitlementsUpdateReport (35=CZ)
- PartyEntitlementsDefinitionRequest (35=DA)
- PartyEntitlementsDefinitionRequestAck (35=DB)

This component may be constructed in two different ways.

The following diagram illustrates the two possibilities of structuring Party Details Group component:

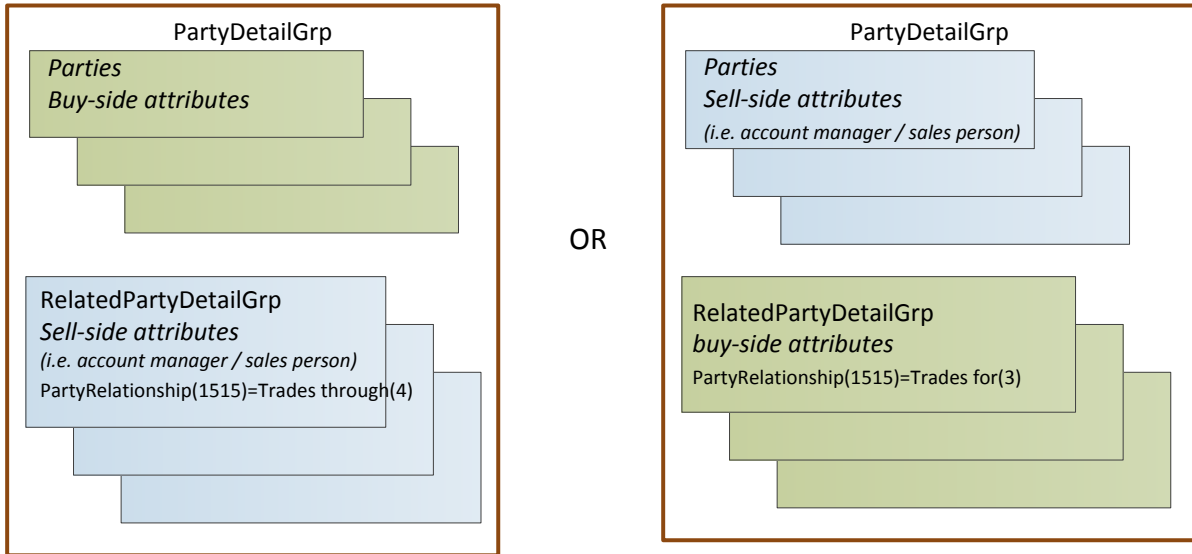


Figure 32: PartyDetailsGrp component Structure

11.8 PartyDetailListRequest (35=CF)

11.8.1 PartyDetailsListRequest(35=CF) Message Structure Diagram

The following diagram illustrates the structure of the PartyDetailsListRequest(35=CF) message:

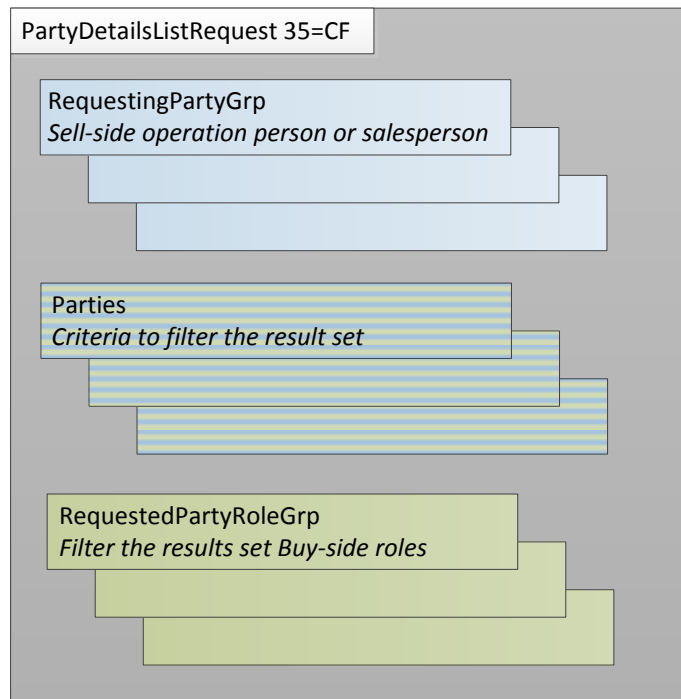


Figure 33: PartyDetailsListRequests(35=CF) Message Structure

11.8.2 PartyDetailsListRequest(35=CF) Message Definition

PartyDetailsListRequest (35=CF)		Dealer -> Execution Venue		
<i>The PartyDetailsListRequest is used to request party detail information.</i>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType = CF	
<u>1505</u>	<u>PartyDetailsListRequestID</u>	Y	Unique identifier for PartyDetailsListRequest.	
	RequestingPartyGrp	N	May be used to identify the party making the request and their role.	
<u>1657</u>	<u>NoRequestingPartyIDs</u>	N	Number of requesting party identifiers.	
-> <u>1658</u>	<u>RequestingPartyID</u>	N	Required when NoRequestingPartyIDs > 0.	
-> <u>1659</u>	<u>RequestingPartyIDSource</u>	N	Required when NoRequestingPartyIDs > 0.	
-> <u>1660</u>	<u>RequestingPartyRole</u>	N	Required when NoRequestingPartyIDs > 0.	
->	RequestingPartySubGrp	N	Sub identifiers for the requesting party.	
->-> <u>1661</u>	<u>NoRequestingPartySubIDs</u>	N	Number of requesting party sub-identifiers.	
->-> > <u>1662</u>	<u>RequestingPartySubID</u>	N	Required when NoRequestingPartySubIDs > 0.	
->-> > <u>1663</u>	<u>RequestingPartySubIDType</u>	N	Required when NoRequestingPartySubIDs > 0.	
	Parties	N	Scope of the query/request for specific party(-ies).	
<u>453</u>	<u>NoPartyIDs</u>	N	Repeating group below should contain unique combinations of PartyID, PartyIDSource, and PartyRole	
-> <u>448</u>	<u>PartyID</u>	N	Used to identify source of PartyID. Required if PartyIDSource is specified. Required if NoPartyIDs > 0.	
-> <u>447</u>	<u>PartyIDSource</u>	N	Used to identify class source of PartyID value (e.g. BIC). Required if PartyID is specified. Required if NoPartyIDs > 0.	
-> <u>452</u>	<u>PartyRole</u>	N	Identifies the type of PartyID (e.g. Executing Broker). Required if NoPartyIDs > 0.	
->	PtysSubGrp	N	Repeating group of Party sub-identifiers.	
	RequestedPartyRoleGrp	N	Scope of the query/request for specific party role(s)	
<u>1508</u>	<u>NoRequestedPartyRoles</u>	N	Number of requested party roles.	
-> <u>1509</u>	<u>RequestedPartyRole</u>	N	Identifies the type of party role requested. Required if NoRequestedPartyRoles > 0.	
	PartyRelationshipGrp	N	Scope of the query/request for specific party relationship(s)	
<u>1514</u>	<u>NoPartyRelationships</u>	N	Number of party relationships.	

-> 1515	PartyRelationship	N	Identifies the type of party relationship requested. Required if NoPartyRelationships > 0.
263	SubscriptionRequestType	N	Subscription Request Type
58	Text	N	Free format text string (Note: this field does not have a specified maximum length)
354	EncodedTextLen	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.
355	EncodedText	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.
	StandardTrailer	Y	The standard FIX message trailer

11.9 PartyDetailsListReport (35=CG)

11.9.1 PartyDetailsListReport(35=CG) Message Structure Diagram

The following diagram illustrates the structure of PartyDetailsListReport message:

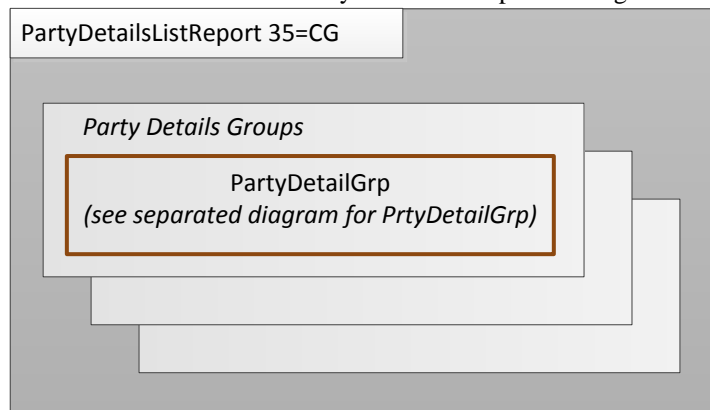


Figure 34: PartyDetailsListReport (35=CG) Message Structure

11.9.2 PartyDetailsListReport(35=CG) Message Definition

PartyDetailsListReport (35=CG)		Execution Venue -> Dealer		
<p>The PartyDetailsListReport message is used to disseminate party details between counterparties. PartyDetailsListReport messages may be sent in response to a PartyDetailsListRequest message or sent unsolicited.</p>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType = CG	
	ApplicationSequenceControl	N	The ApplicationSequenceControl is used for application sequencing and recovery. Consisting of ApplSeqNum (1181), ApplID (1180), ApplLastSeqNum (1350), and ApplResendFlag (1352), FIX application messages that carries this component block will be able to use application level sequencing. ApplID, ApplSeqNum and ApplLastSeqNum fields identify the application id, application sequence number and the previous application sequence number (in case of intentional gaps) on each application message that carries this block.	
1510	PartyDetailsListReportID	Y	Identifier for the PartyDetailsListReport and the PartyDetailsListUpdateReport.	
1505	PartyDetailsListRequestID	N	Conditionally required when responding to the PartyDetailsListRequest message.	
1511	RequestResult	N	Conditionally required when responding to the PartyDetailsListRequest message.	
1512	TotNoParties	N	Total number of PartyListGrp returned.	
893	LastFragment	N	Indicates whether this message is the last in a sequence of messages for those messages that support fragmentation, such as Allocation Instruction, Mass Quote, Security List, Derivative Security List	
	PartyDetailGrp	N	Contains details for a party, including related parties and alternative party identifiers.	
1671	NoPartyDetails	N	Number of party details.	
-> 1691	PartyDetailID	N	The identification of the party. Required when NoPartyDetails(1671) > 0.	
-> 1692	PartyDetailIDSource	N	Used to identify source of PartyID value (e.g. BIC). Required when NoPartyDetails(1671) > 0.	
-> 1693	PartyDetailRole	N	Identifies the type of PartyID (e.g. Executing Broker). Required when NoPartyDetails(1671) > 0.	

->1674	<u>PartyDetailRoleQualifier</u>	N	Qualifies the value of PartyRole(452)	
->	PartyDetailSubGrp	N	Additional party sub-identifiers	
->- >1694	<u>NoPartyDetailSubIDs</u>	Y	Number of party detail sub-identifiers.	
->->- >1695	<u>PartyDetailSubID</u>	N	Required when NoPartyDetailSubIDs > 0.	
->->- >1696	<u>PartyDetailSubIDType</u>	N	Required when NoPartyDetailSubIDs > 0.	
->	PartyDetailAltIDGrp	N	Optionally used to specify alternate IDs to identify the party specified.	
->	RelatedPartyDetailGrp	N	May not be specified in PartyDetailsListUpdateReport(35=CK) if ListUpdateAction(1324) = D(Delete)	
->- >1562	<u>NoRelatedPartyDetailID</u>	N	Number of related party detail identifiers.	
->->- >1563	<u>RelatedPartyDetailID</u>	N	Required if NoRelatedPartyDetails > 0.	
->->- >1564	<u>RelatedPartyDetailIDSource</u>	N	Required if NoRelatedPartyDetails > 0.	
->->- >1565	<u>RelatedPartyDetailRole</u>	N	Required if NoRelatedPartyDetails > 0.	
->->- >1675	<u>RelatedPartyDetailRoleQualifier</u>	N	Qualifies the value of RelatedPartyRole(1565)	
->->->	RelatedPartyDetailSubGrp	N	PartySubGrp for related parties.	
->->->- >1566	<u>NoRelatedPartyDetailSubIDs</u>	N	Number of related party detail sub-identifiers.	
->->->- >	<u>RelatedPartyDetailSubID</u>	N	Required when NoRelatedPartyDetailSubIDs > 0.	
->->->- >	<u>RelatedPartyDetailSubIDType</u>	N	Required when NoRelatedPartyDetailSubIDs > 0.	
->->->	RelatedPartyDetailAltIDGrp	N	Alternative identifiers for parties related to the party specified in the PartyDetailGrp.	
->->->	PartyRelationshipGrp	N	Repeating group of party relationships.	
->->->- >1514	<u>NoPartyRelationships</u>	N	Number of party relationships.	
->->->- >	<u>PartyRelationship</u>	N	Identifies the type of party relationship requested. Required if NoPartyRelationships > 0.	
->1672	<u>PartyDetailStatus</u>	N	Specifies the status of the party information, whether active or suspended (inactive).	
60	<u>TransactTime</u>	N	Timestamp when the business transaction represented by the message occurred.	
58	<u>Text</u>	N	Free format text string (Note: this field does not have a specified maximum length)	
354	<u>EncodedTextLen</u>	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.	

