Post-Trade Processing via FIX Recommended Practices - Futures

February 8, 2016
Revision 0.12
Proposal Status: Release Candidate 1

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 FIX TRADING COMMUNITY GLOBAL TECHNICAL COMMITTEE WILL NOT ALLOW EARLY IMPLEMENTATION TO CONSTRAIN ITS ABILITY TO MAKE CHANGES TO THIS SPECIFICATION PRIOR TO FINAL RELEASE. IT IS INAPPROPRIATE TO USE FIX TRADING COMMUNITY WORKING DRAFTS AS REFERENCE MATERIAL OR TO CITE THEM AS OTHER THAN “WORKS IN PROGRESS”. THE FIX TRADING COMMUNITY GLOBAL TECHNICAL COMMITTEE WILL ISSUE, UPON COMPLETION OF REVIEW AND RATIFICATION, AN OFFICIAL STATUS ("APPROVED") FOR 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).

Table of Contents

Document History

Date	Editor(s)	Description	Revision
6/25/2014	D. Tolman	- first draft	v 0.01
8/14/2014	D. Tolman	- comments from working group review	V0.02
2/26/2015	D. Tolman	- updated give-up/take-up (step-out/step-in) workflow.	V0.03
4/9/2015	D. Tolman	- clarified prices to be returned in “major currency” units. - added commissions and fees draft	V0.04
5/27/2015	D.Tolman	- added CustOrderHandlingInst (1031) to be a rate identifier rather than including actual commission values	V0.05
9/25/2015	D. John	- added Asset Manager best fit process	V0.06
10/1/2015	D. John	- removed TAG 626 = 2 and updated graphic	V0.07
10/3/2015	D. Tolman	- added repeating group for multiple commissions and removed the rate identifier concept - described the workflow for optional inclusion of commissions and fees in the primary workflow and transition to including commissions and fees.	V0.08
10/23/2015	D. Tolman	- added back execution-method - identified all workflows for sell-side and buy-side specification of prices.	V0.10
1/19/2016	D. Tolman	- added tag numbers for commission component	V0.11
2/8/2016	D. Tolman	- Release Candidate 1 - identified all FIX 5.0 or later tags and valid values - general cleanup	V0.12

1 Preface

The purpose of the FIX Trading Community Post-Trade Processing via FIX Initiative is to define industry practices for common usage of the FIX Protocol for post-trade processing, for all asset classes, between buy-sides and sell-sides that can be used bi-laterally as well as through intermediary facilities.

This document is one of a series of Recommended Practices for Post-Trade Processing via FIX specifying guidelines for industry usage of the FIX standard to facilitate parallel implementation across buy-sides, sell-sides and intermediaries.

This document assumes an understanding of the FIX Protocol and post-trade processing in general. This document is written in the context of the Common Post-trade Framework.

2 Scope

This document addresses the post-trade workflow for futures and options on futures via FIX.

Note: While the base protocol is FIX 4.4, additional tags or additional valid values from FIX 5.0 or later have been added as needed to meet industry post-trade processing requirements. These are identified in the message format tables ("[FIX 5.0 or later]") and may require specific exception configuration for FIX engines. The FIX Global Technical Committee has approved this as accepted practice.

3 References

Currently located on the FPL Website at: [http://www.fixtradingcommunity.org/pg/structure/fix-guidelines/best-practiceguidelines-directory]
Under the "**Post Trade**" heading:
General

- Post-Trade Processing via FIX Recommended Practices - Common Framework
- Equity Allocations Via FIX - Recommended Guidelines V1.2.4

4 Glossary

Term	Description
Placements, Orders, Trades, Transactions	• **Placement vs Order**: the buy-side makes a "placement" for all or part of an "order" • **Trade vs Transaction**: a trade is a result of a placement and is before allocation to an account. A transaction is the result of an allocation including client, account, quantity, price, capacity.
Account Types	• **Execution Account**: the account in which trades are held by the execution broker until they are allocated to client clearing accounts. • **Clearing Account**: a client account into which trades are allocated for clearing. • **TOP/Suspense Account**: an internal account used to temporarily hold trades and/or pass trades between brokers.
Prices	• Fair Average Price: average price calculated by dividing the total price of the executions to be allocated by the total quantity of executions to be allocated. There may be a residual value. • Booking Price: the exact price used for a specific allocation of trade executions to a clearing account. This may be may be a rounded average price, with possible residual value, if trades with different prices were allocated into this account. • Price Factor: if the instrument trades on the exchange in currency units (exchange-price format) other than the standard units for the currency (e.g. USD vs. USD) a price factor is used to convert to and from the units (e.g. .01 in the case of USD to/from USD). • Near-Average-Price, Best-fit-average-price: allocating a set of trades at the individual trade prices to a given account to achieve a net result close to the computed fair-average-price. • Exchange Traded Units: some instruments are traded in units other than the standard (e.g. USD vs USD). • Major Currency Units: the standard units for the currency (e.g. USD rather than USD).

Term	Description
Roles and Activities	• Execution: the broker to whom the FIX Order message is sent and who executes the trade. • Clearing: the broker who clears trades for one or more client clearing accounts. • Allocation: the broker to whom the FIX Allocation Instruction message is sent and who is responsible for average price calculation and partitioning of the trade executions across the specified clearing accounts and clearing brokers to achieve booking prices for each account within tolerance relative to the Fair Average Price. • Client Clearing Account Give-up: if instructed the execution broker will transfer ("give-up") a trade allocation to a specified client-account to another broker for clearing. There are two cases that must be addressed – one where the exchange and involved brokers support passing of an average booking price and the second case where they don't and some other process must be employed to achieve a fair average price allocation: • Average-price protocol: • The allocation broker is able to pass an average price for the total quantity allocated of a given account to the give-up clearing broker and the clearing broker will use the average price provided as the booking price. Clearing can be validated by an exact comparison of booking price sent by the allocation broker and the price booked by the clearing broker. • Exact-price protocol: • The allocation broker is not able to pass an average price, with residual, to the clearing broker so some other process must be employed to assure that buy-side can validate that the account in question receives a fair price allocation and that the booking is completed by the clearing broker. • Special Allocation Restrictions: in some cases there may be special allocation restrictions (e.g. Australasian restriction on use of average-priced allocations).

Term	Description
Validation	Allocation Validation: Validate that: • Allocation quantity instructions have been followed. • Any averaged booking prices for a clearing account are within an acceptable tolerance relative to the Fair Average Price. • Clearing Validation: validate that the booking instructions (quantity and price) in the allocation report were followed by the clearing broker.

5 Assumptions

Commissions for futures are paid outside the post-trade workflow.

6 Open Issues and Opportunities

Give-up clearing firm id-based matching: currently there is no mechanism to pass an identifier through the exchange give-up workflow that could be included in the step-in AllocationInstruction(35=J) to the give-up clearing firm. As a result the clearing firm must use economic matching rather than id-based matching.

7 Key Concepts/Processes/Notes

7.1 Instrument Identification

There are two options for instrument identification: RIC codes and Bloomberg YellowKeys.

If other symbology is required, it can used it without any other changes to the spec.

[NOTE: *To re-visited during public review period: Need to asses constraints on using/recommending proprietary symbologies.]*

7.1.1 Futures Instruments

7.1.1.1 Single Leg

1. Symbol(55) = <future-symbol>

2. CFIcode(461) = “FXXXXX”, (future, single)
3. SecurityType(167) = “FUT”
4. SecuritySourceID(22)
    - <root-code><month-code><year-last-digit><space><BBYellowKeyCode> (e.g. “GCG0 Comdty ”)
        - <root>
            - 2 or 3 character code
        - <month-code>

Month	Month Code
January	F
February	G
March	H
April	J
May	K
June	M
July	N
August	Q
September	U
October	V
November	X
December	Z

        - <BBYellowKeyCode>
            - Comdty
            - Index
            - Govt
            - Corp

- 5 = <RIC futures symbol> (*3)

5. SecurityID(48) = <Security ID> per SecurityIDSource(22)
6. MaturityMonthYear(200) = <YYYYMM>
7. SettlType(63) = 0 (regular, default) or 6 (future) (optional)
8. SettlDate(64) = <SettlementDate> (optional)
9. Exchange (required for ambigious symbologies (*3), e.g. RIC)
    - ExDestination(100) = <exchange code> (FIX 4.2)
    - Securityexchange(207) = <exchange code> (FIX 4.4 plus)

7.1.1.2 Multi-Leg

1. Symbol(55) = "[N/A]" – **Note:** Some futures exchanges have predefined strategies and may have a symbology for this. It is left to the broker to send the best/appropriate symbol to the exchange.
2. CFIcode(461) = "FMXXXX" (future, multi-Leg) [**Note:** *To be revisited during public review*]
3. SecurityType(167) = "MLEG"
4. SecuritySubType(762) = <strategy code> (see following table)
5. LegSecurity(600) = <future-symbol>
6. LegSecurityIDSource(603) = See tag 22 above
7. LegSecurityID(602) = See tag 48 above
8. LegCFIcode(608) = "FXXXXX", (future, single)
9. LegSecurityType(609) = "FUT"
10. LegMaturityMonthYear(610) = <YYYYMM>
11. LegSettlType(587) = 0 (regular, default), 6 (future) (optional)
12. LegSettlDate(588) = <SettlementDate> (optional)
13. LegSecurityExchange(616) = <exchange code>(optional) (*3)

7.1.2 Options on Futures (differences)

7.1.2.3 Single-Leg

1. Symbol(55) = < option-on-future-symbol>
2. CFIcode(461) = "O"<"P"/"C">"XFXX", (option, put/call, future)
3. SecurityType(167) = "OOF"
4. SecurityIDSource(22):

A = <BBYK option-on-future symbol>
"<root>MY<blank><strike-price><blank><yy/mm/dd><blank><BBGYellowKeyCode>"
(e.g. "AHBR 2.250 09/18/07 Corp")
5 = <RIC options on futures symbol> (*3)

5. SecurityID(48) = See tag 22
6. MaturityDate(541) = <YYYYMMDD> (local market date) (*1)
7. StrikePrice(202) = <price> (*2)

7.1.2.4 Multi-Leg

1. CFIcode(461) = "OMXFXX" (option, multi-Leg)
2. SecurityType(167) = "MLEG"
3. LegSecurity(600) = <option-on-future-symbol>
4. LegSecurityIDSource(603) = see tag 22 above
5. LegSecurityID(602) = see tag 48 above
6. LegCFIcode(608) = "O"<"P"/"C">"XFXX", (option, put/call, future)
7. LegSecurityType(609) = "OOF"
8. LegMaturityDate(611) = <YYYYMMDD> (local market date) (*1)
9. LegStrikePrice(612) = <price> (*2)

Notes:

(*1) MaturityDate(541) applies to the Option not the underlying Future.
(*2) buy-side sends the strike price in the exchange traded units even though prices are expected to be returned in the basic traded units (e.g. the traded units for corn futures is USD and the strike price will be sent in that form but LastPx is expected to be returned in USD).
(*3) the RIC security ID may be ambiguous and requires the ExDestination(100) or SecurityExchange(207) code to make the proper determination.

7.2 Price Factors

Buy-side expects the broker to automatically convert prices for instruments that trade in currency units different from the major currency units to and from major currency units (e.g. for instruments traded in USD, prices would be returned in USD).

Note: exception for Options on Futures where buy-side sends the strike price in the exchange traded units even though prices are expected to be returned in the basic traded units (e.g. the traded units for corn futures is USD and the strike price will be sent in that form but LastPx(31) is expected to be returned in USD). Note that the strike-price in the BBYK is also in exchange traded units.

7.3 Account Identification

The following identification conventions are used for the different account types.

1. **Execution Account**:
    - Buy-side will **not** specify the account identifier, it is expected that this will be known by the broker based upon the FIX session.
    - Buy-side will specify values for the ClOrdID(11) fields of the orders in the AllocationInstruction(35=J) message:
        - NoOrders(73) = <number of orders>
        - ->ClOrdID(11) = <clOrdID of order>
    - Buy-side will identify the trades to be allocated in the following fields in the execution section of the Allocation Instruction message:
        - NoExecs(124) = <number of execution reports>
        - ->LastQty(32) = quantity of the trade
        - ->LastPx(31) = price of the trade
        - ->ExecID(17) = ExecID of the trade FIX message

2. **Client Clearing Accounts**: buy-side will specify these accounts in the Allocation instruction messages using the account identifier provided by the clearing broker to buy-side (in the format provided by the clearing broker). The following allocation message AllocGrp FIX tags and values are used.
    - **Clearing by Execution Broker:**
        - ProcessCode(81) = 0 (regular)
        - AllocAccount(79) = <ClearingFirmClientAccountID>
    - **Give-up to another Clearing Broker:**
        - ProcessCode(81) = 3 (Give-up/Step-out)
        - AllocAccount(79) = < ClearingFirmClientAccountID >
        - NoNestedPartyIDs(539) = 1
        - ->NestedPartyIDSource(525) = C (Generally accepted market participant identified)
        - ->NestedPartyID(524) = <brokerID> (NASD code)
        - ->NestedPartyRole(538) = 4 (Clearing firm)

7.4 Commissions and Fees

Commissions and fees for futures and options on futures are not paid as part of the primary futures workflow since they are charged on a post-trade basis by the clearing broker through a separate statement and billing process. However, it was decided that it would be valuable to be able to identify/validate commissions and fees as part of the primary workflow.

Assumptions and constraints:

1. Futures have both executing and clearing commissions, as well as fees payable to the exchange and regulatory bodies such as the US National Futures Association.
2. Execution commissions are negotiated as a function of the execution broker and the execution method. The clearing broker (if different from the execution broker) is notified of the agreed commission, and bills the buy-side based on their transactions with the executing broker, then collects the commissions and subsequently rebates the executing broker.
3. Existing workflow for the step-out (give-up) of trades executed at one broker and cleared by another broker does not allow for the actual commission to be passed on the trade from broker to broker.
4. The commission value is typically not computed by the broker during the primary workflow cycle, but rather later – usually during a nightly billing cycle.
5. The primary execution and allocation workflow should allow for buy-sides and sell-sides to clearly identify the method of execution – and hence the correct commission rate to use. This may include identification of the execution method at point of execution, during allocated step-out (give-up) from executing broker to clearing broker, or on allocation of step-in (give-in) trades at the clearing broker.

The FIX post-trade strategy for this is:

- Optionally include execution-method on the AllocationInstruction(35=J)
    - If included the sell-side can use it to determine the commission (buy-side says what it is)
    - Configuration option
- Optionally include commission and fee values
    - If included the sell-side can either:
        - validate as part of the primary workflow (reject the AllocationInstruction(35=J) if commission values don't match)
        - validate post-primary workflow (contact buy-side if commissions and/or fees don't match)
    - Allow specification of multiple commissions with type (execution, clearing) as well as multiple fees
    - For privacy give-up broker commission values may not be included with the primary AllocationInstruction(35=J), but rather just in the "step-in" AllocationInstruction(35=J) to the give-up firm.
    - Configuration option as to whether buy-side will include and/or expect to be validated

Commissions and fees are sent "buy-side-calculated = in other words it is the responsibility of the sell-side to agree-with or reject the commissions and/or fee values if they are viewed as incorrect. There is no provision for communicating modified values back to the buy-side.

Since this is a new capability it is understood that not all sell-sides or buy-sides will be able to process/provide the commissions and fees if provided. They buy-side must know what a given sell-side has committed to:

- If the buy-side does not provide or the sell-side does not yet utilize the information provided, the traditional manual process remains in place.

- If the sell-side does process the commissions and/or fees it is understood that not all sell-sides will be able to validate the commission values during the course of the primary workflow (and reject if there is an issue) since sometimes they are computed the next day. If the sell-side is not able to validate during the primary workflow it is their responsibility to contact the buy-side later to resolve the issue.

7.4.1 Execution Method

The following FIX fields are used to specify execution method:
OrderHandlingInstSource(1032) =2 (FIA Execution Source Code)
CustOrderHandlingInst(1031)
FIA Execution Source Code

A	Phone simple	Voice
B	Phone complex	Voice
C	FCM provided screen	Electronic
D	Other provided screen	Electronic
E	Client provided platform controlled by FCM	Electronic
F	Client provided platform direct to exchange	Electronic
H	Algo engine	Electronic
J	Price at execution (price added at initial order entry, trading, middle office or time of give-up)	Voice
W	Desk - electronic	Voice
X	Desk - pit	Electronic
Y	Client - electronic	voice

7.4.2 Commissions

Futures have both execution and clearing commissions so a new repeating group for commissions has been added to Allocation and Confirmation messages.

NoCommissions	2639	Total number of commissions
->CommissionAmount	2640	Total commission amount
-> CommissionAmountType	2641	Supported values: 2 = Broker - The executing broker's commission. 3 = Clearing broker - The clearing broker's commission.
->CommissionBasis	2642	Supported values: 1 = Per unit 2 = Percent 3 = Absolute - Recommended

7.4.3 MiscFees

Miscellaneous fees repeating group is include in Allocation(35=J) and Confirmation(35=AK) messages.

Currently identified fees are the following:

- Exchange fees where applicable
- NFA fee on futures and options in the US
- Sales tax (sells only)
- Clearing fees on Indian futures

FIX Protocols and Sessions

1. FIX 4.4 Order and Allocation messages may be transmitted on the same or different FIX sessions.
2. Confirm messages may be transmitted on the same or a separate FIX session.

8 Workflows

The following diagram shows the basic workflow with optional AllocationReport(35=AS) and “step-in” AllocationInstruction(35=J).