<u>355</u>	<u>EncodedText</u>	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.	
<u>1328</u>	<u>RejectText</u>	N	Identifies the reason for rejection.	
<u>1664</u>	<u>EncodedRejectTextLen</u>	N	Byte length of encoded (non-ASCII characters) EncodedRejectText(1665) field.	
<u>1665</u>	<u>EncodedRejectText</u>	N	Encoded (non-ASCII characters) representation of the RejectText(1328) field in the encoded format specified via the MessageEncoding(347) field. If used, the ASCII (English) representation should also be specified in the RejectText(1328) field.	
	StandardTrailer	Y	The standard FIX message trailer	

11.10 PartyDetailsListUpdateReport (35=CK)

11.10.1 PartyDetailsListUpdateReport (35=CK) Message Structure Diagram

The following diagram illustrates the structure of PartyDetailsListUpdateReport message:

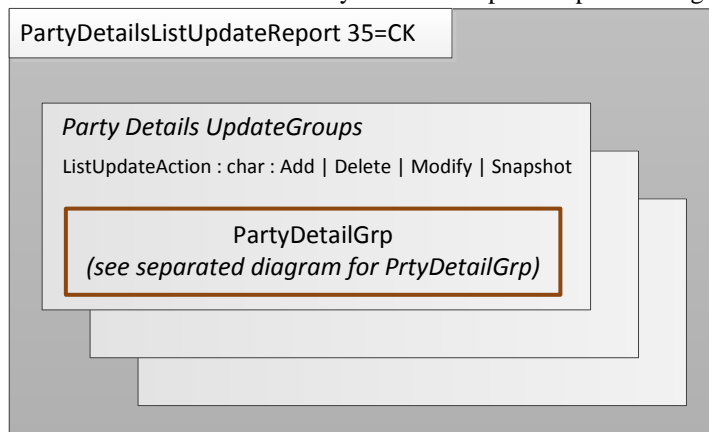


Figure 35: PartyDetailsListUpdateReport (35=CK) Message Structure

11.10.2 PartyDetailsListUpdateReport(35=CK) Message Definition

PartyDetailsListUpdateReport (35=CK)		Execution Venue -> Dealer		
<i>The PartyDetailsListUpdateReport is used to disseminate updates to party detail information.</i>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType = CK	
	ApplicationSequenceControl	N	The ApplicationSequenceControl is used for application sequencing and recovery. Consisting of ApplSeqNum (1181), ApplID (1180), ApplLastSeqNum (1350), and ApplResendFlag (1352), FIX application messages that carries this component block will be able to use application level sequencing. ApplID, ApplSeqNum and ApplLastSeqNum fields identify the application id, application sequence number and the previous application sequence number (in case of intentional gaps) on each application message that carries this block.	
1510	PartyDetailsListReportID	Y	Identifier for the PartyDetailsListReport and the PartyDetailsListUpdateReport.	
1505	PartyDetailsListRequestID	N	Conditionally required when responding to the PartyDetailsListRequest message.	
1512	TotNoParties	N	Total number of PartyListGrp returned.	
893	LastFragment	N	Indicates whether this message is the last in a sequence of messages for those messages that support fragmentation, such as Allocation Instruction, Mass Quote, Security List, Derivative Security List	
	PartyDetailsUpdateGrp	N	Party details component that includes an update action.	
1676	NoPartyUpdates	N	Number of party updates.	
- >1324	ListUpdateAction	N	Required if NoPartyUpdates > 0.	
->	PartyDetailGrp	N	Contains details for a party, including related parties and alternative party identifiers.	
->- >1671	NoPartyDetails	N	Number of party details.	
->->- >1691	PartyDetailID	N	The identification of the party. Required when NoPartyDetails(1671) > 0.	
->->- >1692	PartyDetailIDSource	N	Used to identify source of PartyID value (e.g. BIC). Required when NoPartyDetails(1671) > 0.	
->->- >1693	PartyDetailRole	N	Identifies the type of PartyID (e.g. Executing Broker). Required when	

			NoPartyDetails(1671) > 0.	
->-> >1674	<u>PartyDetailRoleQualifier</u>	N	Qualifies the value of PartyRole(452)	
->->>	PartyDetailSubGrp	N	Additional party sub-identifiers	
->->> > >1694	<u>NoPartyDetailSubIDs</u>	Y	Number of party detail sub-identifiers.	
->->> >->- >1695	<u>PartyDetailSubID</u>	N	Required when NoPartyDetailSubIDs > 0.	
->->> >->- >1696	<u>PartyDetailSubIDType</u>	N	Required when NoPartyDetailSubIDs > 0.	
->->>>	PartyDetailAltIDGrp	N	Optionally used to specify alternate IDs to identify the party specified.	
->->>>	RelatedPartyDetailGrp	N	May not be specified in PartyDetailsListUpdateReport(35=CK) if ListUpdateAction(1324) = D(Delete)	
->->>> >->- >1562	<u>NoRelatedPartyDetailID</u>	N	Number of related party detail identifiers.	
->->>> >->- >1563	<u>RelatedPartyDetailID</u>	N	Required if NoRelatedPartyDetails > 0.	
->->>> >->- >1564	<u>RelatedPartyDetailIDSource</u>	N	Required if NoRelatedPartyDetails > 0.	
->->>> >->- >1565	<u>RelatedPartyDetailRole</u>	N	Required if NoRelatedPartyDetails > 0.	
->->>> >->- >1675	<u>RelatedPartyDetailRoleQualifier</u>	N	Qualifies the value of RelatedPartyRole(1565)	
->->>> >->->>	RelatedPartyDetailSubGrp	N	PartySubGrp for related parties.	
->->>> >->->> >1566	<u>NoRelatedPartyDetailSubIDs</u>	N	Number of related party detail sub-identifiers.	
->->>> >->->> >->- >1567	<u>RelatedPartyDetailSubID</u>	N	Required when NoRelatedPartyDetailSubIDs > 0.	
->->>> >->->> >->- >1568	<u>RelatedPartyDetailSubIDType</u>	N	Required when NoRelatedPartyDetailSubIDs > 0.	
->->>> >->->>	RelatedPartyDetailAltIDGrp	N	Alternative identifiers for parties related to the party specified in the PartyDetailGrp.	
->->>> >->->>	PartyRelationshipGrp	N	Repeating group of party relationships.	
->->>> >->->>	<u>NoPartyRelationships</u>	N	Number of party relationships.	

>1514				
->->- >->->- >- >1515	PartyRelationship	N	Identifies the type of party relationship requested. Required if NoPartyRelationships > 0.	
->->- >1672	PartyDetailStatus	N	Specifies the status of the party information, whether active or suspended (inactive).	
60	TransactTime	N	Timestamp when the business transaction represented by the message occurred.	
58	Text	N	Free format text string (Note: this field does not have a specified maximum length)	
354	EncodedTextLen	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.	
355	EncodedText	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.	
	StandardTrailer	Y	The standard FIX message trailer	

11.11 PartyEntitlementsRequest (35=CU)

11.11.1 PartyEntitlementsRequest (35=CU) Message Structure Diagram

The following diagram illustrates the structure of the PartyEntitlementsRequest message:

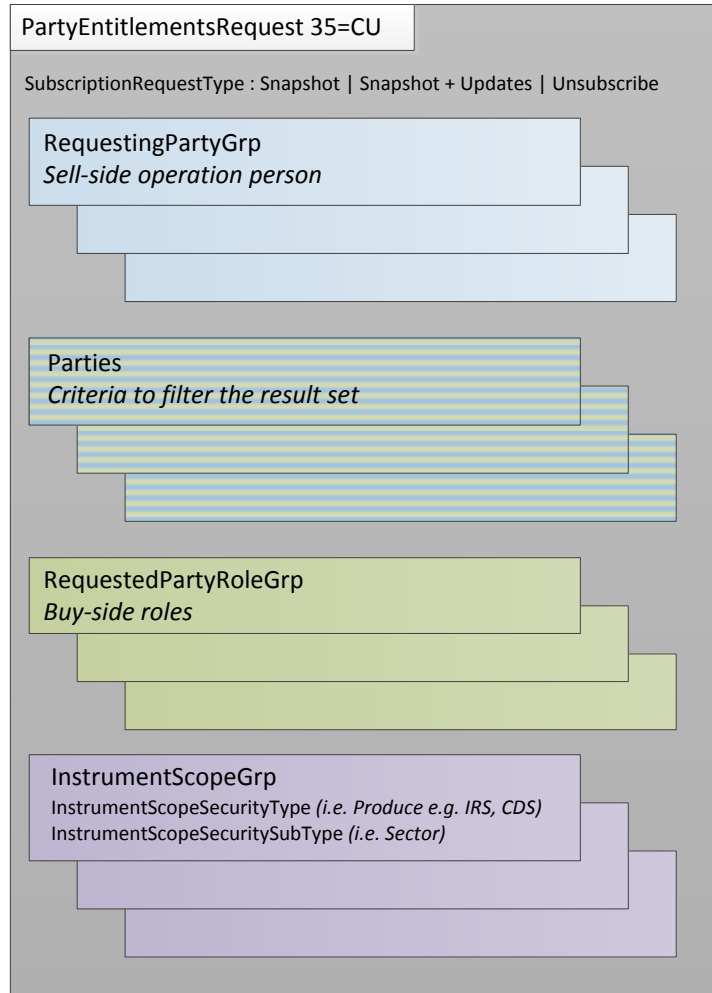


Figure 36: PartyEntitlementsRequest (35=CU) Message Structure

11.11.2 PartyEntitlementRequest (35=CU) Example-1

In this example the sell-side operation person, (i.e. Jane Doe) requests all the entitlements that are associated with specific sales person (i.e. James Bradley).

11.11.2.1 Party Entitlement Request FIXML

```
<?xml version="1.0" encoding="UTF-8"?>
<FIXML xmlns="http://www.fixprotocol.org/FIXML-5-0-SP2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.fixprotocol.org/FIXML-5-0-SP2 ./Fixml/fixml-main-5-0-SP2.xsd">
  <PtyEntlmntReq ReqID="1">
    <ReqPty ID="Bank-1" Src="D" R="1">
      <Sub ID="Jane Doe" Typ="2"/>
    </ReqPty>
    <Pty ID="Bank-1" Src="D" R="1"> <!-- Results set contains only entitlements associated
with sales person: James Bradley -->
      <Sub ID="James Bradley" Typ="9" /> <!-- Sales person -->
      <Sub ID="James-Bradley-ID" Typ="2" /> <!-- Sales person ID -->
    </Pty>
  </PtyEntlmntReq>
</FIXML>
```

11.11.3 PartyEntitlementRequest (35=CU) Example-2

In this example the sell-side operation person, (i.e. Jane Doe) requests all the entitlements for a specific buy side firm (i.e. Hedge-Fund).

11.11.3.1 Party Entitlement Request FIXML

```
<?xml version="1.0" encoding="UTF-8"?>
<FIXML xmlns="http://www.fixprotocol.org/FIXML-5-0-SP2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.fixprotocol.org/FIXML-5-0-SP2 ./Fixml/fixml-main-5-0-SP2.xsd">
  <PtyEntlmntReq ReqID="1">
    <ReqPty ID="Bank-1" Src="D" R="1">
      <Sub ID="Jane Doe" Typ="2"/>
    </ReqPty>
    <Pty ID="User-1" Src="D" R="3"> <!-- Results set contains only entitlements of firm
'Hedge Fund' -->
      <Sub ID="Hedge Fund" Typ="1"/> <!-- firm -->
    </Pty>
  </PtyEntlmntReq>
</FIXML>
```

11.11.4 PartyEntitlementsRequest (35=CU) Message Definition

PartyEntitlementsRequest (35=CU)		Dealer -> Execution Venue		
<i>The PartyEntitlementsRequest message is used to request for entitlement information for one or more party(-ies), specific party role(s), or specific instruments(s).</i>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType=CU	
1770	EntitlementRequestID	N	Unique identifier for PartyEntitlementsRequest(35=CU).	
263	SubscriptionRequestType	N	Subscription Request Type	
	RequestingPartyGrp	N	May be used to identify the party making the request and their role.	
1657	NoRequestingPartyIDs	N	Number of requesting party identifiers.	
-> 1658	RequestingPartyID	N	Required when NoRequestingPartyIDs > 0.	
-> 1659	RequestingPartyIDSource	N	Required when NoRequestingPartyIDs > 0.	
-> 1660	RequestingPartyRole	N	Required when NoRequestingPartyIDs > 0.	
-> 1883	EntitlementStatus	N	Status of entitlement definition for one party.	
->	EntitlementTypeGrp	N	The EntitlementTypeGrp when added to PartyEntitlementsRequest serves as filter criteria for the results set.	
->-> 2345	NoEntitlementTypes	N	Number of Entitlement Types.	
->-> 1775	EntitlementType	N	Required if NoEntitlementTypes>0	
->	RequestingPartySubGrp	N	Sub identifiers for the requesting party.	
->-> 1661	NoRequestingPartySubIDs	N	Number of requesting party sub-identifiers.	
->->-> 1662	RequestingPartySubID	N	Required when NoRequestingPartySubIDs > 0.	
->->-> 1663	RequestingPartySubIDType	N	Required when NoRequestingPartySubIDs > 0.	
	Parties	N	Scope of the query/request for specific party(-ies).	
453	NoPartyIDs	N	Repeating group below should contain unique combinations of PartyID, PartyIDSource, and PartyRole	
-> 448	PartyID	N	Used to identify source of PartyID. Required if PartyIDSource is	

			specified. Required if NoPartyIDs > 0.	
->447	PartyIDSource	N	Used to identify class source of PartyID value (e.g. BIC). Required if PartyID is specified. Required if NoPartyIDs > 0.	
->452	PartyRole	N	Identifies the type of PartyID (e.g. Executing Broker). Required if NoPartyIDs > 0.	
->	PtysSubGrp	N	Repeating group of Party sub-identifiers.	
	RequestedPartyRoleGrp	N	Scope of the query/request for specific party roles. For example, "all information for PartyRole=24".	
1508	NoRequestedPartyRoles	N	Number of requested party roles.	
->1509	RequestedPartyRole	N	Identifies the type of party role requested. Required if NoRequestedPartyRoles > 0.	
1784	EntitlementPlatform	N	The area to which the entitlement is applicable. This can be a trading platform or an offering.	
	InstrumentScopeGrp	N	Scope of the query/request for specific securities.	
1656	NoInstrumentScopes	N	Number of instrument scopes.	
->1535	InstrumentScopeOperator	N	Required when NoInstrumentScopes > 0.	
->	InstrumentScope	N	Used to specify the instrument	
->->1536	InstrumentScopeSymbol	N	Used to limit instrument scope to specified symbol. See Symbol(55) field for description.	
->->1537	InstrumentScopeSymbolSfx	N	Used to limit instrument scope to specified symbol suffix. See SymbolSfx(65) field for description.	
->->1538	InstrumentScopeSecurityID	N	Used to limit instrument scope to specified security identifier. See SecurityID(48) field for description.	
->->1539	InstrumentScopeSecurityIDSource	N	Used to limit instrument scope to specified security identifier source. See SecurityIDSource(22) field for description.	
->->	InstrumentScopeSecurityAltIDGrp	N	Alternative SecurityIDs for an instrument specified in the InstrumentScope.	
->->1543	InstrumentScopeProduct	N	Used to limit instrument scope to specified instrument product category. See Product (460) field for description.	
->->1544	InstrumentScopeProductComplex	N	Used to limit instrument scope to specified product complex. See ProductComplex(1227) field for description.	
->->	InstrumentScopeSecurityGroup	N	Used to limit instrument scope to	

> 154 5			specified security group. See SecurityGroup(1151) field for description.	
->- > 154 6	InstrumentScopeCFIcode	N	Used to limit instrument scope to specified CFIcode. See CFIcode(461) field for description.	
->- > 154 7	InstrumentScopeSecurityType	N	Used to limit instrument scope to specified security type. See SecurityType(167) field for description).	
->- > 154 8	InstrumentScopeSecuritySubType	N	Used to limit instrument scope to specified security sub-type. See SecuritySubType(762) field for description.	
->- > 154 9	InstrumentScopeMaturityMonthYear	N	Used to limit instrument scope to specified maturity month and year. See MaturityMonthYear(200) field for description.	
->- > 155 0	InstrumentScopeMaturityTime	N	Used to limit instrument scope to specified maturity time. See MaturityTime(1079) field for description.	
->- > 155 1	InstrumentScopeRestructuringType	N	Used to limit instrument scope to specified restructuring type. See RestructuringType(1449) field for description.	
->- > 155 2	InstrumentScopeSeniority	N	Used to limit instrument scope to specified seniority type. See Seniority(1450) field for description.	
->- > 155 3	InstrumentScopePutOrCall	N	Used to limit instrument scope to puts or calls. See PutOrCall(201) field for description.	
->- > 155 4	InstrumentScopeFlexibleIndicator	N	Used to limit instrument scope to securities that can be defined using flexible terms or not. See FlexibleIndicator(1244) field for description.	
->- > 155 5	InstrumentScopeCouponRate	Y	Used to limit instrument scope to specified coupon rate. See CouponRate(223) field for description.	
->- > 161 6	InstrumentScopeSecurityExchange	N	Used to limit instrument scope to specified security exchange. See SecurityExchange(207) field for description.	
->- > 155 6	InstrumentScopeSecurityDesc	N	Used to limit instrument scope to specified security description. See SecurityDesc(107) field for description.	
->- > 162 0	InstrumentScopeEncodedSecurityDesc Len	N	Byte length of encoded (non-ASCII characters) InstrumentScopeEncodedSecurityDesc (1621) field	
->-	InstrumentScopeEncodedSecurityDesc	N	Encoded (non-ASCII characters)	

>162 1			representation of the InstrumentScopeSecurityDesc (1556) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the InstrumentScopeSecurityDesc field.	
->- >155 7	<u>InstrumentScopeSettlType</u>	N	Can be used to specify FX tenors.	
58	<u>Text</u>	N	Free format text string (Note: this field does not have a specified maximum length)	
354	<u>EncodedTextLen</u>	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.	
355	<u>EncodedText</u>	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.	
	StandardTrailer	Y	The standard FIX message trailer	

11.12 PartyEntitlementsReport (35=CV)

11.12.1 PartyEntitlementsReport (35=CV) Message Structure Diagram

The following diagram illustrates the structure of the PartyEntitlementsReport message:

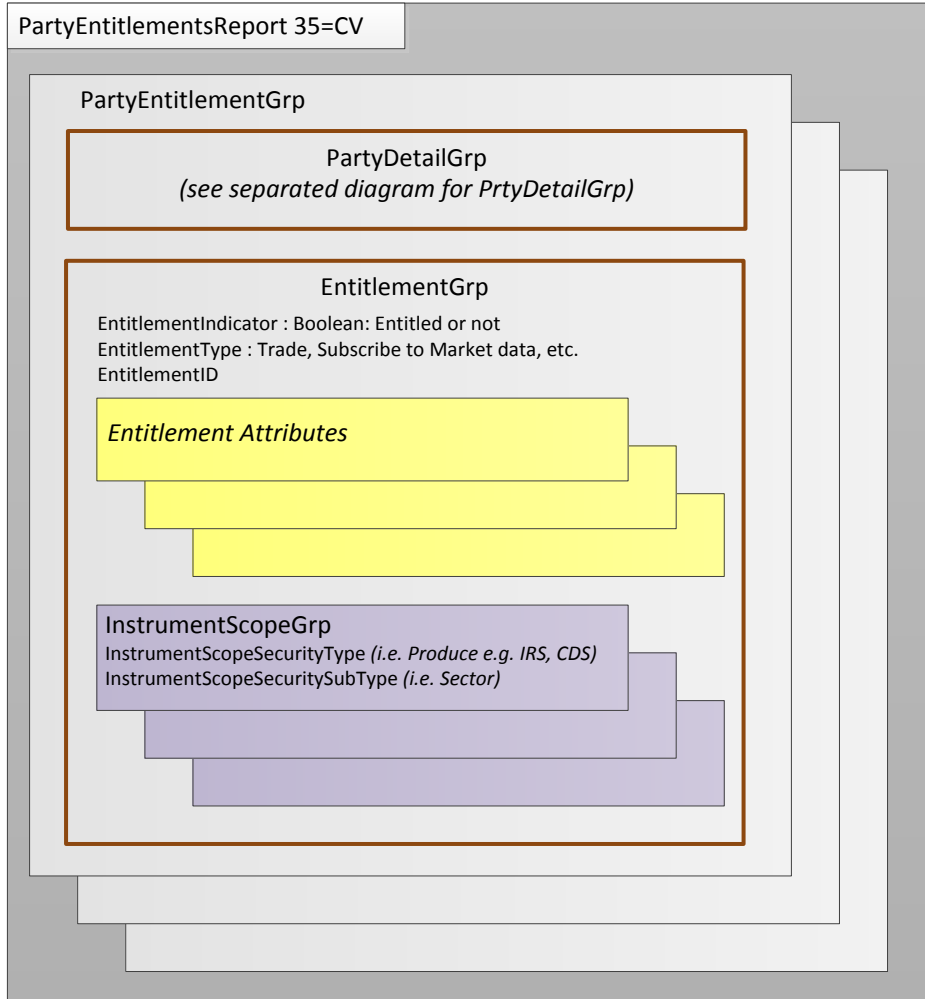


Figure 37: PartyEntitlementsReport (35=CV) Message Structure

11.12.2 PartyEntitlementReport (35=CV) Example

In this example the Execution Venue sends an entitlement report containing data of one user. The data may be summarised as

1. A buy-side user (User 1) is entitled to trade EUR/USD FX Spot, EUR/USD FX Forward and GBP/USD FX Swap, assigned to a Level-1 Pricing stream (Market Data Stream) and has a Credit Limit of USD 10million with Bank-1

The below table contains the business level attributes of the entitlement report:

Attribute	Entitled User
Buy-side firm Identifier	Hedge Fund-AB
Buy-side User ID	User-1
Buy-side user name	Jon A. Smith
Email	jon.a.smith@HedgeFund.com
LEI	LZ123
Sell-side Identifier	Bank-1
Bank Account Identifier Mapping [1]	FundAcc-1
Bank Account Identifier Mapping [2]	FundAcc-2
Instrument Type	FX Spot
Instrument currency pair	EUR/USD
Instrument Type	FX Forward
Instrument currency pair	EUR/USD
Instrument Type	FX Swap
Instrument currency pair	GBP/USD
Enablement Type	Trade
Enablement Status	Enabled
Account Identifier [1]	Account 1
Account Identifier [2]	Account 2
Market Data Stream	Level-1
Credit Limit	10000000
Credit Method	Net
Credit Currency	USD

Table 25: Example 1 – Execution Venue sends an entitlement report

11.12.2.1 Party Entitlement Report FIXML

The following is the FIXML message tags and values that are expected in this example

```
<?xml version="1.0" encoding="UTF-8"?>
<FIXML xmlns="http://www.fixprotocol.org/FIXML-5-0-SP2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.fixprotocol.org/FIXML-5-0-SP2 ./Fixml/fixml-main-5-0-
SP2.xsd">
  <PtyEntlmntRpt RptID="1" TotNoPtys="1" >
    <PtyEntlmnt>
      <!-- buy-side user attributes-->
      <PtyDetl ID="User-1" Src="D" R="3" Stat="1"> <!-- Buy-side User ID Identifier-->
        <Sub ID="Jon A. Smith" Typ="9"/> <!-- User name -->
        <Sub ID="Hedge Fund-AB" Typ="1"/> <!-- Firm Identifier -->
        <Sub ID="jon.a.smith@HedgeFund.com" Typ="8" /> <!-- Email -->
        <Sub ID="Account 1" Typ="19" /> <!-- Buy side account ID 1 -->
        <Sub ID="Account 2" Typ="19" /> <!-- Buy side account ID 2 -->
        <AltPty ID="LZ123" Src="N"/> <!-- LEI -->
        <ReltdPtyDetl ID="Bank-1" Src="D" R="1"> <!-- Sell-side attributes -->
          <Sub ID="FundAcc-1" Typ="19" /> <!-- Bank Account ID mapping 1-->
          <Sub ID="FundAcc-2" Typ="19" /> <!-- Bank Account ID mapping 2-->
        </ReltdPtyDetl>
      </PtyDetl>
      <Entlmnt Ind="Y" Typ="0" > <!-- Entitled to Trade -->
        <Attrib Typ="1500" Datatyp="5" Value="Level-1" /> <!--FIX tag MDStreamID(1500)-
->
        <!-- Custom attribute: Credit limit -->
        <Attrib Typ="4000" Datatyp="1" Value="10000000" />
        <Attrib Typ="4001" Datatyp="5" Value="net" />
        <Attrib Typ="4002" Datatyp="16" Value="usd" />
        <!-- Instrument Scope-->
        <InstrmtScope Oper="1" Sym="eur/usd" Prod="4" SecTyp="FXSPOT"/>
        <InstrmtScope Oper="1" Sym="eur/usd" Prod="4" SecTyp="FXFWD"/>
        <InstrmtScope Oper="1" Sym="gbp/usd" Prod="4" SecTyp="FXSWAP"/>
      </Entlmnt>
    </PtyEntlmnt>
  </PtyEntlmntRpt>
</FIXML>
```