Figure 1 Post-Trade Give-up Clearing

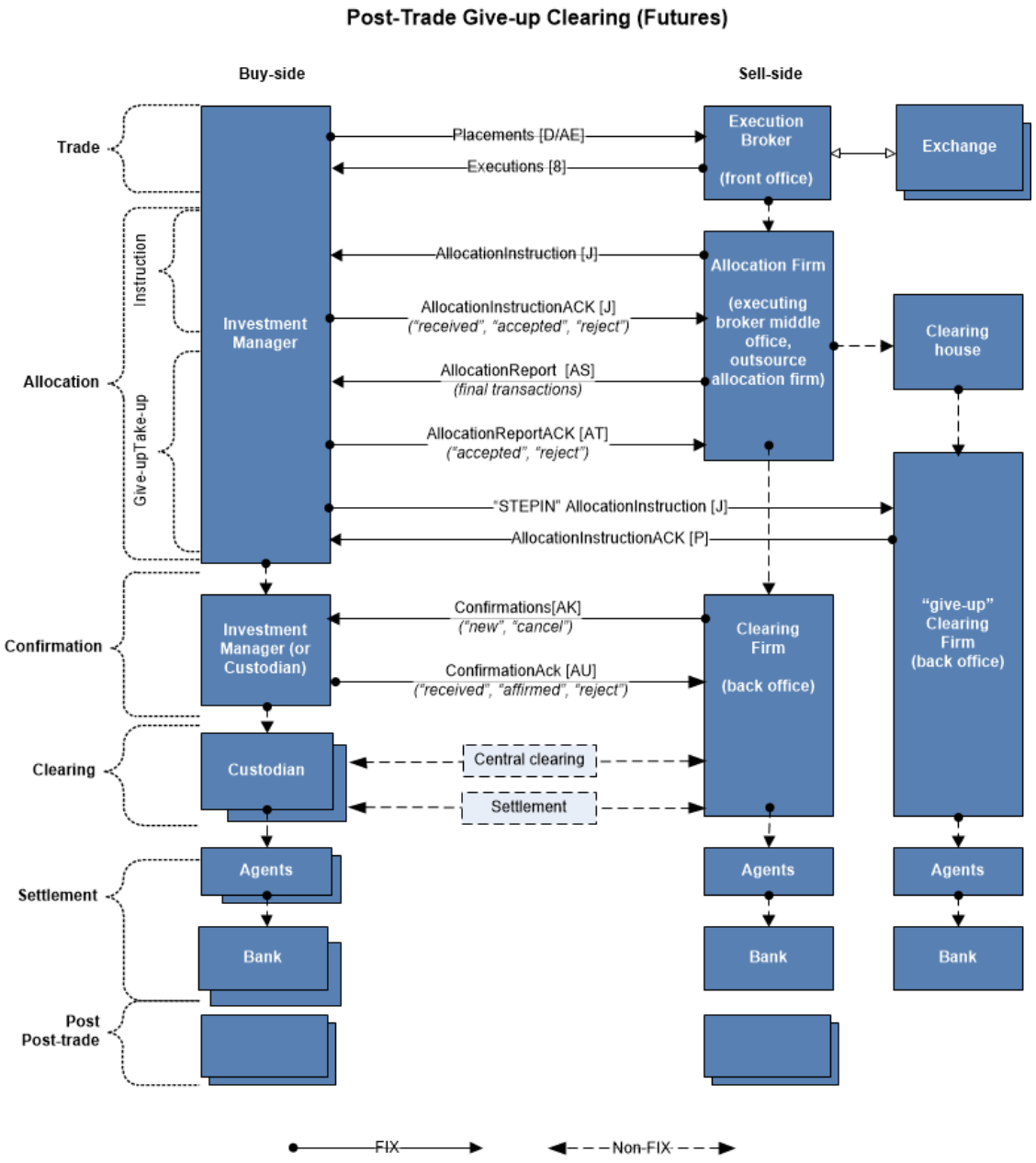


Figure 2 Sell-Side Specified Prices - Futures Order and Allocation Flow

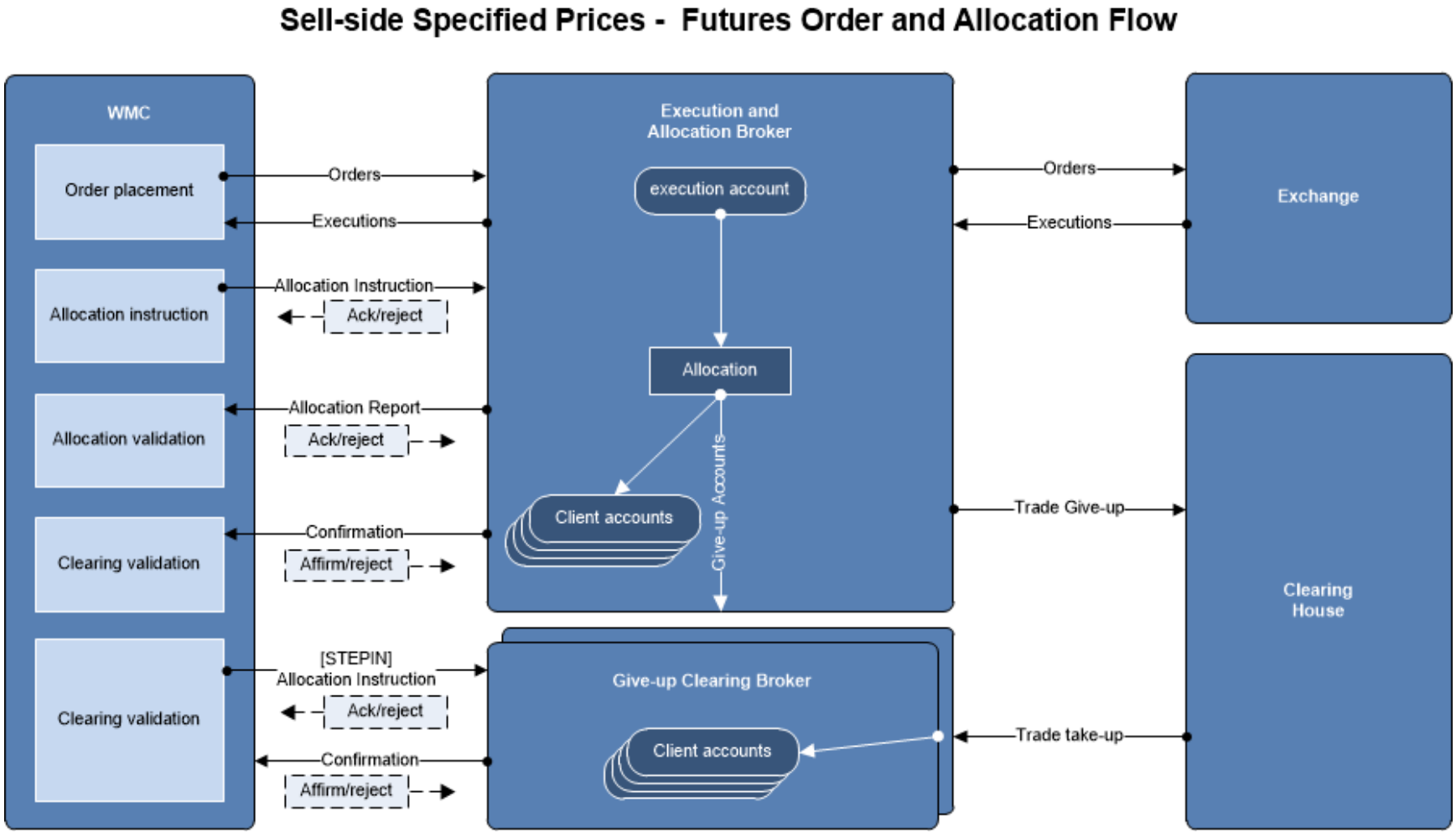
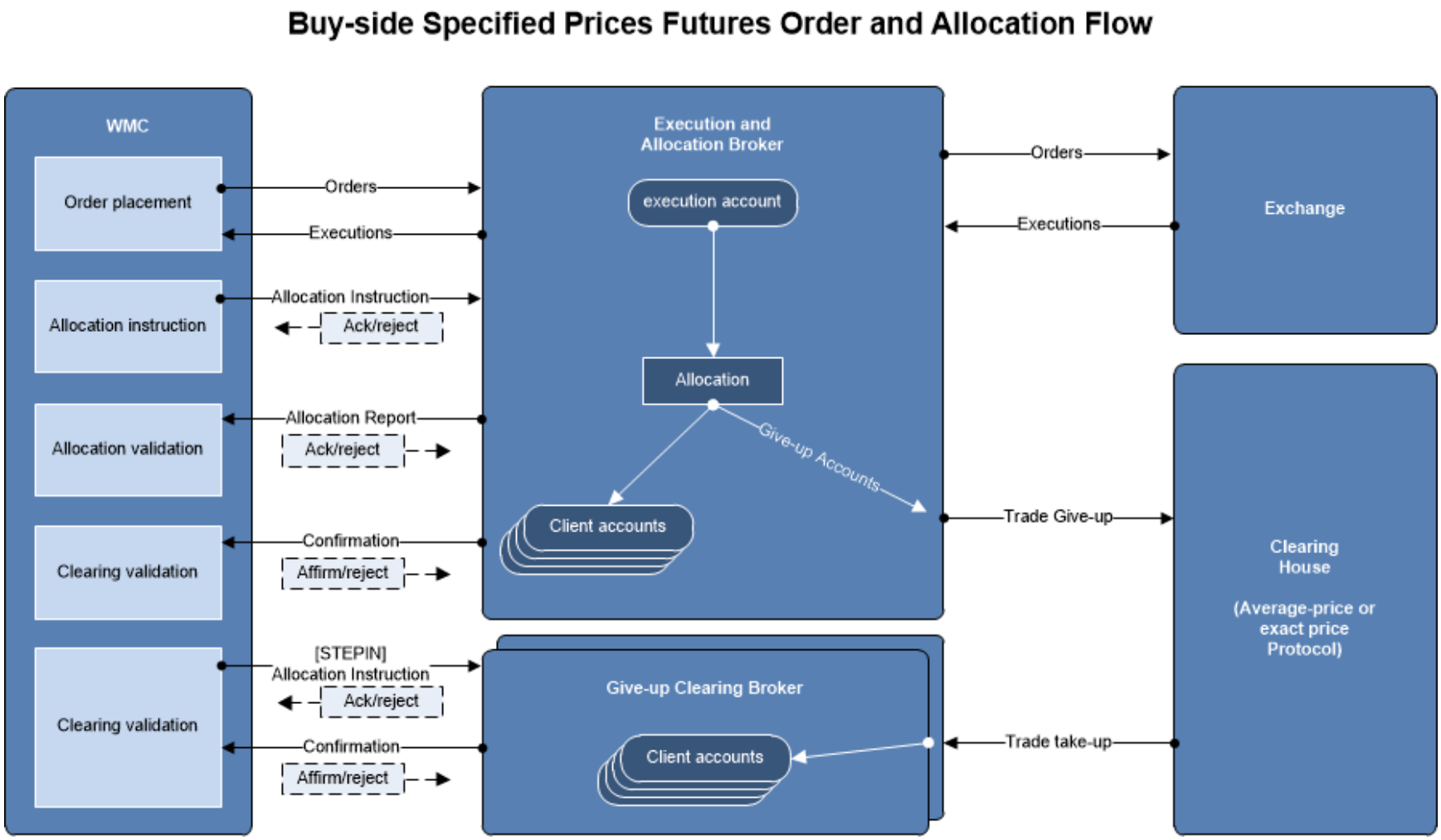


Figure Buy-Side Specified Prices Futures Order and Allocation Flow



The exchange-based give-up process requires that prices may need to be execution prices rather than average prices. The constraints are that some exchanges can give-up transactions at average prices, using an intermediate “average-price” account, but some exchanges only give-up transactions at execution prices. Prices may be controlled by either the sell-side or the buy-side. The following table describes the alternative workflows (configuration option at on-boarding):

Allocation Instruction Options	a. Primary allocation instruction to execution broker (sell-side specification of prices – prices returned in allocation report)	b. Primary allocation instruction to execution broker (buy-side specification of prices)	c. Step-in allocation instruction to clearing firm
1. Execution-price give-up / execution-price allocation	**Allocation Instruction** Orders - List of placements (optional) Executions - List of executions o Quantity(53), Price(44), ExecID(17) Allocations - Regular or give-up o Account(1) o Quantity(53) o IndividualAllocID(467) **Allocation Report returns** - Regular [average priced)(and **Confirmations**) o Account(1) (one instance of each account) o Quantity(53) o AllocAvgPx(153) – account-level average price o IndividualAllocID(467) - Give-up o Account(1) (one instance per price-level) o Quantity(53) o AllocAvgPx(153) – account-level average price o AllocPrice(366) – give-up price o IndividualAllocID(467) (account-level)	**Allocation Instruction** Orders - List of placements (optional) Executions - List of executions o Quantity(53), Price(44), ExecID(17) Allocations - Regular (and **Confirmations**) o Account(1) (one instance) o Quantity(53) o AllocAvgPx(153) – account-level average price o IndividualAllocID(467) - Give-up o Account(1) (one instance per price-level) o Quantity(53) o AllocAvgPx(153) – account-level average price o AllocPrice(366) – give-up price o IndividualAllocID(467) (account-level)	**Allocation Instruction** Orders - ClOrdID(11) = "[STEPIN] Executions - n/a Allocations - Give-in (and **Confirmations**) o Account(1), (one instance per price-level) o Quantity(53) o AllocPrice(366)= execution price o IndividualAllocID(467) [note: - give-ins are identified by quantities of execution prices received from the exchange]

Allocation Instruction Options	**a. Primary allocation instruction to execution broker (sell-side specification of prices – prices returned in allocation report)**	**b. Primary allocation instruction to execution broker (buy-side specification of prices)**	**c. Step-in allocation instruction to clearing firm**
2. Average-price-give-up / average-price allocation (only if supported by the exchange, requires configuration-time average price account identified)	-	**Allocation Instruction** Block - AvgPx(6) = average price Orders - List of placements (optional) Executions - List of executions (optional) Allocations (and **Confirmations**) - Regular or give-up o Account(1), (one instance of each account) o Quantity(53) o IndividualAllocID(467)	**Allocation Instruction** Block - AvgPx(6) = average price Orders - ClOrdID = "[STEPIN] Executions - n/a Allocations - Give-in (and **Confirmations**) o Account(1), (one instance of each account) o Quantity(53) o IndividualAllocID(467) [Note: - give-ins are identified by total quantity and average price from the average-price account configured for this clearing firm.

Allocation Instruction Options	a. Primary allocation instruction to execution broker (sell-side specification of prices – prices returned in allocation report)	b. Primary allocation instruction to execution broker (buy-side specification of prices)	c. Step-in allocation instruction to clearing firm
3. Execution-price give-up / average-price allocation	**Allocation Instruction** Orders - List of placements (optional) Executions - List of executions - Quantity(53), Price(44), ExecID(17) Allocations - Regular or give-up ○ Account(1) ○ Quantity(53) ○ IndividualAllocID(467) **AllocationReport returns** Allocations - Regular (and **Confirmations**) ○ Account(1) ○ Quantity(53) ○ AllocAvgPx(153) – account-level average price ○ IndividualAllocID(467) - Give-up ○ Account(1) ○ Quantity(53) ○ AllocAvgPx(153) – account-level average price ○ IndividualAllocID(467)	**Allocation Instruction** Orders - List of placements (optional) Executions - List of executions - Quantity(53), Price(44), ExecID(17) Allocations - Regular (and **Confirmations**) ○ Account(1) (one instance) ○ Quantity(53) ○ AllocAvgPx(153)= average price for account ○ IndividualAllocID(467) - Give-up ○ Account(1) (one instance per price-level) ○ Quantity(53) ○ AllocAvgPx(153)= average price for account ○ AllocPrice(366) – give-up price ○ IndividualAllocID(467)(optional) ○ Note: ▪ Everything must be allocated and quantity must sum and average price must match.	**Allocation Instruction** Block - AvgPx(6) = average price Orders - ClOrdID(11) = “[STEPIN]” Executions - Aggregated Execution quantity at each price-level that were given-up. Allocations - Give-in (and **Confirmations**) ○ Account(1) (one instance) ○ Quantity(53) ○ IndividualAllocID(467) [Note: give-ins are identified by execution list of price and quantity pairs received from the exchange)

8.1 Trade and Allocation Business Flow (Broker Allocated Best Fit)

1. Trades
    a. One or more orders are placed with an execution broker
    b. Trades completed and execution reports returned to buy-side
2. Primary Allocation Instruction from buy-side
    a. Allocation Instructions are given to execution broker
        i. One or more executions (price and quantity)
        ii. One or more client allocation accounts (quantity)
            1. execution broker cleared
            2. give-up to another firm for clearing
3. Allocation by broker
    a. Execution broker computes a fair average price for the trades and then partitions the trades across the allocation accounts to achieve a fair distribution of prices.
4. Allocation reporting and validation
    a. Execution broker reports to buy-side the average price and allocation information (e.g. give-up shapes) for each account.
    b. Buy-side validates that any averaged booking prices are within expected tolerances and that quantity instructions were followed.
5. Allocation Give-up
    a. Execution broker transfers give-up allocation information to any give-up clearing firms using the method of the given exchange.
6. Step-in Allocation Instruction (if required)
    a. Buy-side communicates step-in-allocation instructions to the give-up clearing broker.
7. Clearing
    a. Clearing firms “take-up” trades from the clearing house and clear trades according to instructions.
8. Clearing Confirmation
    a. Buy-side compares the booking prices from the clearing broker with the allocation report to be sure that instructions were followed.
    b. Buy-side receives final fees and sell-side settlement instructions.
9. Error Condition Resolution
    a. Unknown clearing account
        i. Due to the fact that new accounts are a frequent occurrence the execution broker is expected to institute an exception process that can identify the unknown account code, and then map the unknown account to the broker identified in the allocation instruction and continue with the allocation process. Buy-side can be contacted to confirm the account before mapping but this is not required. This process should be completed without rejecting the allocation instruction.
    b. Allocation Report Price out of tolerance
        i. buy-side notifies broker
        ii. Broker re-computes and reports back corrected allocation, or buy-side accepts the out-of-tolerance allocation and broker proceeds as planned.
    c. Modification of allocation instructions
        i. buy-side sends an Allocation Cancel
        ii. Buy-side resubmits Allocation Instructions if cancel is accepted by broker.

d. Confirm booking price vs. Allocation Report mismatch
    i. See Confirm section – depends upon configuration.
e. Modification of trades post-allocation
    i. buy-side will reject any post-allocation modifications

8.2 Trade and Allocation Business Flow (Buy-side specification of prices)

Buy-sides can optionally specify the allocation prices – either best-fit execution prices or average prices (if the exchange supports average price give-up). This presence of AllocPrice(366) or AllocAvgPx(153) indicates that the buy-side is specifying.