11.12.3 PartyEntitlementsReport (35=CV) Message Definition

PartyEntitlementsReport (35=CV)		Execution Venue -> Dealer		
<i>The PartyEntitlementsReport is used to report entitlements for one or more parties, party role(s), or specific instrument(s).</i>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType=CV	
	ApplicationSequenceControl	N	The ApplicationSequenceControl is used for application sequencing and recovery. Consisting of ApplSeqNum (1181), ApplID (1180), ApplLastSeqNum (1350), and ApplResendFlag (1352), FIX application messages that carries this component block will be able to use application level sequencing. ApplID, ApplSeqNum and ApplLastSeqNum fields identify the application id, application sequence number and the previous application sequence number (in case of intentional gaps) on each application message that carries this block.	
1771	EntitlementReportID	Y	Identifier for the PartyEntitlementsReport(35=CV).	
1770	EntitlementRequestID	N	Conditionally required when responding to PartyEntitlementsRequest(35=CU).	
1511	RequestResult	N	Conditionally required when responding to Party Entitlements Request.	
1512	TotNoParties	N	Total number of PartyListGrp returned.	
893	LastFragment	N	Indicates whether this message is the last in a sequence of messages for those messages that support fragmentation, such as Allocation Instruction, Mass Quote, Security List, Derivative Security List	
	PartyEntitlementGrp	N	Conveys a list of parties (optionally including related parties) and the entitlements for each.	
1772	NoPartyEntitlements	N	Number of party entitlement values.	
->	PartyDetailGrp	N	Required if NoPartyEntitlements(1772) > 0.	
->- >167 1	NoPartyDetails	N	Number of party details.	
->->- >169 1	PartyDetailID	N	The identification of the party. Required when NoPartyDetails(1671) > 0.	
->->-	PartyDetailIDSource	N	Used to identify source of PartyID	

>169 2			value (e.g. BIC). Required when NoPartyDetails(1671) > 0.	
->-> >169 3	PartyDetailRole	N	Identifies the type of PartyID (e.g. Executing Broker). Required when NoPartyDetails(1671) > 0.	
->-> >167 4	PartyDetailRoleQualifier	N	Qualifies the value of PartyRole(452)	
- >188 3	EntitlementStatus	N	Status of entitlement definition for one party.	
->-> >	PartyDetailSubGrp	N	Additional party sub-identifiers	
->-> > >169 4	NoPartyDetailSubIDs	Y	Number of party detail sub-identifiers.	
->-> >-> >169 5	PartyDetailSubID	N	Required when NoPartyDetailSubIDs > 0.	
->-> >-> >169 6	PartyDetailSubIDType	N	Required when NoPartyDetailSubIDs > 0.	
->-> >	PartyDetailAltIDGrp	N	Optionally used to specify alternate IDs to identify the party specified.	
->	EntitlementGrp	N	Required unless omitted to indicate the removal of entitlements for the party(-ies) specified in the PartyDetailGrp component.	
->- >177 3	NoEntitlements	N	Number of entitlement values.	
->-> >177 4	EntitlementIndicator	N	Required if NoEntitlements(1773) > 0.	
->-> >177 5	EntitlementType	N	Absence of this field indicates the meaning of the entitlement is implicit.	
->-> >	EntitlementAttribGrp	N	conveys a list of one or more attributes related to an entitlement. An entitlement may contain an EntitlementType, which states what can be done at a gross level, e.g. that a party can make markets. It may be limited further within EntitlementGrp, e.g. that such market making is allowed only for a list of stocks. The EntitlementAttribGrp contains fine details clarifying or limiting the	

			EntitlementType, e.g. that such market making must be conducted with a specific minimum quote size and a specific maximum spread.	
->-> >177 6	<u>EntitlementID</u>	N	Unique identifier for a specific NoEntitlements(1773) repeating group instance.	
->-> >178 4	<u>EntitlementPlatform</u>	N	The area to which the entitlement is applicable. This can be a trading platform or an offering.	
->-> >	InstrumentScopeGrp	N	Repeating group of InstrumentScope Components. Used to specify the instruments to which a request applies.	
->-> > >165 6	<u>NoInstrumentScopes</u>	N	Number of instrument scopes.	
->-> >>- >153 5	<u>InstrumentScopeOperator</u>	N	Required when NoInstrumentScopes > 0.	
->-> >>->	InstrumentScope	N	Used to specify the instrument	
->-> >>- > >153 6	<u>InstrumentScopeSymbol</u>	N	Used to limit instrument scope to specified symbol. See Symbol(55) field for description.	
->-> >>- > >153 7	<u>InstrumentScopeSymbolSfx</u>	N	Used to limit instrument scope to specified symbol suffix. See SymbolSfx(65) field for description.	
->-> >>- > >153 8	<u>InstrumentScopeSecurityID</u>	N	Used to limit instrument scope to specified security identifier. See SecurityID(48) field for description.	
->-> >>- > >153 9	<u>InstrumentScopeSecurityIDSource</u>	N	Used to limit instrument scope to specified security identifier source. See SecurityIDSource(22) field for description.	
->-> >>- >>	InstrumentScopeSecurityAltIDGrp	N	Alternative SecurityIDs for an instrument specified in the InstrumentScope.	
->-> >>- > >154 3	<u>InstrumentScopeProduct</u>	N	Used to limit instrument scope to specified instrument product category. See Product (460) field for description.	
->->	<u>InstrumentScopeProductComplex</u>	N	Used to limit instrument scope to	

>>- >- >154 <u>4</u>			specified product complex. See ProductComplex(1227) field for description.	
->>- >>- >- >154 <u>5</u>	<u>InstrumentScopeSecurityGroup</u>	N	Used to limit instrument scope to specified security group. See SecurityGroup(1151) field for description.	
->>- >>- >- >154 <u>6</u>	<u>InstrumentScopeCFIcode</u>	N	Used to limit instrument scope to specified CFIcode. See CFIcode(461) field for description.	
->>- >>- >- >154 <u>7</u>	<u>InstrumentScopeSecurityType</u>	N	Used to limit instrument scope to specified security type. See SecurityType(167) field for description).	
->>- >>- >- >154 <u>8</u>	<u>InstrumentScopeSecuritySubType</u>	N	Used to limit instrument scope to specified security sub-type. See SecuritySubType(762) field for description.	
->>- >>- >- >154 <u>9</u>	<u>InstrumentScopeMaturityMonthYear</u>	N	Used to limit instrument scope to specified maturity month and year. See MaturityMonthYear(200) field for description.	
->>- >>- >- >155 <u>0</u>	<u>InstrumentScopeMaturityTime</u>	N	Used to limit instrument scope to specified maturity time. See MaturityTime(1079) field for description.	
->>- >>- >- >155 <u>1</u>	<u>InstrumentScopeRestructuringType</u>	N	Used to limit instrument scope to specified restructuring type. See RestructuringType(1449) field for description.	
->>- >>- >- >155 <u>2</u>	<u>InstrumentScopeSeniority</u>	N	Used to limit instrument scope to specified seniority type. See Seniority(1450) field for description.	
->>- >>- >- >155 <u>3</u>	<u>InstrumentScopePutOrCall</u>	N	Used to limit instrument scope to puts or calls. See PutOrCall(201) field for description.	
->>- >>- >- >155 <u>4</u>	<u>InstrumentScopeFlexibleIndicator</u>	N	Used to limit instrument scope to securities that can be defined using flexible terms or not. See FlexibleIndicator(1244) field for description.	

->->- >->- >- >155 5	<u>InstrumentScopeCouponRate</u>	Y	Used to limit instrument scope to specified coupon rate. See CouponRate(223) field for description.	
->->- >->- >- >161 6	<u>InstrumentScopeSecurityExchange</u>	N	Used to limit instrument scope to specified security exchange. See SecurityExchange(207) field for description.	
->->- >->- >- >155 6	<u>InstrumentScopeSecurityDesc</u>	N	Used to limit instrument scope to specified security description. See SecurityDesc(107) field for description.	
->->- >->- >- >162 0	<u>InstrumentScopeEncodedSecurityDescLen</u>	N	Byte length of encoded (non-ASCII characters) InstrumentScopeEncodedSecurityDesc (1621) field	
->->- >->- >- >162 1	<u>InstrumentScopeEncodedSecurityDesc</u>	N	Encoded (non-ASCII characters) representation of the InstrumentScopeSecurityDesc (1556) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the InstrumentScopeSecurityDesc field.	
->->- >->- >- >155 7	<u>InstrumentScopeSettlType</u>	N	Can be used to specify FX tenors.	
->->- >	MarketSegmentScopeGrp	N	conveys a list of markets and, optionally, their market segments. Note that the component MarketSegmentGrp exists, but is not useful for this purpose, as it conveys additional information not appropriate in this context.	
->->- >- >131 0	<u>NoMarketSegments</u>	N	Number of Market Segments on which a security may trade.	
->->- >->- >130 1	<u>MarketID</u>	N	Required if NoMarketSegments(1310) > 0.	
->->- >->- >130 0	<u>MarketSegmentID</u>	N	Identifies the market segment	

->-> >178 2	<u>EntitlementStartDate</u>	N	Indicates the starting date of the entitlement.	
->-> >178 3	<u>EntitlementEndDate</u>	N	Indicates the starting date of the entitlement.	
60	<u>TransactTime</u>	N	Timestamp when the business transaction represented by the message occurred.	
58	<u>Text</u>	N	Free format text string (Note: this field does not have a specified maximum length)	
354	<u>EncodedTextLen</u>	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.	
355	<u>EncodedText</u>	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.	
1328	<u>RejectText</u>	N	Identifies the reason for rejection.	
1664	<u>EncodedRejectTextLen</u>	N	Byte length of encoded (non-ASCII characters) EncodedRejectText(1665) field.	
1665	<u>EncodedRejectText</u>	N	Encoded (non-ASCII characters) representation of the RejectText(1328) field in the encoded format specified via the MessageEncoding(347) field. If used, the ASCII (English) representation should also be specified in the RejectText(1328) field.	
	StandardTrailer	Y	The standard FIX message trailer	

11.13 PartyEntitlementsUpdateReport (35=CZ)

11.13.1 PartyEntitlementsUpdateReport (35=CZ) Message Structure Diagram

The following diagram illustrates the structure of the PartyEntitlementsUpdateReport message:

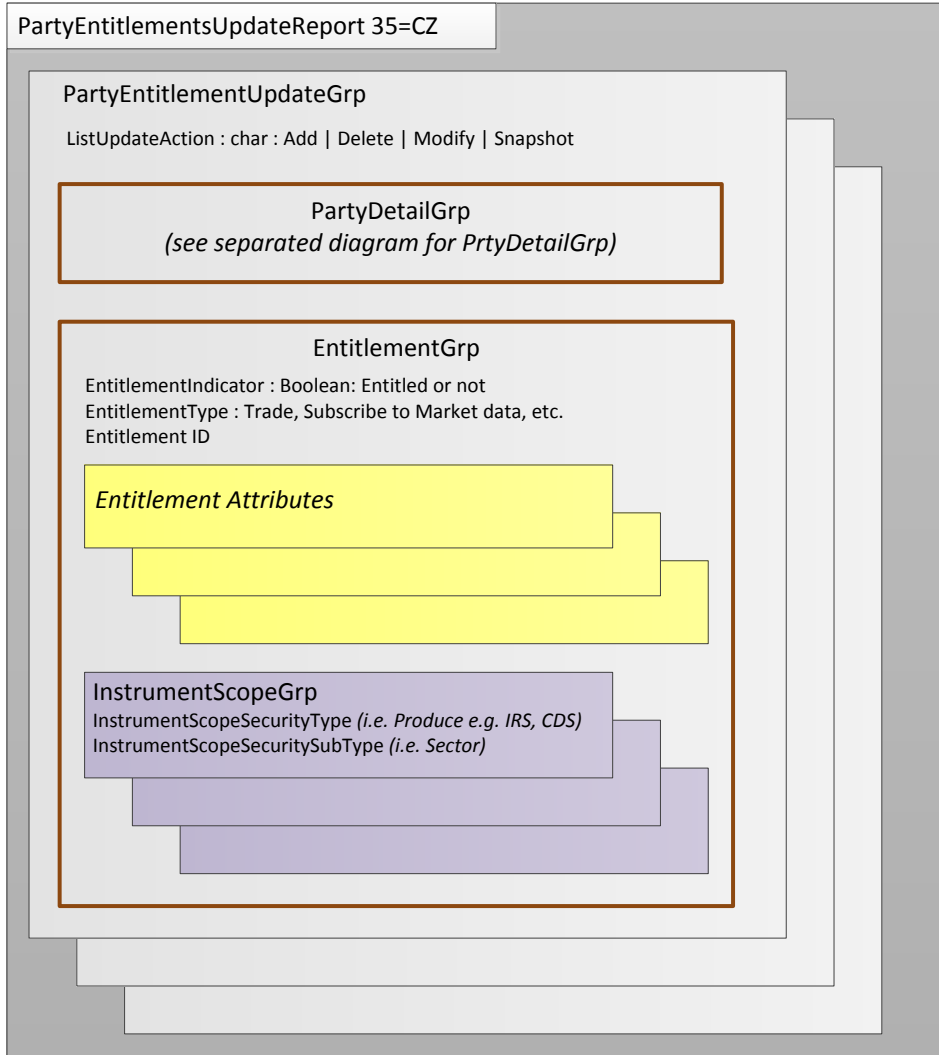


Figure 38: PartyEntitlementsUpdateReport (35=CZ) Message Structure

11.13.2 PartyEntitlementsUpdateReport (35=CZ) Message Definition

PartyEntitlementsUpdateReport (35=CZ) Execution Venue -> Dealer

The PartyEntitlementsUpdateReport(35=CZ) is used to convey incremental changes to party entitlements. It is similar to the PartyEntitlementsReport(35=CV). This message uses the PartyEntitlementsUpdateGrp component which includes the ability to specify an update action using ListUpdateAction(1324).

Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType=CZ	
	ApplicationSequenceControl	N	The ApplicationSequenceControl is used for application sequencing and recovery. Consisting of ApplSeqNum (1181), ApplID (1180), ApplLastSeqNum (1350), and ApplResendFlag (1352), FIX application messages that carries this component block will be able to use application level sequencing. ApplID, ApplSeqNum and ApplLastSeqNum fields identify the application id, application sequence number and the previous application sequence number (in case of intentional gaps) on each application message that carries this block.	
1771	EntitlementReportID	Y	Identifier for the PartyEntitlementsReport(35=CV).	
1770	EntitlementRequestID	N	Conditionally required when responding to PartyEntitlementsRequest(35=CU).	
1512	TotNoParties	N	Total number of PartyListGrp returned.	
893	LastFragment	N	Indicates whether this message is the last in a sequence of messages for those messages that support fragmentation, such as Allocation Instruction, Mass Quote, Security List, Derivative Security List	
	PartyEntitlementUpdateGrp	Y	Specifies the updated entitlements to be enforced for the given party(-ies) and related party(-ies).	
1772	NoPartyEntitlements	N	Number of party entitlement values.	
- >1324	ListUpdateAction	N	Required if NoPartyEntitlements(1772).	
->	PartyDetailGrp	N	Optional when ListUpdateAction(1324) = M(Modify) or D>Delete) and EntitlementRefID(1885) is provided.	
->- >1671	NoPartyDetails	N	Number of party details.	

->-> >169 <u>1</u>	<u>PartyDetailID</u>	N	The identification of the party. Required when NoPartyDetails(1671) > 0.	
->-> >169 <u>2</u>	<u>PartyDetailIDSource</u>	N	Used to identify source of PartyID value (e.g. BIC). Required when NoPartyDetails(1671) > 0.	
->-> >169 <u>3</u>	<u>PartyDetailRole</u>	N	Identifies the type of PartyID (e.g. Executing Broker). Required when NoPartyDetails(1671) > 0.	
->-> >167 <u>4</u>	<u>PartyDetailRoleQualifier</u>	N	Qualifies the value of PartyRole(452)	
- >188 <u>3</u>	<u>EntitlementStatus</u>	N	Status of entitlement definition for one party.	
->-> >	PartyDetailSubGrp	N	Additional party sub-identifiers	
->-> > >169 <u>4</u>	<u>NoPartyDetailSubIDs</u>	Y	Number of party detail sub-identifiers.	
->-> >>- >169 <u>5</u>	<u>PartyDetailSubID</u>	N	Required when NoPartyDetailSubIDs > 0.	
->-> >>- >169 <u>6</u>	<u>PartyDetailSubIDType</u>	N	Required when NoPartyDetailSubIDs > 0.	
->-> >	PartyDetailAltIDGrp	N	Optionally used to specify alternate IDs to identify the party specified.	
->	EntitlementGrp	N	Optional when ListUpdateAction(1324) = M(Modify) or D(Delete) and EntitlementRefID(1885) is provided.	
->- >177 <u>3</u>	<u>NoEntitlements</u>	N	Number of entitlement values.	
->-> >177 <u>4</u>	<u>EntitlementIndicator</u>	N	Required if NoEntitlements(1773) > 0.	
->-> >177 <u>5</u>	<u>EntitlementType</u>	N	Absence of this field indicates the meaning of the entitlement is implicit.	
->-> >	EntitlementAttribGrp	N	conveys a list of one or more attributes related to an entitlement. An entitlement may contain an EntitlementType, which states what can be done at a gross level, e.g. that a party can make markets. It may be limited further within EntitlementGrp, e.g. that	

			such market making is allowed only for a list of stocks. The EntitlementAttribGrp contains fine details clarifying or limiting the EntitlementType, e.g. that such market making must be conducted with a specific minimum quote size and a specific maximum spread.	
->-> >177 6	<u>EntitlementID</u>	N	Unique identifier for a specific NoEntitlements(1773) repeating group instance.	
->-> >178 4	<u>EntitlementPlatform</u>	N	The area to which the entitlement is applicable. This can be a trading platform or an offering.	
->-> >	InstrumentScopeGrp	N	Repeating group of InstrumentScope Components. Used to specify the instruments to which a request applies.	
->-> > >165 6	<u>NoInstrumentScopes</u>	N	Number of instrument scopes.	
->-> >>- >153 5	<u>InstrumentScopeOperator</u>	N	Required when NoInstrumentScopes > 0.	
->-> >>->	InstrumentScope	N	Used to specify the instrument	
->-> >>- > >153 6	<u>InstrumentScopeSymbol</u>	N	Used to limit instrument scope to specified symbol. See Symbol(55) field for description.	
->-> >>- > >153 7	<u>InstrumentScopeSymbolSfx</u>	N	Used to limit instrument scope to specified symbol suffix. See SymbolSfx(65) field for description.	
->-> >>- > >153 8	<u>InstrumentScopeSecurityID</u>	N	Used to limit instrument scope to specified security identifier. See SecurityID(48) field for description.	
->-> >>- > >153 9	<u>InstrumentScopeSecurityIDSource</u>	N	Used to limit instrument scope to specified security identifier source. See SecurityIDSource(22) field for description.	
->-> >>- >>	InstrumentScopeSecurityAltIDGrp	N	Alternative SecurityIDs for an instrument specified in the InstrumentScope.	
->-> >>-	<u>InstrumentScopeProduct</u>	N	Used to limit instrument scope to specified instrument product	

>- > <u>154</u> <u>3</u>			category. See Product (460) field for description.	
->-> >->- >- > <u>154</u> <u>4</u>	<u>InstrumentScopeProductComplex</u>	N	Used to limit instrument scope to specified product complex. See ProductComplex(1227) field for description.	
->-> >->- >- > <u>154</u> <u>5</u>	<u>InstrumentScopeSecurityGroup</u>	N	Used to limit instrument scope to specified security group. See SecurityGroup(1151) field for description.	
->-> >->- >- > <u>154</u> <u>6</u>	<u>InstrumentScopeCFIcode</u>	N	Used to limit instrument scope to specified CFIcode. See CFIcode(461) field for description.	
->-> >->- >- > <u>154</u> <u>7</u>	<u>InstrumentScopeSecurityType</u>	N	Used to limit instrument scope to specified security type. See SecurityType(167) field for description).	
->-> >->- >- > <u>154</u> <u>8</u>	<u>InstrumentScopeSecuritySubType</u>	N	Used to limit instrument scope to specified security sub-type. See SecuritySubType(762) field for description.	
->-> >->- >- > <u>154</u> <u>9</u>	<u>InstrumentScopeMaturityMonthYear</u>	N	Used to limit instrument scope to specified maturity month and year. See MaturityMonthYear(200) field for description.	
->-> >->- >- > <u>155</u> <u>0</u>	<u>InstrumentScopeMaturityTime</u>	N	Used to limit instrument scope to specified maturity time. See MaturityTime(1079) field for description.	
->-> >->- >- > <u>155</u> <u>1</u>	<u>InstrumentScopeRestructuringType</u>	N	Used to limit instrument scope to specified restructuring type. See RestructuringType(1449) field for description.	
->-> >->- >- > <u>155</u> <u>2</u>	<u>InstrumentScopeSeniority</u>	N	Used to limit instrument scope to specified seniority type. See Seniority(1450) field for description.	
->-> >->- >- > <u>155</u> <u>3</u>	<u>InstrumentScopePutOrCall</u>	N	Used to limit instrument scope to puts or calls. See PutOrCall(201) field for description.	
->->	<u>InstrumentScopeFlexibleIndicator</u>	N	Used to limit instrument scope to	

->->- >- >155 <u>4</u>			securities that can be defined using flexible terms or not. See FlexibleIndicator(1244) field for description.	
->->- >->- >- >155 <u>5</u>	<u>InstrumentScopeCouponRate</u>	Y	Used to limit instrument scope to specified coupon rate. See CouponRate(223) field for description.	
->->- >->- >- >161 <u>6</u>	<u>InstrumentScopeSecurityExchange</u>	N	Used to limit instrument scope to specified security exchange. See SecurityExchange(207) field for description.	
->->- >->- >- >155 <u>6</u>	<u>InstrumentScopeSecurityDesc</u>	N	Used to limit instrument scope to specified security description. See SecurityDesc(107) field for description.	
->->- >->- >- >162 <u>0</u>	<u>InstrumentScopeEncodedSecurityDescLen</u>	N	Byte length of encoded (non-ASCII characters) InstrumentScopeEncodedSecurityDesc (1621) field	
->->- >->- >- >162 <u>1</u>	<u>InstrumentScopeEncodedSecurityDesc</u>	N	Encoded (non-ASCII characters) representation of the InstrumentScopeSecurityDesc (1556) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the InstrumentScopeSecurityDesc field.	
->->- >->- >- >155 <u>7</u>	<u>InstrumentScopeSettlType</u>	N	Can be used to specify FX tenors.	
->->- >	MarketSegmentScopeGrp	N	conveys a list of markets and, optionally, their market segments. Note that the component MarketSegmentGrp exists, but is not useful for this purpose, as it conveys additional information not appropriate in this context.	
->->- >- >131 <u>0</u>	<u>NoMarketSegments</u>	N	Number of Market Segments on which a security may trade.	
->->- >->- >130 <u>1</u>	<u>MarketID</u>	N	Required if NoMarketSegments(1310) > 0.	

->->- >->- >130 0	<u>MarketSegmentID</u>	N	Identifies the market segment	
->->- >178 2	<u>EntitlementStartDate</u>	N	Indicates the starting date of the entitlement.	
->->- >178 3	<u>EntitlementEndDate</u>	N	Indicates the starting date of the entitlement.	
- >188 5	<u>EntitlementRefID</u>	N	Optional when PartyDetailGrp is provided or ListUpdateAction(1324) = A(Add).	
60	<u>TransactTime</u>	N	Timestamp when the business transaction represented by the message occurred.	
58	<u>Text</u>	N	Free format text string (Note: this field does not have a specified maximum length)	
354	<u>EncodedTextLen</u>	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.	
355	<u>EncodedText</u>	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.	
	StandardTrailer	Y	The standard FIX message trailer	

11.14 PartyEntitlementsDefinitionRequest (35=DA)

11.14.1 PartyEntitlementsDefinitionRequest (35=DA) Message Structure Diagram

The following diagram illustrates the structure of the PartyEntitlementsDefinitionRequest message:

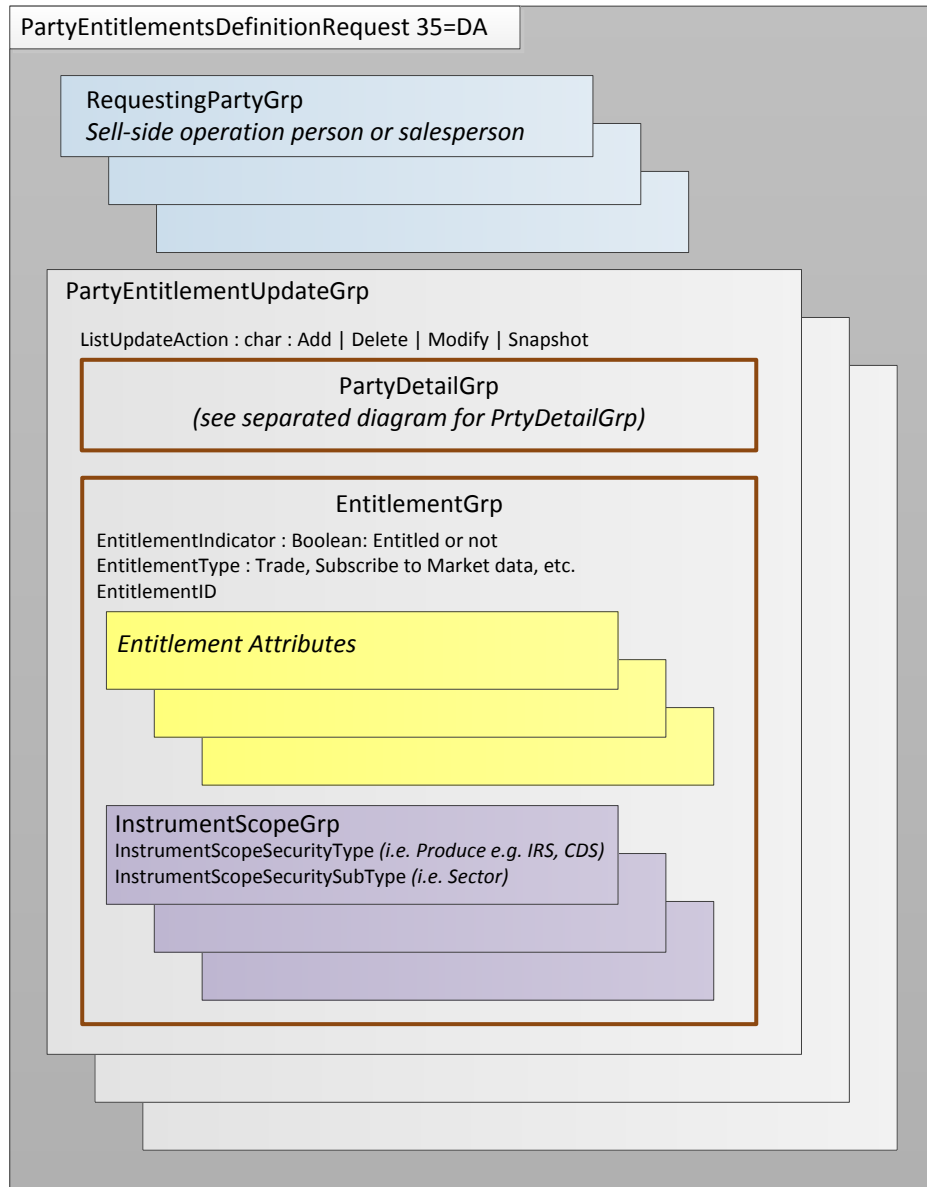


Figure 39: PartyEntitlementsDefinitionRequest (35=DA) Message Structure

11.14.2 Party Entitlements Definition Request Example

In this example the sell-side sends an entitlement definition request containing data for a single user to the Execution Venue.

The below table contains the business level attributes of the entitlement definition

Attribute	New buy side Firm
Buy-side firm Identifier	Hedge Fund-AB
Buy-side User ID	User-1
Buy-side user name	Jon A. Smith
Email	jon.a.smith@HedgeFund.com

LEI	LZ123
Sell-side Identifier	Bank-1
Bank Account Identifier Mapping [1]	FundAcc-1
Bank Account Identifier Mapping [2]	FundAcc-2
Instrument Type	FX Spot
Instrument currency pair	EUR/USD
Instrument Type	FX Forward
Instrument currency pair	EUR/USD
Instrument Type	FX Swap
Instrument currency pair	GBP/USD
Enablement Type	Trade
Enablement Status	Enabled
Modified By	Andy Smith
Account Identifier [1]	Account 1
Account Identifier [2]	Account 2
Market Data Stream	Level-1
Credit Limit	10000000
Credit Method	Net
Credit Currency	USD

Table 26: Example 1 – Sell-side sends an entitlement request report

11.14.2.1 PartyEntitlementDefinitionRequest FIXML

The following is the FIXML message tags and values that are expected in this example

```
<?xml version="1.0" encoding="UTF-8"?>
<FIXML xmlns="http://www.fixprotocol.org/FIXML-5-0-SP2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.fixprotocol.org/FIXML-5-0-SP2 ./Fixml/fixml-main-5-0-SP2.xsd">
  <PtyEntlmntDefReq RptID="1" TotNoPtys="1" >
    <ReqPty ID="Bank-1" Src="D" R="1">
      <Sub ID="Andy Smith" Typ="2"/>
    </ReqPty>
    <PtyEntlmntUpd ListUpdActn="A" >
      <!-- buy-side user attributes-->
      <PtyDetl ID="User-1" Src="D" R="3" Stat="1"> <!-- Buy-side User ID Identifier-->
        <Sub ID="Jon A. Smith" Typ="9"/> <!-- User name -->
        <Sub ID="Hedge Fund-AB" Typ="1"/> <!-- Firm Identifier -->
        <Sub ID="jon.a.smith@HedgeFund.com" Typ="8" /> <!-- Email -->
        <Sub ID="Account 1" Typ="19" /> <!-- Buy side account ID 1 -->
        <Sub ID="Account 2" Typ="19" /> <!-- Buy side account ID 2 -->
        <AltPty ID="LZ123" Src="N"/> <!-- LEI -->
        <ReltdPtyDetl ID="Bank-1" Src="D" R="1"> <!-- Sell-side attributes -->
          <Sub ID="FundAcc-1" Typ="19" /> <!-- Bank Account ID mapping 1-->
          <Sub ID="FundAcc-2" Typ="19" /> <!-- Bank Account ID mapping 2-->
        </ReltdPtyDetl>
      </PtyDetl>
      <Entlmnt Ind="Y" Typ="0" > <!-- Entitled to Trade -->
        <Attrib Typ="1500" Datatyp="5" Value="Level-1" /> <!--FIX tag MDStreamID(1500)-
-->
        <!-- Custom attribute: Credit limit -->
        <Attrib Typ="4000" Datatyp="1" Value="10000000" />
        <Attrib Typ="4001" Datatyp="5" Value="net" />
        <Attrib Typ="4002" Datatyp="16" Value="usd" />
        <!-- Instrument Scope-->
        <InstrmtScope Oper="1" Sym="eur/usd" Prod="4" SecTyp="FXSPOT"/>
        <InstrmtScope Oper="1" Sym="eur/usd" Prod="4" SecTyp="FXFWD"/>
        <InstrmtScope Oper="1" Sym="gbp/usd" Prod="4" SecTyp="FXSWAP"/>
      </Entlmnt>
    </PtyEntlmntUpd>
  </PtyEntlmntDefReq>
</FIXML>
```

11.14.3 PartyEntitlementsDefinitionRequest (35=DA) Message Definition

PartyEntitlementsDefinitionRequest (35=DA)		Dealer -> Execution Venue		
<i>The PartyEntitlementsDefinitionRequest(35=DA) is used for defining new entitlements, and modifying or deleting existing entitlements for the specified party(-ies).</i>				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType=DA	
1770	EntitlementRequestID	Y	Unique identifier for PartyEntitlementsRequest(35=CU).	
	RequestingPartyGrp	N	Can be used to identify the party making the request and their role.	
1657	NoRequestingPartyIDs	N	Number of requesting party identifiers.	
- > 1658	RequestingPartyID	N	Required when NoRequestingPartyIDs > 0.	
- > 1659	RequestingPartyIDSource	N	Required when NoRequestingPartyIDs > 0.	
- > 1660	RequestingPartyRole	N	Required when NoRequestingPartyIDs > 0.	
->	RequestingPartySubGrp	N	Sub identifiers for the requesting party.	
->- > 1661	NoRequestingPartySubIDs	N	Number of requesting party sub-identifiers.	
->->- > 1662	RequestingPartySubID	N	Required when NoRequestingPartySubIDs > 0.	
->->- > 1663	RequestingPartySubIDType	N	Required when NoRequestingPartySubIDs > 0.	
	PartyEntitlementUpdateGrp	Y	Specifies the entitlements to be defined, modified or deleted for the given party(-ies) and related party(-ies).	
1772	NoPartyEntitlements	N	Number of party entitlement values.	
- > 1324	ListUpdateAction	N	Required if NoPartyEntitlements(1772).	
->	PartyDetailGrp	N	Optional when ListUpdateAction(1324) = M(Modify) or D(Delete) and EntitlementRefID(1885) is provided.	
->- > 1671	NoPartyDetails	N	Number of party details.	
->->- > 1691	PartyDetailID	N	The identification of the party. Required when NoPartyDetails(1671) > 0.	

->->- >169 2	<u>PartyDetailIDSource</u>	N	Used to identify source of PartyID value (e.g. BIC). Required when NoPartyDetails(1671) > 0.	
->->- >169 3	<u>PartyDetailRole</u>	N	Identifies the type of PartyID (e.g. Executing Broker). Required when NoPartyDetails(1671) > 0.	
->->- >167 4	<u>PartyDetailRoleQualifier</u>	N	Qualifies the value of PartyRole(452)	
->->- >	PartyDetailSubGrp	N	Additional party sub-identifiers	
->->- >- >169 4	<u>NoPartyDetailSubIDs</u>	Y	Number of party detail sub-identifiers.	
->->- >>- >169 5	<u>PartyDetailSubID</u>	N	Required when NoPartyDetailSubIDs > 0.	
->->- >>- >169 6	<u>PartyDetailSubIDType</u>	N	Required when NoPartyDetailSubIDs > 0.	
->->- >	PartyDetailAltIDGrp	N	Optionally used to specify alternate IDs to identify the party specified.	
->	EntitlementGrp	N	Optional when ListUpdateAction(1324) = M(Modify) or D(Delete) and EntitlementRefID(1885) is provided.	
->- >177 3	<u>NoEntitlements</u>	N	Number of entitlement values.	
->->- >177 4	<u>EntitlementIndicator</u>	N	Required if NoEntitlements(1773) > 0.	
->->- >177 5	<u>EntitlementType</u>	N	Absence of this field indicates the meaning of the entitlement is implicit.	
->->- >	EntitlementAttribGrp	N	Conveys a list of one or more attributes related to an entitlement. An entitlement may contain an EntitlementType, which states what can be done at a gross level, e.g. that a party can make markets. It may be limited further within EntitlementGrp, e.g. that such market making is allowed only for a list of stocks. The EntitlementAttribGrp contains fine details clarifying or limiting the EntitlementType, e.g. that such market making must be conducted	

			with a specific minimum quote size and a specific maximum spread.	
->-> >177 6	<u>EntitlementID</u>	N	Unique identifier for a specific NoEntitlements(1773) repeating group instance.	
->-> >178 4	<u>EntitlementPlatform</u>	N	The area to which the entitlement is applicable. This can be a trading platform or an offering.	
->-> >	InstrumentScopeGrp	N	Repeating group of InstrumentScope Components. Used to specify the instruments to which a request applies.	
->-> > >165 6	<u>NoInstrumentScopes</u>	N	Number of instrument scopes.	
->-> >>- >153 5	<u>InstrumentScopeOperator</u>	N	Required when NoInstrumentScopes > 0.	
->-> >>->	InstrumentScope	N	Used to specify the instrument	
->-> >>- > >153 6	<u>InstrumentScopeSymbol</u>	N	Used to limit instrument scope to specified symbol. See Symbol(55) field for description.	
->-> >>- > >153 7	<u>InstrumentScopeSymbolSfx</u>	N	Used to limit instrument scope to specified symbol suffix. See SymbolSfx(65) field for description.	
->-> >>- > >153 8	<u>InstrumentScopeSecurityID</u>	N	Used to limit instrument scope to specified security identifier. See SecurityID(48) field for description.	
->-> >>- > >153 9	<u>InstrumentScopeSecurityIDSource</u>	N	Used to limit instrument scope to specified security identifier source. See SecurityIDSource(22) field for description.	
->-> >>- >>	InstrumentScopeSecurityAltIDGrp	N	Alternative SecurityIDs for an instrument specified in the InstrumentScope.	
->-> >>- > >154 3	<u>InstrumentScopeProduct</u>	N	Used to limit instrument scope to specified instrument product category. See Product (460) field for description.	
->-> >>- >	<u>InstrumentScopeProductComplex</u>	N	Used to limit instrument scope to specified product complex. See ProductComplex(1227) field for	

>154 4			description.	
->-> >->- >- >154 5	<u>InstrumentScopeSecurityGroup</u>	N	Used to limit instrument scope to specified security group. See SecurityGroup(1151) field for description.	
->-> >->- >- >154 6	<u>InstrumentScopeCFIcode</u>	N	Used to limit instrument scope to specified CFIcode. See CFIcode(461) field for description.	
->-> >->- >- >154 7	<u>InstrumentScopeSecurityType</u>	N	Used to limit instrument scope to specified security type. See SecurityType(167) field for description).	
->-> >->- >- >154 8	<u>InstrumentScopeSecuritySubType</u>	N	Used to limit instrument scope to specified security sub-type. See SecuritySubType(762) field for description.	
->-> >->- >- >154 9	<u>InstrumentScopeMaturityMonthYear</u>	N	Used to limit instrument scope to specified maturity month and year. See MaturityMonthYear(200) field for description.	
->-> >->- >- >155 0	<u>InstrumentScopeMaturityTime</u>	N	Used to limit instrument scope to specified maturity time. See MaturityTime(1079) field for description.	
->-> >->- >- >155 1	<u>InstrumentScopeRestructuringType</u>	N	Used to limit instrument scope to specified restructuring type. See RestructuringType(1449) field for description.	
->-> >->- >- >155 2	<u>InstrumentScopeSeniority</u>	N	Used to limit instrument scope to specified seniority type. See Seniority(1450) field for description.	
->-> >->- >- >155 3	<u>InstrumentScopePutOrCall</u>	N	Used to limit instrument scope to puts or calls. See PutOrCall(201) field for description.	
->-> >->- >- >155 4	<u>InstrumentScopeFlexibleIndicator</u>	N	Used to limit instrument scope to securities that can be defined using flexible terms or not. See FlexibleIndicator(1244) field for description.	
->-> >->-	<u>InstrumentScopeCouponRate</u>	Y	Used to limit instrument scope to specified coupon rate. See	