1. Trades
    a. One or more orders are placed with an execution broker
    b. Trades completed and execution reports returned to buy-side
2. Allocation by buy-side
    a. Best-fit execution prices
        i. Used for give-ups
        ii. Buy-side computes a fair average price for the trades and then partitions the trade execution prices across the allocation accounts to achieve a fair distribution of prices.
        iii. During this best fit allocation process, executions that have the same account, price, security, and side will be aggregated for the purposes of best fit calculation.
        iv. AllocPrice (366) is used to communicate execution prices.
    b. Average price
        i. May be used if the exchange supports average-price give-ups
        ii. Buy-side computes mathematical average price for the account
            1. Issue: could the average prices be different across accounts or do they have to be the same. Could exchange average priced give-up support this?
        iii. AllocAvgPx (153) is used to communicate average prices
    c. Notes:
        i. If the buy-side specifies prices accounts must be either all average priced or all execution priced and 366 or 153 must be included for all entries.
3. Primary Allocation Instruction from buy-side
    a. Allocation Instructions are given to execution broker
        i. One or more executions (price and quantity)
        ii. One or more client allocation accounts (quantity)
            1. execution broker cleared
            2. give-up to another firm for clearing
4. Allocation Give-up
    a. Execution broker transfers give-up allocation information to any give-up clearing firms using the method of the given exchange.
5. Step-in Allocation Instruction (if required)
    a. Buy-side communicates step-in-allocation instructions to the give-up clearing broker.
6. Clearing
    a. Clearing firms “take-up” trades from the clearing house and clear trades according to instructions.
7. Clearing Confirmation
    a. Buy-side compares the booking prices from the clearing broker with the allocation report to be sure that instructions were followed.
    b. Buy-side receives final fees and sell-side settlement instructions.
8. Error Condition Resolution
    a. Unknown clearing account
        i. Due to the fact that new accounts are a frequent occurrence the execution broker is expected to institute an exception process that can identify the unknown account

code, and then map the unknown account to the broker identified in the allocation instruction and continue with the allocation process. Buy-side can be contacted to confirm the account before mapping but this is not required. This process should be completed without rejecting the allocation instruction.

b. Allocation Report Price out of tolerance
    i. buy-side notifies broker
    ii. Broker re-computes and reports back corrected allocation, or buy-side accepts the out-of-tolerance allocation and broker proceeds as planned.
c. Modification of allocation instructions
    i. buy-side sends an Allocation Cancel
    ii. Buy-side resubmits Allocation Instructions if cancel is accepted by broker.
d. Confirm booking price vs. Allocation Report mismatch
    i. See Confirm section – depends upon configuration.
e. Modification of trades post-allocation
    i. buy-side will reject any post-allocation modifications

8.3 Allocation Workflow Notes

Synchronization of the "step-in" AllocationInstruction(35=J): the question was discussed as to whether or not the buy-side should wait for the primary AllocationInstruction(35=J) to be accepted before sending the step-in AllocationInstruction(35=J). The conclusion was that it would be left to the discretion of the buy-side. If the secondary AllocationInstruction(35=J) messages are sent before the primary is accepted there is the risk that the secondary instructions will need to be "canceled".

9 Futures Allocation Message Content and Workflow

9.1 FIX Allocation Messages

Buy-side uses the following **FIX 4.4** Allocation messages types:

1. AllocationInstruction(35=J)
    - AllocTransType(71) =0 (New)
    - AllocTransType(71) =2 (Cancel)
2. AllocationACK(35=P)
3. AllocationReport(35=AS)
    - AllocTransType(71) =0 (New)
4. AllocationReportACK(35=AT)
5. Confirmation(35=AK)
    - AllocTransType(71) =0 (New)
    - AllocTransType(71) =2 (Cancel)
6. ConfirmationACK(35=AU)

9.1.1 Allocation Message Sequence

Buy-side utilizes the following sequence of FIX 4.4 messages for allocation instruction and confirmation:

(Primary Instruction)

1. **buy-side sends >>> Allocation Instruction**

- AllocType(626)
  - 1 = Calculated (includes commissions and misc-fees)
  - 2 = Preliminary (without commissions and MiscFees),

2. **broker validates**
   - Block
     - Trades (ClOrdID(11)'s)
     - ExecutionReport(35=8)'s (ExecID(17))
   - Trade level values
     - Commissions and Fees (if included in AllocationInstruction(35=J))
3. **Broker responds <<< Allocation ACK**
   - AllocStatus(87)
     Supported values:
     - 0 = Accepted - And processed
     - 3 = Received - Message has been received but not yet processed (optional)
     - 6 = Allocation pending - Block has been matched but allocation has not started yet (optional)
4. **(if problem) Broker responds <<< Allocation ACK**
   - AllocStatus(87)
     Supported values:
     - 1 = (block level reject)
       - Problem with message (e.g. unknown account, disagreement on commissions and/or fees, block)
       - No allocations have been performed
       - Buy-side sends another allocation instruction after resolving the problem.

(After Allocation [optional - sell-side responsible for best-fit price determination])

5. **Broker sends <<< Allocation Report (prior to clearing or give-up)**
   - AllocStatus(87)
     Supported values:
     - 0 = Accepted and successfully processed
       1. Broker cleared accounts have been allocated and transactions re-shaped for give-up (best-fit average price)
       2. No accounts have been cleared
       3. Give-up accounts have NOT been forwarded to clearing broker
6. **Buy-side validates**
   - Reshaped transactions
   - Account-level average price (tag 153) vs tolerance
7. **buy-side responds >>> Allocation Report ACK**
   - AllocStatus(87)
     Supported values:
     - 0 = Accepted)
     - 1 = Block level reject
       - Fair-average price out of tolerance
       - Out-of-band resolution
       - Broker sends an updated Allocation Report or buy-side sends Cancel/New Allocation Instruction

(Secondary Step-in-allocation)

8. **buy-side sends >>> Step-in AllocationInstruction(35=J)s to give-up firms(s)**

- Step 1. AllocationInstruction(35=J)
  - Identified by ClOrdID(11) = "[STEPIN]"
  - New transaction-ids (IndividualAllocID(467)) for re-shaped transactions
  - ProcessCode(81) = 2 (step-in)
- Step 2. Same workflow as primary allocation instruction but no AllocationReport(35=AS).

(After Clearing)

Confirm Message from Allocation or give-up clearing firm

9. **Clearing firm sends <<< Confirmation (one for each transaction)**
   - ConfirmStatus(665)
     Supported values:
     - = 4 (confirmed)
       - Account has been cleared
10. **Buy-side validates**
    - Validates Transaction level values
      - Fees and taxes
        1. Note: expectation is sell-side-calc so buy-side will use the sell-side fees as long as they are within tolerance.
    - Stores sell-side settlement instructions
11. **buy-side responds >>> Confirmation ACK**
    - AffirmStatus(940)
      Supported values:
      - 3 = (affirmed)
        - buy-side marks trade as cleared
        - sell-side may send Confirmation "cancel" followed by Confirmation "new" to update.
          - Out-of-band resolution is required if one or the other side does not support.
      - 2 = (reject)
        - Booking price from allocation report did not match the booking price from the confirmation.
        - Fees exceed tolerance
        - Sell-side sends another Confirmation "new" with changes or buy-side agreement that they will accept it this time.

Notes:

- Rejection of Allocation Instruction: if an Allocation Instruction is rejected buy-side will send a new Allocation Instruction once the issue has been resolved.
- Error in Allocation Instruction: If buy-side discovers a problem with an AllocationInstruction(35=J) they will send an allocation cancel message followed by a new AllocationInstruction(35=J) message (with new AllocID(70). The broker is expected to accept these messages, make the indicated changes (either automatically or manually) and respond with an updated AllocationReport(35=AS). If the broker does not support the allocation cancel message they must be prepared for buy-side to inform them of the cancellation out-of-band and then accept a new Allocation Instruction, make the necessary changes and respond with a new AllocationReport(35=AS).
- More details on AllocationInstruction(35=J)/AllocTransType = 2 (Cancel) message are described in the Post-trade-Recommended Practices for Equites. Usage of AllocationInstruciton(35=J) /AllocTransType = 1 (Replace) is not part of the best practices workflow.
- Post allocation trade modification busts or corrections will be DKed

9.1.2 Sell-side Best-fit Price Determination Workflow Notes

When the sell-side computes the best-fit execution prices the AllocationReport(35=AS) is used to return the reshaped transactions back to the buy-side.

A client account will have only one instance in the AllocationInstruction(35=J) message but may have multiple entries in the AllocationReport(35=AS), with different allocation quantity and booking price (see next section for example).

The following fields are returned in the AllocationReport(35=AS)/**AllocGroup**:

1. NoAllocs(78)= <number of allocation account entries>
2. ->AllocAccount(79)= <client account>
3. ->AllocQty(80)= <quantity allocated to this account>
4. ->AllocPrice(366)= <booking price for the executions allocated to this account, which may be an average price>
    - Precision: rounded to the contract tick value as specified by exchange
5. ->AllocAvgPx(153)= <average price of all allocations to this account in this Allocation Report>
    - Precision: equal or greater than the contract tick value as specified by the exchange
6. ->IndividualAllocID(467)= <individualAllocID(467) from the AllocationInstruction(35=J)>
    - Note: there may be multiple entries for the same account with the same IndividualAllocID(467).

The buy-side compares the AllocAvgPx(153) to the Fair Average Price in AvgPx(6), and will warn the trader if the variance is too great. This should be a configuration option.

> *Note: Probably don't want to reject if out of tolerance because if there are a small number of execution reports with a wide price variation it may not be possible to best-fit average price within the tolerance.*

For example: 3 executions, 2 accounts – ACCT-1 with exact-price transfer and ACCT-2 with average-price transfer. (Broker identification and session fields not included)

Instruction	**Report**
35=J	35=AS
70=INST-1	70= INST-1
626=2	794=3
857=1	857=1
	87=0
71=0	71=0
73=1	73=1
11=127272536	11=127272536
124=3	**124=3**
32=1	32=1.0
17=EXEC-1	17= EXEC-1
31=6.724	31=6.724
32=3	32=3.0
17= EXEC-2	17= EXEC-2
31=6.726	31=6.726
32=2	32=2.0
17= EXEC-3	17= EXEC-3

31=6.724	31=6.724	
54=2	54=2	
55=NGX8 Comdty	55=NGX8 Comdty	
48=NGX8 Comdty	48=NGX8 Comdty	
22=A	22=A	
53=6	53=6.0	
6=6.725	6=6.725	
75=20081021	75=20081021	
78=2	**78=3**	
	(exact price give-up protocol)	
79=ACCT-1	79=ACCT-1	(to give-up broker)
80=4	**366=6.724**	(to give-up broker)
	80=2.0	(to give-up broker)
	153=6.7250000000000005	
	79=ACCT-1	(to give-up broker)
	366=6.726	(to give-up broker)
	80=2.0	(to give-up broker)
	153=6.7250000000000005	
	(average price give-up protocol)	
79=ACCT-2	79=ACCT-2	(to give-up broker)
80=2	**366=6.725**	(to give-up broker)
	80=2.0	(to give-up broker)
	153=6.7250000000000001	

9.1.2.5 Special Allocation Restrictions

Buy-side will configure its allocation instructions to conform to any such restrictions or work out an appropriate protocol with the broker on a case by case basis

Currently identified instances:

- Australian no average-price clearing restriction.

9.1.3 Electronic Confirmation of Clearing

Buy-side expects to receive electronic confirmation of booked price from the clearing broker for each cleared transaction (execution broker or give-up broker).

9.1.3.6 Confirmation(35=AK) Message

The FIX Confirm message is used for this purpose with the following tags:

The following fields are returned in the Confirmation message:

- ConfirmID(664) = unique ID created by broker
- ConfirmTransType(666) = 0(new)

- ConfirmType(773) = 2(confirmation)
- ConfirmStatus(665) = 4(confirmed)
- LegalConfirm(650) = Y(legal confirmation)
- AllocID(70) = <allocID from Allocation Instruction>
- IndividualAllocID(467) = <IndividualAllocID(467) from AllocationInstruction(35=J) AllocGrp entry>
- AllocAccount(79) = <client account>
- AllocQty(80) = <quantity allocated to this account>
- AvgPx(6) = <booking price of the executions allocated to this account>
- MaturityMonthYear(200)
- TradeDate(75) = YYYYMMDD(date of allocation instruction)
- TransactTime(60)
- Side(54)
- Symbol(55)
- SecurityID(48)
- SecurityIDSource(22)
- SecurityType(167)
- CFIcode(461)
- SettltType(63)
- SettltDate(64) = <settlement date>
- NoPartyIDs(453) = 2
- ->PartyRole(452) = 1(executing broker)
- ->PartyIDSource(447) = “C”
- ->PartyID(448) = <brokerID>(see broker codes table)
- ->PartyRole(452) = 4(clearing broker)
- ->PartyIDSource(447) = “C”
- ->PartyID(448) = <brokerID>(see broker codes table)

FIX required Confirmation message fields that will be ignored if included:

- NoLegs (555)
- NoCapacities (862)
- OrderCapacity (528)
- OrderCapacityQty (863)
- GrossTradeAmt (381)
- NetMoney (118)

Notes:

- If the allocation broker does not support Confirmation(35=AU) messages the allocation session may be configured to assume that the AllocationReport(35=AS) implies confirmation of the account cleared by the allocating broker.

- If a give-up broker does not support Confirmation(35=AU) messages then Confirmation(35=AU) message will not be expected to be received by buy-side for the transaction and a manual clearing confirmation will have to be made out-of-band.
- At the end of day any un-confirmed trade clearings will have to be manually confirmed by buy-side trade administrators. Confirmation(35=AU) messages that arrive post this manual confirmation will be rejected.

9.1.3.7 ConfirmationAck(35=AU)

The following fields are returned in the ConfirmationAck(35=AU) message:

a. ConfirmID(664) = Unique ID created by broker
b. TradeDate(75) = YYYYMMDD(date of allocation instruction)
c. TransactTime(60)
d. AffirmStatus (940) = 2 (Rejected), 3 (Affirmed)

9.1.4 AllocationInstruction Cancel

AllocationInstruction(35=J)/AllocTransType(71) = 2 (Cancel) is part of the standard workflow. Please refer to the FIX Post-trade Recommended Practices for Equities for a detailed specification of this workflow. AllocationInstruction(35=J)/AllocTransType(71)= 1 (Replace) is not a supported part of the workflow because of the difficulty of dealing with modifications to reshaped transactions.

Buy-side expects to be able to send FIX AllocationInstruction(35=J)/AllocTransType(71) = 2 (Cancel) messages, and then resubmit the AllocationInstruction(35-J) with new AllocID(70). If the broker rejects the Allocation Cancel the issue will be resolved out-of-band. If the broker does not support FIX AllocationInstruction(35=J)/AllocTransType(71) = 2 (Cancel) then the entire resolution process will be handled out-of-band.

10 Multi-Leg Futures Trading

Buy-side will use FIX 4.4 for multi-leg orders.

10.1 Multi-Leg Order Messages

The following are the additional FIX 4.4 message types buy-side will utilize:

10.1.1 NewOrderMultileg(35=AB)

- ClOrdID(11) = <buy-side order identifier>
- Symbol(55) = "[N/A)" (*1)
- Side(54) = “B” (*1)
- SecuritySubType(762) = <strategy code> (see following table)
- CFIcode(461) = “FMXXXX”
- SecurityType(167) = “MLEG”
- Quantity(38) = <quantity>

- OrderType(40) = 1 (market), 2(limit)
- Price (44) = <differential price> (*2)
- TimeInForce(59) = 0(day)
- NoLegs(555) = <no legs>
- ->LegSecurity(600) = “<root>MY <BBGYellowKey>” (e.g. “GCG0 Comdty ”)
- ->LegSecuritySourceID(603) = A (Bloomberg) or 5 (RIC)
- ->LegSecurityID(602) = “<root>MY <BBGYellowKey>” (e.g. “GCG0 Comdty”) or <RIC code>
- ->LegSecurityExchange(616)
    - ExDestination(100): If not using BBGYellowKey,
    - [Not included] if using BBGYellowKey
- ->LegCFIcode(608) = “FXXXXX”, (future, single)
- ->LegSettlType(587) = “0” or “6”
- ->LegSettltDate(588) = <settlementDate>
- ->LegSecurityType(609) = “FUT”
- ->LegRefID(654) = <buy-side leg identifier>
- ->LegRatio(623) = 1
- ->LegSide(624) =1 (buy), =2 (sell)
- ->LegSettlType(587) = 0 (regular) or 6 (future)
- ->LegSettlDate(588)(optional) = <YYYYMMDD>

Notes:

- Multi-leg orders requiring tails will be submitted as two orders: the first a symmetrical multi-leg order and the second a separate order for the tail contract.
- (*1) listed spread symbols: at this time buy-side will not specify any listed spread symbols. There is also no specified order to the legs and the broker must identify the strategy through the strategy code and leg symbols and then order the legs as required by the exchange to trade the instrument. At some point in the future buy-side may start utilizing listed-spread symbols at which time tags 55 and 54 will be utilized.
- (*2) varies by strategy type.

10.1.2 Strategy Codes

Spread Strategy	SecuritySubType(762)
Calendar	
Foreign Exchange	FX
Reduced Tick	RT
Standard	SP
Equities	EQ
Butterfly	BF
Condor	CF

Strip	FS
Inter-commodity	IS
ini S&P MidCap 400 E-mini Russell 2000	EC
Crack 1:1	C1
Pack	PK
Month Pack	MP
Pack Butterfly	PB
Double Butterfly	DF
Pack Spread	PS
Bundle	FB
Bundle Spread	BS

10.1.3 Execution Reports –(35=8]

The following execution report types are expected to result in a single ExecutionReport(35=8) message:

- ExecType(150)= 0 (New) – For ACK
- ExecType(150)= 8 (Rejected) – For order rejection

The following execution report types may have just the summary execution report or the summary and the legs

- ExecType(150)= 4 (Canceled)
- ExecType(150)= 6 (Pending Cancel)

All other execution report types are expected to have multiple execution report messages:

- ExecType(150)= F (Partial fill or fill)
- ExecType(150)= 6 (Pending Cancel)
- ExecType(150)= H (Trade cancel)
- ExecType(150)= E (Pending replace)
- ExecType(150)= 5 (Replace)

These report types will have one summary execution report and one leg execution report for each leg (*2). They will be formatted as described in the following table:

Field	Summary ExecutionReport(35=8)	Leg ExecutionReport(35=8)
MultiLegReportingType(442)	Supported values: 3 = Multi-leg security	Supported values: 2 = Individual leg of a multi-leg security
OrderID(37)	<OrderID>	Same as summary
ExecID(17)	<unique exec id>	<unique exec id>
ClOrdID(11)	<ClOrdID>	Same as summary

Symbol(55)	“[N/A]”	<symbol>
SecurityType(167)	“MLEG”	“FUT”
CFICode(461)	“FMXXXX”	“FXXXXX”
SettlType(63)	Null	Supported values: 0 = Regular settlement 6 = Future
SettlDate(64)	Null	<SettlementDate>
LegRefID(654)	Null	<LegRefID(654)>
ExecType(150)	Supported values: 3 = Done for day 4 = Canceled 5 = Replaced 6 = Pending cancel E = Pending replace F = Trade – Partial fill or fill G = Trade correct H = Trade cancel	Same as summary
OrdStatus(39)	Supported values: 0 = New 1 = Partially filled 2 = Filled 3 = Done for day 4 = Canceled 5 = Replaced 6 = Pending cancel E = Pending replace	Same as summary
Side(54)	Supported values: B = Buy	Supported values: <LegSide(624)>
OrderQty(38)	<order qty>	Same as summary
LastQty(32)	<quantity of this fill>	Same as summary
LeavesQty(151)	<leaves quantity>	Same as summary
CumQty(14)	<cumulative quantity>	Same as summary
Price(44)	<same as order>	<same as order leg>
LastPx(31)	<net price>	<leg price>
AvgPx(6)	<average net>	<leg average price>

Notes:

(*2) **Matched and unmatched Execution Report Handling:**

- **Matched Leg Execution Reports**: generally, for multi-leg execution reports it is expected that the execution reports for each fill will arrive in sets, one per leg, with matching quantities according to the leg ratios, along with a summary execution report. There is no assumption on the order of the leg execution reports relative to the order message repeating group order, nor to the ordering of the summary execution report to the legs that are summarized.
- **Unmatched Leg Execution Reports**: certain exchanges (e.g. **EUREX)** do not necessarily return matching leg fill quantities. These unmatched quantity fills must be passed back via FIX so that they can be included in the Allocation Instruction for clearing. They are expected to still arrive in sets with summary execution reports but there may be more than one execution report for each leg in the set and the quantities do not necessarily match. The following mechanism will be used for such un-matched fills:

One or more groups that include a summary execution report along with one or more execution reports for each leg with matching subtotals for each leg's execution reports.

For example an order for 10 for a 2 leg strategy could result in a summary execution report and two execution reports for one leg with quantities of 5 and 5 respectively and 3 execution reports for the other leg with quantities of 1, 3 and 6 for a total of 6 execution reports (one summary and 5 legs). Complete filling of an order may include multiple such same-subtotal sets. There is no assumption on the order of the leg execution reports relative to the order message repeating group order, nor to the ordering of the summary execution report to the legs that are summarized.

10.1.4 MultilegOrderCancelReplace(35=AC)

Multi-leg orders are replaced using the MultilegOrderCancelReplace(35=AC) message. The Buy-side will only replace price and quantity.

10.1.5 Cancel(35=F)

Multi-leg orders utilize the Cancel(35=F) cancel message.

10.2 Allocation of Multi-leg trades

Each leg of a multi-leg allocation will have a separate AllocationInstruction(35=J) message (as opposed to a multi-leg allocation instruction) and separate AllocationReport(35=AS) messages are expected to be returned. The Allocation instructions for each leg will have the same accounts and allocation instructions and a parallel set of leg execution reports from one or more orders and only those execution reports (unmatched execution report sets must have matching quantity totals when ratios are taken into account). This submission of synchronized execution reports allows fair average price allocation to be performed independently but still result in a fair differential average price.

Allocation Instructions for all legs will be sent. The Allocation Instruction messages for multi-leg allocations will also be linked in the event that the broker's algorithm requires access to all legs at the same time to do the allocation.

Allocation Instructions are linked with the following fields:

- AllocLinkID(196)= <unique identifier for the multi-leg allocation> (same on all linked leg AllocationInstruction(35=J) Messages)
- AllocLinkType(197)= <number-of-legs>

10.3 Special Allocation of Multi-leg trades

The following situations have been identified:

- Singapore exchange requires a balanced allocation. This is handled by submitting only one mleg-order in allocation instructions for Singapore exchange allocation

11 FIX 4.4 Message Formats – Futures

Note: While the base protocol is FIX 4.4, additional tags or additional valid values from FIX 5.0 or later have been added as needed to meet industry post-trade processing requirements. These are identified in the message format tables ("[FIX 5.0 or later]") and may require specific exception configuration for FIX engines. The FIX Global Technical Committee has approved this as accepted practice.

Legend for Required column

- Y = yes
- N = no
- C = conditionally with criteria in parenthesis
- recmd = optional, but recommended
- Opt = Optional
- n/a = Not applicable

11.1 Futures Order Messages – Futures - Outright

11.1.1 NewOrderSingle(35=D), OrderCancelReplace(35=G), OrderCancelRequest(35=F)

FIX tag	FIX tag #	Included NewOrder	Included Replace	Included Cancel	Valid values – Future
SenderSubID	50	Y	Same	Same	<trader initials>
MessageType	35	D	G	F	
ClOrdID	11	Y	Same	Same	9 digit number
*<**Parties**> Component*					
NoPtyIDs	453	Y	Same	Same	Supported values: 1
->PartyID	448	Y	Same	Same	<emailAddress>
->PartyIDSource	447	Y	Same	Same	Supported values: C = Generally accepted market participant identifier
->PartyIDRole	452	Y	Same	Same	Supported values: 11 =Iinitiating trader
*End <**Parties**> Component*					
Account	1	N	N	N	Not included (*1)
Currency	15	Y	Same	Same	<Currency code>
SettlmntTyp	63	Opt	Opt	Opt	0 = Regular 6 = Future
SettltDate	64	Opt	Opt	opt	<SettlementDate>
HandlInst	21	Y	Same	N	Supported values: 3 = Manual order best execution
ExecInst	18	Y	Same	N	Supported values: 1 = Not held
ExDestination	100	(Y if using BB non-YK symbology)	Same	Same	<Exchange code> (see table)
*<**Instrument**> Component*					
Symbol	55	Y	Same	Same	If 22 = A: <BBYK-future-symbol> If 22 =5: <BB non-YK future symbol>
SecurityID	48	Y	Same	Same	(see ID source)
SecurityIDSource	22	Y	Same	Same	Supported values:

FIX tag	FIX tag #	Included NewOrder	Included Replace	Included Cancel	Valid values – Future
					5 = Bloomberg Symbol non-YK A = Bloomberg Yellowkey (*2)
CFIcode	461	Y	Same	Same	"FXXXXX"
SecurityType	167	Y	Same	Same	"FUT"
MaturityMonthYear	200	Y	Same	Same	(for standardized instruments)
*End <**Instrument**> Component*					
Side	54	Y	Same	Same	Supported values: 1 = Buy 2 = Sell
TransactTime	60	Y	Y	Y	<dateTime>
*<**OrderQtyData**> Component*					
OrderQty	38	Y	Y	Same	<Order quantity>
*End <**OrderQtyData**> Component*					
OrdType	40	Y	Y	Same	Supported values: 1 = Market 2 = Limit
Price	44	Y	Y	Same	<Limit price>
TimeInForce	59	Y	Same	Same	Supported values: 0 = Day

Notes:

1. Account (tag 1) is not included with the New Order message, any broker-internal execution holding account must be defaulted by broker.