>- > <u>155</u> <u>5</u>			CouponRate(223) field for description.	
->->- >->- >- > <u>161</u> <u>6</u>	<u>InstrumentScopeSecurityExchange</u>	N	Used to limit instrument scope to specified security exchange. See SecurityExchange(207) field for description.	
->->- >->- >- > <u>155</u> <u>6</u>	<u>InstrumentScopeSecurityDesc</u>	N	Used to limit instrument scope to specified security description. See SecurityDesc(107) field for description.	
->->- >->- >- > <u>162</u> <u>0</u>	<u>InstrumentScopeEncodedSecurityDescLen</u>	N	Byte length of encoded (non-ASCII characters) InstrumentScopeEncodedSecurityDesc (1621) field	
->->- >->- >- > <u>162</u> <u>1</u>	<u>InstrumentScopeEncodedSecurityDesc</u>	N	Encoded (non-ASCII characters) representation of the InstrumentScopeSecurityDesc (1556) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the InstrumentScopeSecurityDesc field.	
->->- >->- >- > <u>155</u> <u>7</u>	<u>InstrumentScopeSettlType</u>	N	Can be used to specify FX tenors.	
->->- >	MarketSegmentScopeGrp	N	conveys a list of markets and, optionally, their market segments. Note that the component MarketSegmentGrp exists, but is not useful for this purpose, as it conveys additional information not appropriate in this context.	
->->- >- > <u>131</u> <u>0</u>	<u>NoMarketSegments</u>	N	Number of Market Segments on which a security may trade.	
->->- >->- > <u>130</u> <u>1</u>	<u>MarketID</u>	N	Required if NoMarketSegments(1310) > 0.	
->->- >->- > <u>130</u> <u>0</u>	<u>MarketSegmentID</u>	N	Identifies the market segment	
->->- > <u>178</u>	<u>EntitlementStartDate</u>	N	Indicates the starting date of the entitlement.	

<u>2</u>				
->->- > <u>178</u> <u>3</u>	<u>EntitlementEndDate</u>	N	Indicates the starting date of the entitlement.	
- > <u>188</u> <u>5</u>	<u>EntitlementRefID</u>	N	Optional when PartyDetailGrp is provided or ListUpdateAction(1324) = A(Add).	
<u>58</u>	<u>Text</u>	N	Free format text string (Note: this field does not have a specified maximum length)	
<u>354</u>	<u>EncodedTextLen</u>	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.	
<u>355</u>	<u>EncodedText</u>	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.	
	StandardTrailer	Y	The standard FIX message trailer	

11.15 PartyEntitlementsDefinitionRequestAck (35=DB)

11.15.1 PartyEntitlementsDefinitionRequestAck (35=DB) Message Structure Diagram

The following diagram illustrates the structure of the PartyEntitlementsDefinitionRequestAck message:

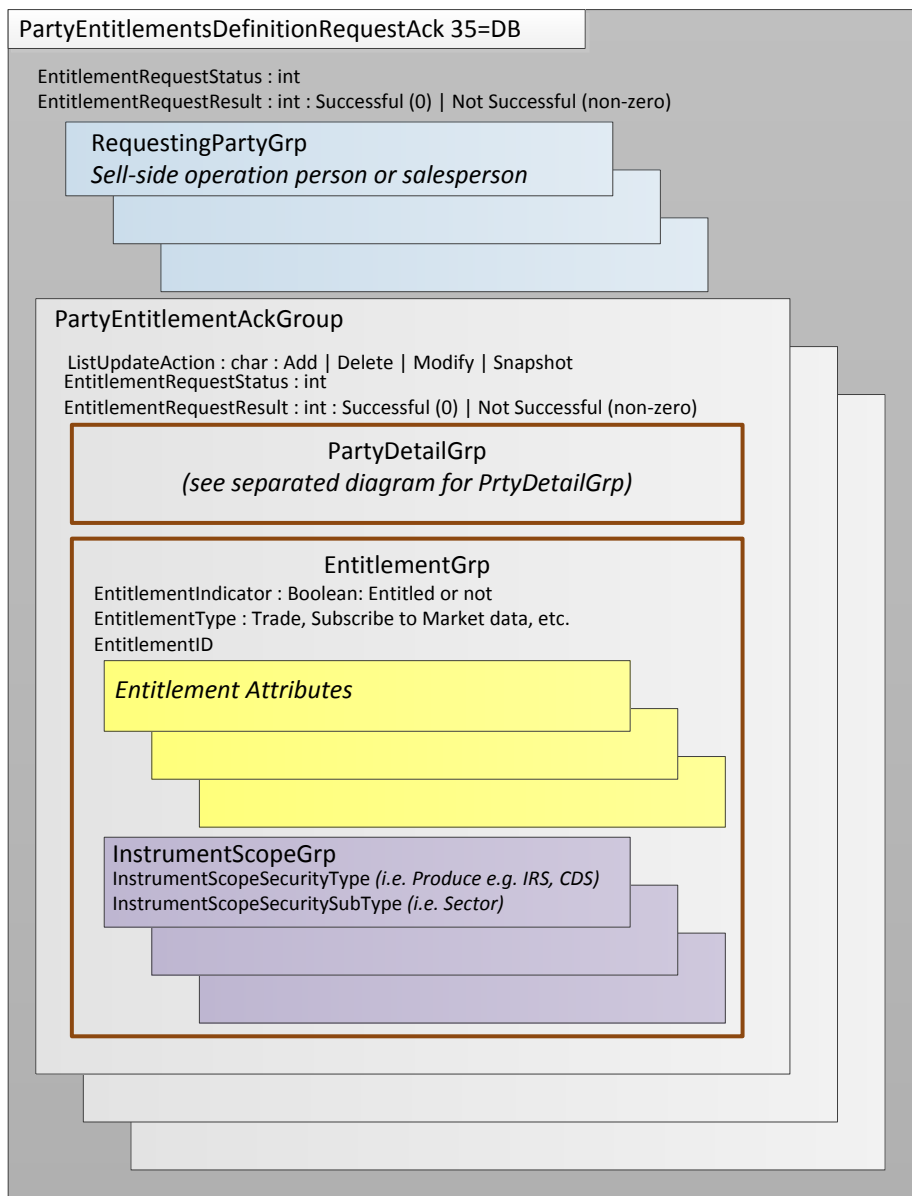


Figure 40: PartyEntitlementsDefinitionRequestAck (35=DB) Message Structure

11.15.2 PartyEntitlementsDefinitionRequestAck(35=DB) Message Definition

PartyEntitlementsDefinitionRequestAck (35=DB)		Execution Venue -> Dealer		
The PartyEntitlementsDefinitionRequestAck(35=DB) is used as a response to the PartyEntitlementsDefinitionRequest(35=DA) to accept (with or without changes) or reject the definition of party entitlements.				
Tag	FieldName	Req'd	Description	Comment
	StandardHeader	Y	MsgType=DB	
1770	EntitlementRequestID	Y	Unique identifier for PartyEntitlementsRequest(35=CU).	

1882	EntitlementRequestStatus	Y	Status of party entitlements definition request.
1881	EntitlementRequestResult	N	Result of risk limit definition request.
	RequestingPartyGrp	N	Identifies the party making the request.
1657	NoRequestingPartyIDs	N	Number of requesting party identifiers.
- >165 8	RequestingPartyID	N	Required when NoRequestingPartyIDs > 0.
- >165 9	RequestingPartyIDSource	N	Required when NoRequestingPartyIDs > 0.
- >166 0	RequestingPartyRole	N	Required when NoRequestingPartyIDs > 0.
->	RequestingPartySubGrp	N	Sub identifiers for the requesting party.
->- >166 1	NoRequestingPartySubIDs	N	Number of requesting party sub-identifiers.
->-> >166 2	RequestingPartySubID	N	Required when NoRequestingPartySubIDs > 0.
->-> >166 3	RequestingPartySubIDType	N	Required when NoRequestingPartySubIDs > 0.
	PartyEntitlementAckGrp	N	The PartyEntitlementAckGrp component is used in the PartyEntitlementsDefinitionRequestAck (35=DB) message to provide the status of each action (add, modify or delete) requested by the PartyEntitlementsDefinitionRequest(35=DA) message.
1772	NoPartyEntitlements	N	Number of party entitlement values.
- >132 4	ListUpdateAction	N	Required if NoPartyEntitlements(1772).
- >188 3	EntitlementStatus	N	Required if NoPartyEntitlements(1772).
- >188 4	EntitlementResult	N	Result of entitlement definition for one party.
- >132 8	RejectText	N	Identifies the reason for rejection.
- >166 4	EncodedRejectTextLen	N	Byte length of encoded (non-ASCII characters) EncodedRejectText(1665) field.
- >166 5	EncodedRejectText	N	Encoded (non-ASCII characters) representation of the RejectText(1328) field in the encoded format specified via the MessageEncoding(347) field. If used, the ASCII (English) representation

			should also be specified in the RejectText(1328) field.	
->	PartyDetailGrp	N	Optional when ListUpdateAction(1324) = M(Modify) or D(Delete) and EntitlementRefID(1885) is provided.	
->- >167 <u>1</u>	NoPartyDetails	N	Number of party details.	
->->- >169 <u>1</u>	PartyDetailID	N	The identification of the party. Required when NoPartyDetails(1671) > 0.	
->->- >169 <u>2</u>	PartyDetailIDSource	N	Used to identify source of PartyID value (e.g. BIC). Required when NoPartyDetails(1671) > 0.	
->->- >169 <u>3</u>	PartyDetailRole	N	Identifies the type of PartyID (e.g. Executing Broker). Required when NoPartyDetails(1671) > 0.	
->->- >167 <u>4</u>	PartyDetailRoleQualifier	N	Qualifies the value of PartyRole(452)	
->->- >	PartyDetailSubGrp	N	Additional party sub-identifiers	
->->- >- >169 <u>4</u>	NoPartyDetailSubIDs	Y	Number of party detail sub-identifiers.	
->->- >->- >169 <u>5</u>	PartyDetailSubID	N	Required when NoPartyDetailSubIDs > 0.	
->->- >->- >169 <u>6</u>	PartyDetailSubIDType	N	Required when NoPartyDetailSubIDs > 0.	
->->- >	PartyDetailAltIDGrp	N	Optionally used to specify alternate IDs to identify the party specified.	
->	EntitlementGrp	N	Optional when ListUpdateAction(1324) = M(Modify) or D(Delete) and EntitlementRefID(1885) is provided.	
->- >177 <u>3</u>	NoEntitlements	N	Number of entitlement values.	
->->- >177 <u>4</u>	EntitlementIndicator	N	Required if NoEntitlements(1773) > 0.	
->->- >177 <u>5</u>	EntitlementType	N	Absence of this field indicates the meaning of the entitlement is implicit.	
->->- >	EntitlementAttribGrp	N	conveys a list of one or more attributes related to an entitlement. An entitlement may contain an EntitlementType, which states what can be done at a gross level, e.g. that a party can make markets. It may be	

			limited further within EntitlementGrp, e.g. that such market making is allowed only for a list of stocks. The EntitlementAttribGrp contains fine details clarifying or limiting the EntitlementType, e.g. that such market making must be conducted with a specific minimum quote size and a specific maximum spread.	
->-> >177 6	<u>EntitlementID</u>	N	Unique identifier for a specific NoEntitlements(1773) repeating group instance.	
->-> >178 4	<u>EntitlementPlatform</u>	N	The area to which the entitlement is applicable. This can be a trading platform or an offering.	
->-> >	InstrumentScopeGrp	N	Repeating group of InstrumentScope Components. Used to specify the instruments to which a request applies.	
->-> > >165 6	<u>NoInstrumentScopes</u>	N	Number of instrument scopes.	
->-> >>- >153 5	<u>InstrumentScopeOperator</u>	N	Required when NoInstrumentScopes > 0.	
->-> >>->	InstrumentScope	N	Used to specify the instrument	
->-> >>- > >153 6	<u>InstrumentScopeSymbol</u>	N	Used to limit instrument scope to specified symbol. See Symbol(55) field for description.	
->-> >>- > >153 7	<u>InstrumentScopeSymbolSfx</u>	N	Used to limit instrument scope to specified symbol suffix. See SymbolSfx(65) field for description.	
->-> >>- > >153 8	<u>InstrumentScopeSecurityID</u>	N	Used to limit instrument scope to specified security identifier. See SecurityID(48) field for description.	
->-> >>- > >153 9	<u>InstrumentScopeSecurityIDSourc</u> e	N	Used to limit instrument scope to specified security identifier source. See SecurityIDSource(22) field for description.	
->-> >>- >>	InstrumentScopeSecurityAltIDGr p	N	Alternative SecurityIDs for an instrument specified in the InstrumentScope.	
->-> >>- >	<u>InstrumentScopeProduct</u>	N	Used to limit instrument scope to specified instrument product category. See Product (460) field for description.	

>154 3				
->-> >>- >- >154 4	<u>InstrumentScopeProductComple</u> x	N	Used to limit instrument scope to specified product complex. See ProductComplex(1227) field for description.	
->-> >>- >- >154 5	<u>InstrumentScopeSecurityGroup</u>	N	Used to limit instrument scope to specified security group. See SecurityGroup(1151) field for description.	
->-> >>- >- >154 6	<u>InstrumentScopeCFICode</u>	N	Used to limit instrument scope to specified CFICode. See CFICode(461) field for description.	
->-> >>- >- >154 7	<u>InstrumentScopeSecurityType</u>	N	Used to limit instrument scope to specified security type. See SecurityType(167) field for description).	
->-> >>- >- >154 8	<u>InstrumentScopeSecuritySubTyp</u> e	N	Used to limit instrument scope to specified security sub-type. See SecuritySubType(762) field for description.	
->-> >>- >- >154 9	<u>InstrumentScopeMaturityMonth</u> Year	N	Used to limit instrument scope to specified maturity month and year. See MaturityMonthYear(200) field for description.	
->-> >>- >- >155 0	<u>InstrumentScopeMaturityTime</u>	N	Used to limit instrument scope to specified maturity time. See MaturityTime(1079) field for description.	
->-> >>- >- >155 1	<u>InstrumentScopeRestructuringTy</u> pe	N	Used to limit instrument scope to specified restructuring type. See RestructuringType(1449) field for description.	
->-> >>- >- >155 2	<u>InstrumentScopeSeniority</u>	N	Used to limit instrument scope to specified seniority type. See Seniority(1450) field for description.	
->-> >>- >- >155 3	<u>InstrumentScopePutOrCall</u>	N	Used to limit instrument scope to puts or calls. See PutOrCall(201) field for description.	
->-> >>-	<u>InstrumentScopeFlexibleIndicato</u> r	N	Used to limit instrument scope to securities that can be defined using	

>- >155 <u>4</u>			flexible terms or not. See FlexibleIndicator(1244) field for description.	
->-> >>- >- >155 <u>5</u>	<u>InstrumentScopeCouponRate</u>	Y	Used to limit instrument scope to specified coupon rate. See CouponRate(223) field for description.	
->-> >>- >- >161 <u>6</u>	<u>InstrumentScopeSecurityExchang</u> <u>e</u>	N	Used to limit instrument scope to specified security exchange. See SecurityExchange(207) field for description.	
->-> >>- >- >155 <u>6</u>	<u>InstrumentScopeSecurityDesc</u>	N	Used to limit instrument scope to specified security description. See SecurityDesc(107) field for description.	
->-> >>- >- >162 <u>0</u>	<u>InstrumentScopeEncodedSecurity</u> <u>DescLen</u>	N	Byte length of encoded (non-ASCII characters) InstrumentScopeEncodedSecurityDesc (1621) field	
->-> >>- >- >162 <u>1</u>	<u>InstrumentScopeEncodedSecurity</u> <u>Desc</u>	N	Encoded (non-ASCII characters) representation of the InstrumentScopeSecurityDesc (1556) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the InstrumentScopeSecurityDesc field.	
->-> >>- >- >155 <u>7</u>	<u>InstrumentScopeSettlType</u>	N	Can be used to specify FX tenors.	
->-> >- >131 <u>0</u>	MarketSegmentScopeGrp	N	conveys a list of markets and, optionally, their market segments. Note that the component MarketSegmentGrp exists, but is not useful for this purpose, as it conveys additional information not appropriate in this context.	
->-> >- >131 <u>0</u>	<u>NoMarketSegments</u>	N	Number of Market Segments on which a security may trade.	
->-> >>- >130 <u>1</u>	<u>MarketID</u>	N	Required if NoMarketSegments(1310) > 0.	
->-> >>- >130	<u>MarketSegmentID</u>	N	Identifies the market segment	

<u>0</u>				
->->- > <u>178</u> <u>2</u>	<u>EntitlementStartDate</u>	N	Indicates the starting date of the entitlement.	
->->- > <u>178</u> <u>3</u>	<u>EntitlementEndDate</u>	N	Indicates the starting date of the entitlement.	
- > <u>188</u> <u>5</u>	<u>EntitlementRefID</u>	N	Optional when PartyDetailGrp is provided or ListUpdateAction(1324) = A(Add).	
<u>58</u>	<u>Text</u>	N	Free format text string (Note: this field does not have a specified maximum length)	
<u>354</u>	<u>EncodedTextLen</u>	N	Byte length of encoded (non-ASCII characters) EncodedText (355) field.	
<u>355</u>	<u>EncodedText</u>	N	Encoded (non-ASCII characters) representation of the Text (58) field in the encoded format specified via the MessageEncoding (347) field. If used, the ASCII (English) representation should also be specified in the Text field.	
	StandardTrailer	Y	The standard FIX message trailer	

Appendix I – Entitlement Attributes

The following table provides common custom entitlement attributes that can be used in the EntitlementAttribGrp for the scope of FX OTC client enablement in implementations. Additional entitlement attributes may be bilaterally agreed between Execution Venue and Sell-Side and should be allocated EntitlementAttribType values from 5000 onwards. It should be noted that a similar list of attributes exists for Fixed Income securities and can be found in the document: *Recommended Practices for Party Entitlement, OTC Markets: Fixed income instruments.*

Entitlement Attribute Types for FX OTC D2C Markets			
Entitlement AttribType (1778)	Attribute Name	EntitlementAttrib Datatype(1779)	Description
4050	Credit Method	Enumerated String	For trade entitlement it is common to attach credit limits for the aggregated traded quantities. These can be calculated in different methods. Values may include: <ul style="list-style-type: none"> • Net • Absolute
4051	Credit Limit	Integer	Credit Limit for specific entitlement (See Credit Method) Note that the currency may be in EntitlementAttribCurrency(1781)
4052	STP responsibility	String	Indicator who will perform STP for specific entitlement. Values may include: <ul style="list-style-type: none"> • Venue • Sell-side • <Third party name>
4053	Trade limit single order	Integer	Limit quantity for single order
4054	Pricing Route	String	Values may include: <ul style="list-style-type: none"> • Auto Pricing • Manual Pricing
4055	Giveup By	String	Name of firm who will do the give ups
4056	Benchmark	String	Benchmark name which is part of the entitlement attributes
4057	Strategy Name	String	Algo strategies or transaction strategies entitled
4058	Subject to regulation	String	Values may include: <ul style="list-style-type: none"> • Dodd Frank • MIFID • MIFIR
4059	Delivery Location	String	Entitled delivery location (city or country code) of the contract, instrument, or settlement.
4060	Onshore Delivery	Boolean	Specific to FX Instrument which is delivered onshore (i.e. inside the issuing country) vs. delivered offshore (i.e. outside the issuing country)

Appendix II – Document Figures

Figure 1: D2C Venues - Entitlement Information Flow 13
 Figure 2: Prime Broker Venues - Entitlement Information Flow 14
 Figure 3: D2D venues..... 14
 Figure 4: Complex venues combining buy-side, Prime Broker and Liquidity Provider 15
 Figure 5: Entitlement Status transition..... 19
 Figure 6: Overview Diagram 20
 Figure 7: Activity Diagram..... 21
 Figure 8: Entitlements Data Model Diagram 22
 Figure 9: Scenario WF1: New buy-side request – accepted 25
 Figure 10: Scenario WF2: New buy-side request – rejected or deferred 27
 Figure 11: Scenario WF3: Sell-side setup a new buy-side..... 29
 Figure 12: Scenario WF4: Amendment request (incl. new fund/account) - accepted 30
 Figure 13: Scenario WF5: Amendment request (incl. new fund/account) – rejected or deferred..... 32
 Figure 14: Scenario WF6: Amendment by sell-side - accepted 34
 Figure 15: Scenario WF7: Amendment by sell-side – partially accepted and partially rejected 35
 Figure 16: Scenario WF8: Amendment by sell-side – rejected 38
 Figure 17: Scenario WF9: Reports snapshot..... 39
 Figure 18: Scenario WF10: Reports update 41
 Figure 19: Scenario WF11: Suspension of client by sell-side 43
 Figure 20 Scenario WF12: Off-board a client by request..... 44
 Figure 21 Scenario WF13: Off-board or Suspend a Firm, User or Account 45
 Figure 22: Scenario SA.1: Buy-side client requests to be assigned to a Pricing Stream – sell-side assigns 56
 Figure 23: Scenario SA.2: Buy-side client requests to be assigned to a Pricing Stream – sell-side rejects..... 58
 Figure 24: Scenario SA.3: Sell-side assigns a buy-side client to a Pricing Stream 60
 Figure 25: Scenario SA.4: Sell-side terminates (unassigns) a Pricing Stream from the buy-side client 61
 Figure 26: Scenario WF.M.20: A Sell-Side User Login 63
 Figure 27: Scenario WF.M.21: Sell-Side User Logoff..... 64
 Figure 28: Scenario WF.M.22: Sell-Side Requests Instrument types and sub-types 65
 Figure 29: Scenario WF.M.23: Sell-Side Requests a Snapshot of Buy-Side Entities/Users 66
 Figure 30: Scenario WF.M.24: Sell-Side Requests Snapshot + Updates of Buy-Side Entities/Users..... 68
 Figure 31: Scenario WF.M.25: Sell side sends a message to the buy-side 70
 Figure 32: PartyDetailsGrp component Structure 80
 Figure 33: PartyDetailsListRequests(35=CF) Message Structure 81
 Figure 34: PartyDetailsListReport (35=CG) Message Structure 83
 Figure 35: PartyDetailsListUpdateReport (35=CK) Message Structure..... 87
 Figure 36: PartyEntitlementsRequest (35=CU) Message Structure 91
 Figure 37: PartyEntitlementsReport (35=CV) Message Structure 97
 Figure 38: PartyEntitlementsUpdateReport (35=CZ) Message Structure..... 106
 Figure 39: PartyEntitlementsDefinitionRequest (35=DA) Message Structure..... 113
 Figure 40: PartyEntitlementsDefinitionRequestAck (35=DB) Message Structure 121

Appendix III – Document Tables

Table 1: EntitlementSubType(2402) proposed enumeration	18
Table 2: Scenario WF1: New buy-side request – accepted	26
Table 3: Scenario WF2: New buy-side request – Rejected or deferred	28
Table 4: Scenario WF3: Sell-side setup a new buy-side	29
Table 5: Scenario WF4: Amendment request (incl. new fund/account) – accepted	31
Table 6: Scenario WF5: Amendment request (incl. new fund/account) – rejected or deferred	33
Table 7: Scenario WF6: Amendment by sell-side – accepted	34
Table 8: Scenario WF7: Amendment by sell-side – partially accepted and partially rejected	37
Table 9: Scenario WF8: Amendment by sell-side – rejected	38
Table 10: Scenario WF9: Reports snapshot	39
Table 11: Scenario WF10: Reports update	42
Table 12: Scenario WF11: Suspension of client by sell-side	43
Table 13: Scenario WF12: Off-board a client by request	44
Table 14: Scenario WF13: Off-board or Suspend a Firm, User or Account	46
Table 15: Scenario SA.1: Buy-side client requests to be assigned to a Pricing Stream – sell-side assigns	57
Table 16: Scenario SA.2: Buy-side client requests to be assigned to a Pricing Stream – sell-side rejects	59
Table 17: Scenario SA.2: Sell-side assigns a buy-side client to a Pricing Stream	60
Table 18: Scenario SA.4: Sell-side terminates (unassigns) a Pricing Stream from the buy-side client	61
Table 19: Scenario WF.M.20: A Sell-Side User Login	63
Table 20: Scenario WF.M.21: Sell-Side User Logoff	64
Table 21: Scenario WF.M.22: Sell-Side Requests Instrument types and sub-types	65
Table 22: Scenario WF.M.23: Sell-Side Requests a Snapshot of Buy-Side Entities/Users	66
Table 23: Scenario WF.M.24: Sell-Side Requests Snapshot + Updates of Buy-Side Entities/Users	69
Table 24: Scenario WF.M.25: Sell side sends a message to the buy-side	70
Table 25: Example 1 – Execution Venue sends an entitlement report	98
Table 26: Example 1 – Sell-side sends an entitlement request report	114