11.1.2 Execution Report Message – Futures - Outright

FIX tag	FIX tag #	Required	Description
MessageType	35	Y	Supported values: 8
targetSubID	57	N	(Mirror order in included)
OrderID	37	Y	
ClOrdID	11	Y	
OrigClOrdID	41	C (as required by FIX standard)	
*<**Parties**> Component*			
NoPtyIDs	453	N	<number> (multiple parties may be returned)
->PartyID	**448**	**N**	**<id>**
->PartyIDSource	447	N	Supported values: C = Generally accepted market participant identifier
->PartyIDRole	452	N	Supported values: 1 = Executing Firm - Others will be accepted but ignored
*End <**Parties**> Component*			
ExecID	17	Y	
ExecType	150	Y	Supported values: 0 = New 3 = Done-for-day 4 = Canceled 5 = Replaced 6 = Pending cancel 8 = Rejected A = Pending new E = Pending replace F = Trade G = Trade correct H = Trade cancel

FIX tag	FIX tag #	Required	Description
OrdStatus	39	Y	0 = new 1 = partial 2 = filled 3 = done-for-day 4 = canceled 6 = pending cancel 8 = rejected A = pending new E = Pending replace
SettlType	63	opt	Supported values: 0 = Regular – Default if not specified 6 = Future
SettltDate	63	Y	<Settlement date>
<***Instrument***> *Component*			
Symbol	55	Y	
SecurityID	48	Y	
SecurityIDSource	22	Y	
CFIcode	461	Y	
SecurityType	167	Y	
MaturityMonthYear	200	C(if included in order)	
End <***Instrument***> *Component*			
Side	54	Y	
<***OrderQtyData***> *Component*			
OrderQty	38	Y	
End <***OrderQtyData***> *Component*			
OrdType	40	Y	
Price	44	C(limit)	
Currency	15	N	
TimeInForce	59	N	
ExecInst	18	N	
LastQty	32	Y	
LastPx	31	Y	Apply multipliers as needed to convert to base currency units (e.g. dollars and cents)
LastMkt	30	Y	<MIC code>
LeavesQty	151	Y	

FIX tag	FIX tag #	Required	Description
CumQty	14	Y	
AvgPx	6	Y	Apply multipliers as needed to convert to base currency units (e.g. dollars and cents)
TradeDate	75	Y	
TransactTime	60	Y	
HandlInst	21	N	
OrderHandlingInstSource	1032	recmd	{*FIX 5.0 or later} Supported values: 2 = FIA Execution Source Code
CustOrderHandlingInst	1031	recmd	{*FIX 5.0 or later} Supported values: A = Phone simple [PhoneSimple] B = Phone complex C = FCM provided screen [FCMProvidedScreen] D = Other provided screen [OtherProvidedScreen] E = Client provided platform, controlled by FCM [ClientProvidedPlatformControlledByFCM] F = Client provided platform, direct to exchange [ClientProvidedPlatformDirectToExchange] H = Algo engine [AlgoEngine] J = Price at execution (price added at initial order entry, trading, middle office or time of give-up) [PriceAtExecution] W = Desk - Electronic [DeskElectronic] X = Desk - Pit [DeskPit] Y = Client - Electronic [ClientElectronic] Z = Client - Pit [ClientPit]

11.2 Futures Order Messages – Multi-leg

Buy-side utilizes FIX 4.4 standard message flow and fields as described below.

11.2.1 FIX 4.4 - NewOrderMultileg(35=AB), MultilegOrderCancelReplace(35=AC), OrderCancel(35=F)

FIX tag	FIX tag #	Included NewOrder	Included Replace	Included Cancel	Valid values – Future
MessageType	35	AB	AC	F	
ClOrdID	11	Y	same	same	<unique client generated id>
*<**Parties**> Component*					
NoPtyIDs	453	Y	same	same	Supported values: 1
->PartyID	448	Y	same	same	<trader-email-Address>
->PartyIDSource	447	Y	same	same	Supported values: C
->PartyIDRole	452	Y	same	same	Supported values: 11 = initiating trader
*End <**Parties**> Component*					
Account	1	N	N	N	(not included)
Currency	15	Y	same	same	<Currency code>
ExecInst	18	Y	same	opt	Supported values: 1 = not held
HandlInst	21	Y	same	opt	Supported values: 3 = manual order best execution
*<**OrderQtyData**> Component*					
OrderQty	38	Y	Y	Y	<Order quantity>
*End <**OrderQtyData**> Component*					
OrderType	40	Y	Y	Y	Supported values: 1 = Market 2 = Limit
Price	44	(Y limit orders)	Y	same	<differential price> (varies by strategy)
*<**Instrument**> Component*					
Symbol	55	Y	same	same	Supported values: "[N/A]"
SecurityType	167	Y	same	same	Supported values:

FIX tag	FIX tag #	Included NewOrder	Included Replace	Included Cancel	Valid values – Future
					MLEG = Multileg instrument
SecuritySubType	762	Y	same	same	<strategy-code> (see strategy codes table)
CFIcode	461	Y	same	same	Supported values: "FMXXXX"
*End <**Instrument**> Component*					
Side	54	Y	same	same	Supported values: B = As defined
TimeInForce	59	Y	same	same	Supported values: 0 = Day
TransactTime	60	Y	same	same	<dateTime>
*<**LegOrdGrp**> Component*					
NoLegs	555	Y	same	N	<number>
*<**InstrumentLeg**> Component*					
->LegSymbol	600	Y	same	N	<symbol> (see tag 603 for format)
->LegSecurityID	602	Y	same	N	<id> (see tag 603)
->LegSecurityIDSource	603	Y	same	N	Supported values: 5 = RIC A = Bloomberg Yellowkey (*2)
->LegCFIcode	608	Y	same	N	FXXXXX"
->LegSecurityType	609	Y	same	N	Supported values: FUT
->LegMaturityMonthYear	610	Y	same	same	<YYYYMM>
->LegSecurityExchange	616	(y when using BB non-YK symbology)	same	N	< Exchange code> (see table)
->LegRatioQty	623	Y	same	same	Supported values: 1.0
->LegSide	624	Y	same	same	Supported values: 1 = Buy 2 = Sell
->LegCurrency	556	Optional	same	same	Same as tag 15
*End <**InstrumentLeg**> Component*					
->LegQty	687	Optional	same	same	Same as tag 38
->LegRefID	654	Y	same	same	<buy-side leg identifier> (unique within context of this

FIX tag	FIX tag #	Included NewOrder	Included Replace	Included Cancel	Valid values – Future
					ClOrdID)
->LegSettlType	587	Opt	same	N	Supported values: 0 = Regular – Default if not specified 6 = Future
->LegSettltDate	588	Y	Same	N	<SettlemnetDate>
*End <**LegOrdGrp**> Component*					

11.2.2 ExecutionReport(35=8) message: Multi-leg – Futures

FIX tag	FIX tag #	Required	Summary – notes	Leg - notes
MessageType	35=8	Y		
OrderID	37	Y	<OrderID>	Same as summary
ClOrdID	11	Y	<ClOrdID>	Same as summary
*<**InstrmtLegExecGrp**> Component*				
LegRefID	654	C(leg only)	n/a	<legRefID>
*End <**InstrmtLegExecGrp**> Component*				
ExecID	17	Y	<unique exec id>	<unique exec id> (each execution report message must be unique)
ExecType	150	Y	Supported values: 0 = New 3 = Done for day 4 = Canceled 5 = Replaced 6 =Pending Cancel 8 = Rejected A = Pending new E = Pending replace F = Trade – Partial fill or fill G = Trade Correct H = Trade Cancel	Same as summary
OrdStatus	39	Y	Supported values: 0 = New 1 = Partially filled 2 = Filled 3 = Done fo Day 4 = Canceled 6 = Pending Cancel 8 = Rejected A = Pending New E = Pending replace	Same as summary
SettlType	63	Opt, C(leg only)	N/A	0 = Regular – Default if not specified 6 = Future
SetllDate	64	C(leg only)	N/A	<Settlement date>
*<**Instrument**> Component*				

FIX tag	FIX tag #	Required	Summary – notes	Leg - notes
Symbol	55	Y	Supported values: [N/A]	<LegSecurity>
SecutityIDSource	22	C(leg only)	N/A	<LegID-source>
SecurityID	48	C(leg only)	N/A	<LegSecurityID>
SecurityType	167	Y	Supported values: MLEG	"FUT"
SecuritySubType	762	N		
CFIcode	461	Y	Supported values: "FMXXXX"	"FXXXXX"
*End <**Instrument**> Component*				
Side	54	Y	Supported values: B = Buy	<LegSide>
*<**OrderQtyData**> Component*				
OrderQty	38	Y	<order qty>	Same as summary
*End <**OrderQtyData**> Component*				
OrdType	40	Y	<order type>	Same as summary
Price	44	C(if limit order, summary only)	<differential price>	na
Currency	15	N		
TimeInForce	59	N	<time-in-force>	Same as summary
ExecInst	18	N		
LastQty	32	Y	<quantity of this fill>	Same as summary
LastPx	31	Y	<net price>	<leg-price> Apply multipliers as needed to convert to base currency units (e.g. dollars and cents)
LastMkt	30	C(leg)		<MIC code>
LeavesQty	151	Y	<leaves quantity>	Same as summary
CumQty	14	Y	<cumulative quantity>	Same as summary
AvgPx	6	Y	<average net>	<leg average price> Apply multipliers as needed to convert to base currency units (e.g. dollars and cents)
TradeDate	75	C(leg)		<YYYYMMDD>

FIX tag	FIX tag #	Required	Summary – notes	Leg - notes
TransactTime	60	Y	<date-time>	<date-time>
HandlInst	21	N		
MultiLegReportingType	442	Y	Supported values: 3 = Multi-leg summary	2 = Individual leg
OrderHandlingInstSource	1032	recmd	{*FIX 5.0 or later} Supported values: 2 = FIA Execution Source Code	
CustOrderHandlingInst	1031	recmd	{*FIX 5.0 or later} Supported values: A = Phone simple [PhoneSimple] B = Phone complex C = FCM provided screen [FCMProvidedScreen] D = Other provided screen [OtherProvidedScreen] E = Client provided platform controlled by FCM [ClientProvidedPlatformControlledByFCM] F = Client provided platform direct to exchange [ClientProvidedPlatformDirectToExchange] H = Algo engine [AlgoEngine] J = Price at execution (price added at initial order entry, trading, middle office or time of give-up) [PriceAtExecution] W = Desk - electronic [DeskElectronic] X = Desk - pit [DeskPit] Y = Client - electronic [ClientElectronic] Z = Client - pit [ClientPit]	

11.3 FIX 4.4 Allocation Instruction/Report Messages

11.3.1 Futures

FIX tag	FIX tag #	Future - Valid values	Included with Allocation Instruction	Required in returned Allocation Report
MessageType	35		J	AS
senderSubID	50	<trader initials>	Y	N
AllocID	70	<allocID from Allocation Instruction message>	Y	Y
AllocTransType	71	Supported values: 0 =New	Y	Y
AllocType	626	Supported values: 1 =Calculated -Required if comissions and/or fees are included 2 = Buy-side Preliminary - No comission or fees included	Y	n/a
AllocReportID	755	<unique identifier for this message>	n/a	Y
AllocReportType	794	Supported values: 3 = Sellside calculated using any preliminary fees	n/a	Y
AllocStatus	87	Supported values: 0 = Accepted - Successfully processed	n/a	Y
AllocLinkID	196	<unique identifier for all legs of a Mleg order> - Used to identify a the group of allocations associated with an multileg placement. The identifier is created by the buy-side.	(Y mleg allocs)	(Y mleg allocs)
AllocLinkType	197	<number of legs of a Mleg order>	(Y mleg allocs)	(Y mleg allocs)
AllocNoOrdersType	857	Supported values: 1 = Explicit List Provided	Y	Y
*<**OrdAllocGrp**> Component*				
NoOrders	73	<integer>	Y	Y
->ClOrdID	11	Supported values: <ClOrdID> (list of ClOrdID(11)'s of the orders) or	Y	Y

FIX tag	FIX tag #	Future - Valid values	Included with Allocation Instruction	Required in returned Allocation Report
		"[STEPIN]" - for step-in AllocationInstruction, ClOrdID(11)'s are not relevant or "[MANUAL]" - for manual placements where there are no ClOrdID(11)'s		
*End <**OrdAllocGrp**> Component*				
*<**ExecAllocGrp**> Component*				
NoExecs	124	<integer>	C (not required for step-in)	Y
->LastQty	32	<numberOfShares>	C	Y
->ExecID	17	<execID of execution report>	C	Y
->LastPx	31	<price>	C	Y
*End <**ExecAllocGrp**> Component*				
Side	54	Supported values: 1 = Buy 2 = Sell	Y	Y
*<**ExecAllocGrp**> Component*				
Symbol	55	<symbol>	Y	Y
SecurityID	48	(see instrument definition)	Y	Y
IDSource	22	(see instrument definition)	Y	Y
SecurityType	167	Supported values: FUT = Futures	Y	Y
CFIcode	461	Supported values: "FXXXXX"	Y	Y
SecurityExchange	207	(see instrument definition)	Optional	Optional
MaturityMonthYear	200	<MMYY> (maturity month for standardized securities)	Y	Y
*End <**ExecAllocGrp**> Component*				
Quantity	53	<total quantity>	Y	Y
AvgPx	6	<imputed weighted average price of all executions>	Y	Y
TradeDate	75	<date>	Y	Y

FIX tag	FIX tag #	Future - Valid values	Included with Allocation Instruction	Required in returned Allocation Report
SettlType	63	Supported values: 0 = Regular – Default if not specified	Opt	Opt
SettlDate	64	<Settlement date>	Y	Y
OrderHandlingInstSource	1032	{*FIX 5.0 or later} Supported values: 2 = FIA Execution Source Code	Recmd [**Note:** *Inclusion of this field within the AllocationInstruction(35=J) message is TBD pending the completion of a corresponding Gap Analysis*]	recmd
CustOrderHandlingInst	1031	{*FIX 5.0 or later} Supported values: A = Phone simple [PhoneSimple] B = Phone complex C = FCM provided screen [FCMProvidedScreen] D = Other provided screen [OtherProvidedScreen] E = Client provided platform controlled by FCM [ClientProvidedPlatformControlledByFCM] F = Client provided platform direct to exchange [ClientProvidedPlatformDirectToExchange] H = Algo engine [AlgoEngine] J = Price at execution (price added at initial order entry, trading, middle office or time of give-up) [PriceAtExecution] W = Desk - Electronic [DeskElectronic] X = Desk - Pit [DeskPit] Y = Client - Electronic [ClientElectronic] Z = Client - Pit [ClientPit]	Recmd [**Note:** *Inclusion of this field within the AllocationInstruction(35=J) message is TBD pending the completion of a corresponding Gap Analysis*]	recmd
*<**AllocGrp**> Component*				
NoAllocs	78	<integer>	Y	Y
->AllocAccount	79	<ClearingBrokerAccountID>	Y	Y

FIX tag	FIX tag #	Future - Valid values	Included with Allocation Instruction	Required in returned Allocation Report
-> AllocPrice	366	<price> Booking price for these trades. Note: the pair AllocAccount(79) and AllocPrice(366) must be unique – allocations to a given account are rolled-up by price. Note: if AllocPrice(366) is included in the AllocationInstruction(35=J) message in at least one entry, it must be included for all entries.	Opt (buy-side specified best-fit prices)	Y
->AllocQty	80	<qty>	Y	Y
-> IndividualAllocID	467	<buy-side generated id> Note: the [STEPIN)AllocationInsturuction has new transactionIds for any re-shaped transactionsfrom the AllocationReport	Y	Y
->ProcessCode	81	Supported values: 0 = Regular 2 = Step-in - All ProcessCode(81) tags for step-in AllocationInstruction(35=J) messages must be set to 2 (Step-In) 3 = Give-up	Y	Y
*<**NestedParties**> Component*				
->NoNestedPartyIDs	539	Supported values: 1 - For give-up	C (81=3)	C (81=3)
--->NestedPartyID	524	<brokerID> (see Broker codes table)	C (81=3)	Y C (81=3)
--->NestedPartyIDSource	525	Supported values: C = Generally accepted market participant identifier	Y C (81=3)	C (81=3)
--->NestedPartyRole	538	Supported values: 4 = Clearing firm	C (81=3)	C (81=3)
*End <**NestedParties**> Component*				
*<**AllocCommissionDataGrp**> Component*				
->NoAllocCommissions	2653	{*FIX 5.0 or later} Total number of commissions	Opt	C(if included in J)
-->AllocCommissionAmount	2654	{*FIX 5.0 or later}	C (NoAllocCommissions>1)	C(if included in J)

FIX tag	FIX tag #	Future - Valid values	Included with Allocation Instruction	Required in returned Allocation Report
		Total commission amount		
--> AllocCommissionAmountType	2655	{*FIX 5.0 or later} Supported values: 2 = Broker - The executing broker's commission. 3 = Clearing broker - The clearing broker's commission	C (NoAllocCommissions>1)	C(if included in J)
-->AllocCommissionBasis	2656	{*FIX 5.0 or later} Supported values: 1 = Per unit 2 = Percent 3 = Absolute - Recommeded	C (NoAllocCommissions>1)	C(if included in J)
*End <**AllocCommissionDataGrp**> Component*				
-> AllocAvPx	153	<price> Average price for this account. Used for tolerance validation of sell-side best-fit execution price generation Used for buy-side specificaiton of account-level average prices – can be used for give-ups via exchanges that support average price give-ups. Note: if 153 is in cluded in the allocation instruction for one entry it must be included for all entries.	Opt (buy-side specified account-level average prices)	Y
*<**MiscFeesGrp**> Component*				
->NoMiscFees	136	<number of fees>	Opt	C(if included in J)
-->MiscFeeAmt	137	<amount>	C (136>1)	C(if included in J)
-->MiscFeeCurr	138	<currency code> Note: The default for this is Currency (tag15) and if MiscFeeCurr is included it must be the same as Currency (tag15). Without an FX rate there would be no way of using in calculations.	opt	C(if included in J)
-->MiscFeeType	139	The Practices utilize these types to represent the industry standard OMGEO/SWIFT aggregations of fees: Supported values:	C (if 136 specified)	C(if included in J)

FIX tag	FIX tag #	Future - Valid values	Included with Allocation Instruction	Required in returned Allocation Report
		2 =Tax (TRAX/TTAX) 4 = Exchange (LOCL/FEES) 7 = Other (OTHR/MISC) 10 = Per Transaction (CHAR/BROK) Note: total fees are coded as 7		
*<**MiscFeesSubGrp**> Component*				
--> NoMiscFeeSubTypes	2633	<number of sub types> note: if MiscFeeSubType is specified for one fee it must be specified for all		Recmd
---> MiscFeeSubType	2634	<market specific fee code> Semi-human readable market specific fee code. See http://www.fixtradingcommunity.org/codelists#Misc_Fee_Sub_Types for latest code list.		C(if 2633>0)
---> MiscFeeSubTypeAmt	2635	<amount of specified MiscFee> Note: Sum of all subtype amount fields must equal MiscFeeAMt(137), if different these take precedence.		C(if 2633>0)
---> MiscFeeSubTypeDesc	2636	<Description> Optional human readable description of fee. **Note** that these are there for additional clarity if there is any confusion.		opt
*End <**MiscFeesSubGrp**> Component*				
-->MiscFeeBasis	891	Supported values: 0 = Absolute (default, recommended) 1 = Per unit 2 = Percentage	opt	C(if included in J)
*End <**MiscFeesGrp**> Component*				
*End <**AllocGrp**> Component*				

11.4 FIX 4.4 Allocation Instruction ACK message (buy-side or Clearing Firm)

FIX tag	FIX tag #	Valid values	Required
MessageType	35	= P	Y
AllocID	70	<id from Allocation Instruction>	Y
TradeDate	75	<date> (from Allocation Instruction)	N
TransactTime	60		Y
AllocStatus	87	Supported values: 0 = Accepted 1 = Reject 3 = Received not yet processed 6 =Pending (block matched, not allocated yet)	Y

11.5 FIX 4.4 AllocationReportACK(35=AT) message

FIX tag	FIX tag #	Valid values	Included
MessageType	35	AT	Y
AllocReportID	755	<id from AllocationReport>	Y
AllocID	70	<id from Allocation Instruction>	Y
TradeDate	75	<from Allocation Instruction>	N
TransactTime	60		Y
AllocStatus	87	Supported values: 0 =Aaccepted 1 = Reject (block level) 3 = Received not yet processed	Y

11.6 Confirmation(35=AK) message

FIX tag	FIX tag #	Confirmation Message Valid values	Required
Message			
MsgType	35	AK	Y
ConfirmID	664	<unique ID created by broker>	Y
ConfirmRefID	772	<id of replaced or canceled>	C (666=2)

FIX tag	FIX tag #	Confirmation Message Valid values	Required
ConfirmTransType	666	Supported values: 0 = New 2 = Cancel	Y
ConfirmType	773	Supported values: 2 = Confirmation	Y
LegalConfirm	650	Supported values: Y = Indicated legal confirmation when ConfirmTransType=0	C (666=0)
ConfirmStatus	665	Supported values: 4 = Confirmed) Note: “confirmed” means that this is the sell-side view.	Y
Parties Section			
*<**Parties**> Component*			
NoPartyIDs	453	Supported values: 3 – For Executing broker, order origination firm, and clearing firm. 4 – If Large Trader Reportable account is also included	Y
Executing Broker Instance			
->PartyID	448	<BIC code>	Y
->PartyIDSource	447	Supported values: B = BIC	Y
->PartyRole	452	Supported values: 1 = Executing firm	Y
->NoPartySubIDs	802	Supported values: 2	Y
--->PartySubID	523	<full legal name for executing broker>	Y
--->PartySubIDType	803	Supported values: 5 = Full legal name of firm	Y
--->PartySubID	523	<postal address for executing broker>	Y
--->PartySubIDType	803	Supported values: 6 = Postal address	Y
End Executing Broker Instance			

FIX tag	FIX tag #	Confirmation Message Valid values	Required
Order Origination Firm Instance			
->PartyID	448	<BIC code>	Y
->PartyIDSource	447	Supported values: B = BIC	Y
->PartyRole	452	Supported values: 13 =Order origination firm	Y
End Order Origination Firm Instance			
Clearing Firm Instance			
->PartyID	448	<BIC code>	Y
->PartyIDSource	447	Supported values: B = BIC	Y
->PartyRole	452	Supported values: 4 = Clearing firm	Y
End Clearing Firm Instance			
LargeTrader Reportable Account Instance			
->PartyID	448	Supported values: <LTI>	C
->PartyIDSource	447	**{*FIX 5.0 or later}** Supported values: D	C
->PartyRole	452	**{*FIX 5.0 or later}** Supported values: 52 = Large trader reportable account	C
LargeTrader Reportable Account Instance			
*End <**Parties**> Component*			
End Parties Section			

FIX tag	FIX tag #	Confirmation Message Valid values	Required
Trade Identification Section			
AllocID	70	<AllocID(70) from AllocationInstruction(35-J)> Note: if ConfirmTransType(666) = 2 (Cancel) then AllocID(70) depends upon why this AllocationInstruction(35-J)/ AllocTransType(71) = 2 (Cancel) was generated: - A response to AllocationInstruction(35-J)/ *AllocTransType(71)* = 2 (Cancel)has the AllocID(70) of the AllocationInstruction(35-J)/ *AllocTransType(71)* = 2 (Cancel) - A modification of a Confirmation(35=AK) by the sell-side has the AllocID(7)) of the current AllocationInstruction(35=J)/ AllocTransType(71) = 0 (New) or 1 (Replace) - A Confirmation [cancel)in response to an AllocationInstruction(35=J)/AllocTransType(71) = 1 (Replace) has the AllocID(70) of the AllocationInstruction(35=J)/ AllocTransType(71) = 1 (Replace) Note: if this value is not available because the allocation instruction was communicated in some other fashion than FIX use "N/A" for the value.	Y
IndividualAllocID	467	<transaction-id> From AllocationInstruction(35=J) / IndividualAllocID(467) **Note**: if this value is not available because it was not provided by the buy-side it is recommended that the sell-side generate a transaction-id for use by the buy-side. If this is not possible use "N/A".	Y
Text	58	<reason for cancellation>	C (666=2)
TransactTime	60	<time> Time this message was generated	Y
TradeDate	75	<date> (TradeDate(75) of the placement execution reports)	Y
<***Instrument***> *Component*			
Symbol	55	<symbol>	Y
SecurityID	48	<security-id>	Y

FIX tag	FIX tag #	Confirmation Message Valid values	Required
SecurityIDSource	22	Supported values: A = BBYK 5 = RIC	Y
CFIcode	461	Supported values: "FMXXXX"	N
SecurityType	167	Supported values: FUT OOF **{*FIX 5.0 or later}**	N
MaturityMonthYear	200	<YYYMM>	C(167=FUT)
MaturityDate	541	<YYYMMDD> (local market date) applies to the option not the underlying future	C(167=OOF)
StrikePrice	202	<price> (in exchange traded units)	C(167=OOF)
SecurityDesc	107	<text description of security)	N
*End <**Instrument**> Component*			
AllocQty	80	<quantity allocated to this account>	Y
QtyType	854	Supported values: 0 = Unit (shares, par) (default) 1 = contracts	N
Side	54	Supported values: 1 = Buy 2 = Sell	Y
Currency	15	<currency code> (trade currency) **Note**: all amounts in the AllocationInstruction(35=J) must be denominated in this currency. The only exception is that SettlemetnCurrency(120) and associated fields could be different currency.	Y
*<**CpctyConfGrp**> Component*			
NoCapacities	862	>=1	Y
->OrderCapacity	528	Supported values: A = Agency P = Principal M = Mixed **{*FIX 5.0 or later}**	Y
->Order capacity quantity	863	<Quantity> (executed in this capacity)	Y

FIX tag	FIX tag #	Confirmation Message Valid values	Required
*End <**CpctyConfGrp**> Component*			
Account Identification Section			
AllocAccount	79	<client account>	Y
AllocAcctIDSource	661	Supported values: 4 = OMGEO (AlertID) 99 = Other	Y
AllocAccountType	798	Supported values: 1 = Account is carried on customer side of books (default)	N
End Account Identification Section			
Financial Detail Section			
AvgPx	6	<booking price> (of the executions allocated to this account)	Y
PriceType	423	Supported values: 1 = percentage (e.g. Percent of par) 2 = per unit (default)	N
GrossTradeAmt	381	<amount> (Total amount traded (e.g. AllocQty (80) * (AvgPx (6) or AllocAvgPx(153))) expressed in trade currency) **Note**: if PriceType(423) is Percent of par, then this will be: AllocQty(80) * AvgPx(6) / 100.	C(OOF?)
NetMoney	118	<amount> (Total amount due as the result of the transaction (e.g. for Buy order - principal + commission + fees) reported in trade currency)	C(OOF?)
SettlDate	64	<YYYYMMDD>	Y
End Financial Detail Section			
Commissions Section			
*<**CommissionDataGrp**> Component*			
NoCommissions	2639	**{*FIX 5.0 or later}** <Total number of commissions>	C (if included in J)
-> CommissionAmount	2640	**{*FIX 5.0 or later}** <Total commission amount>	C (if included in J)

FIX tag	FIX tag #	Confirmation Message Valid values	Required
-> CommissionAmountTyp e	2641	**{*FIX 5.0 or later}** Supported values: 2 = Broker - The executing broker's commission. 3 = Clearing broker - The clearing broker's commission.	C (if included in J)
-> CommissionBasis	2642	**{*FIX 5.0 or later}** Supported values: 1 = Per unit 2 = Percent 3 = Absolute (recommeded)	C (if included in J)
*End <**CommissionDataGrp**> Component*			
Fees Section			
*<**MiscFeesGrp**> Component*			
NoMiscFees	136	<integer> (not included if there are no fees)	C (if included in J)
-> MiscFeeAmt	137	<amount>	C (136 >0)
-> MiscFeeCurr	138	<currency-code> Note: The default for this is Currency (tag15) and if MiscFeeCurr is included it must be the same as Currency (tag15). Without an FX rate there would be no way of using in calculations.	N
-> MiscFeeType	139	The Practices utilize these types to represent the industry standard OMGEO/SWIFT aggregations of fees: Supported values: 4 = Exchange (LOCL/FEES) 2 = Tax (TRAX/TTAX) 10 = Per Transaction (CHAR/BROK) 7 = Other (OTHR/MISC)	C (for each tag 137)
*<**MiscFeesSubGrp**> Component*			
-> NoMiscFeeSubTypes	2633	**{*FIX 5.0 or later}** <number of sub types> note: if MiscFeeSubType is specified for one fee it must be specified for all	Recmd

FIX tag	FIX tag #	Confirmation Message Valid values	Required
--> MiscFeeSubType	2634	**{*FIX 5.0 or later}** <market specific fee code> Semi-human readable market specific fee code. See http://www.fixtradingcommunity.org/codelists#Misc_Fee_Sub_Types for latest code list. Initial code list included in appendix.If	C(if 2633 > 0)
--> MiscFeeSubTypeAmt	2635	**{*FIX 5.0 or later}** <amount of specified MiscFee> Sum of all subtype amount must equal MiscFeeAMt(137) but if different these fields take precedence.	C(if 2633 > 0)
--> MiscFeeSubTypeDesc	2636	**{*FIX 5.0 or later}** Optional human readable description of fee. Note that these are there for additional clarity if there is any confusion. In the end there should be no need for these names.	opt
*End <**MiscFeesSubGrp**> Component*			
-> MiscFeeBasis	891	0 absolute (default)	N
*End <**MiscFeesGrp**> Component*			
Fees Section			

11.7 FIX 4.4 Confirmation(35=AK): Scenario – Confirmation Status - Confirmed

FIX tag	FIX tag #	Valid values	Required
MessageType	35	AK	Y
ConfirmID	664	<unique ID created by broker>	Y
ConfirmTransType	666	Supported values: 0 = New	Y
ConfirmType	773	Supported values: 2 = Confirmation	Y
ConfirmStatus	665	Supported values: 4 = Confirmed)	Y
LegalConfirm	650	Supported values: Y = Yes	Y
*<**Parties**> Component*			
NoPartyIDs	453	Supported values: 2	Y
->PartyRole	452	Supported values: 1 = Executing broker	Y
->PartyIDSource	447	Supported values: C = Generally accepted market participant identifier (e.g. NASD mnemonic)	Y
->PartyID	448	<brokerID> (NASD code)	Y
->PartyRole	452	Supported values: 4 = Clearing firm	Y
->PartyIDSource	447	Supported values: C = Generally accepted market participant identifier (e.g. NASD mnemonic)	Y
->PartyID	448	<brokerID> (NASD code)	Y
*End <**Parties**> Component*			
AllocID	70	<allocID from Allocation Instruction>	Y
IndividualAllocID	467	<individualAllocID(467) from Allocation Instruction>	Y
TransactTime	60	<time> Time this message was generated	Y
TradeDate	75	<YYYYMMDD> (date of allocation instruction)	Y
*<**Instrument**> Component*			
Symbol	55	(see instrument definition)	Y
SecurityID	48	(see instrument definition)	Y
SecurityIDSource	22	(see instrument definition)	Y
CFIcode	461	(see instrument definition)	N

FIX tag	FIX tag #	Valid values	Required
SecurityType	167	(see instrument definition)	N
MaturityMonthYear	200	<maturity date> (for standardized securities)	Y
*End <**Instrument**> Component*			
AllocQty	80	<quantity allocated to this account>	Y
Side	54	Supported values: 1 = Buy 2 = Sell	Y
AllocAccount	79	<client account>	Y
AvgPx	6	<booking price of the executions allocated to this account>	Y
SettlType	63	Supported values: 0 = Regular – Default if not specified 6 = Future Note: these have same intpretation for futures	Opt
SettlDate	64	<Settlement date>	Y

11.8 FIX 4.4 ConfirmationAck(35=AU) Message

FIX tag	FIX tag #	Valid values	Included
MessageType	35	AU	Y
ConfirmID	664	<unique ID for this message>	Y
TradeDate	75	<YYYYMMDD> (date of allocation instruction)	Y
TransactTime	60	<time> (Time this message was generated)	Y
AffirmStatus	940	Supported values: 2 = Rejected, 3 = Affirmed	Y

12 FIX 4.4 Message Formats – Options on Futures

12.1 Futures Order Messages – Options on Futures – Outright -Differences

12.1.1 New Order Message - Options on Futures – Outright - Differences

FIX tag	FIX tag #	Included New	Included Replace	Included Cancel	Option on Future (differences) - Valid values
<***Instrument***> *Component*					
Symbol	55	Y	Y	Y	<optionOnfuture-symbol>
SecurityType	167	Y	Y	Y	Supported values: OOF = Options on futures
CFIcode	461	Y	Y	Y	Supported values: "O"<"P"/"C">"XFXX"
MaturityDate	541	Y	Y	Y	<YYYYMMDD> (local market date of option) (*1)
StrikePrice	202	Y	Y	Y	<price>(*2)
ExDestination	100				Not included for options
<***Instrument***> *Component*					

Notes:

(*1) MaturityDate applies to the Option not the underlying Future.

(*2) buy-side sends the strike price in the exchange traded units even though prices are expected to be returned in the basic traded units (e.g. the traded units for corn futures is USD and the strike price will be sent in that form but LastPx is expected to be returned in USD). Note that the strike-price in the BBYK is also in exchange traded units.

12.1.2 Execution Report Message – Options on Futures – Differences

FIX tag	FIX tag #	Required	Notes
*<**Instrument**> Component*			
MaturityDate	541	Y	
StrikePrice	202	Y	
*End <**Instrument**> Component*			

12.2 Futures Order Messages MLEG – Options on Futures – Differences

12.2.1 Options on Futures New Order MLEG – Differences

FIX tag	FIX tag #	Included New	Included Replace	Included Cancel	Option on Future (differences) - Valid values
*<**InstrumentLeg**> Component*					
LegSymbol	600	Y	Y	Y	< optionOnfuture-symbol>
LegCFIcode	608	Y	Y	Y	Supported values: “O”<”P”/”C”>”XFXX”, (option, put/call, future)
LegSecurityType	609	Y	Y	Y	Supported values: OOF = Options on futures
LegMaturityDate	611	Y	Y	Y	<YYYYMMDD> (local market date of option) (*1)
LegStrikePrice	612	Y	Y	Y	<price> (*2)
*End <**InstrumentLeg**> Component*					

Notes:

(*1) MaturityDate applies to the Option not the underlying Future.
(*2) buy-side sends the strike price in the exchange traded units even though prices are expected to be returned in the basic traded units (e.g. the traded units for corn futures is USD and the strike price will be sent in that form but LastPx is expected to be returned in USD). Note that the strike-price in the BBYK is also in exchange traded units.

12.2.2 Execution Report Message MLEG – Options on Futures – Differences

FIX tag	FIX tag #	Required	Notes
*<**InstrumentLeg**> Component*			
LegMaturityDate	611	Y	
LegStrikePrice	612	Y	Same as order
*End <**InstrumentLeg**> Component*			

12.3 Allocation - Options on Futures – Differences

FIX tag	FIX tag #	Option on Future (differences) - Valid values	Included with Allocation Instruction	Required in returned Allocation Report
*<**Instrument**> Component*				
Symbol	55	<optionOnfuture-symbol>	Y	Y
CFIcode	461	Supported value: “O”<”P”/”C”>”XFXX”	Y	Y
SecurityType	167	Supported values: OOF = Options on futures	Y	Y
MaturityDate	541	<YYYMMDD> (local market date) of option	Y	Y
StrikePrice	202	<price>	Y	Y
*End <**Instrument**> Component*				

13 Appendices

13.1 16.1 Example FIX Messages (tbd)

tbd