Cboe Europe
FIX Specification

Version 3.33

09 February 2024

Contents

1 Overview

This document describes Cboe interpretation and implementation of the FIX 4.2 specification. Cboe uses a subset of the FIX 4.2 protocol for order entry and drop copies. It is assumed that the reader is familiar with the FIX 4.2 protocol as described by the FIX Protocol Organisation.

Cboe operates separate pairs of integrated and dark order books on a UK venue distinguished by the identifiers BXE and CXE. Cboe also operates a pair of integrated and dark order books on a NL venue distinguished by the identifier DXE.

The functionality described in this document applies equally to BXE, CXE and DXE FIX sessions, unless specified otherwise.

1.1 Hours of Operation

Refer to the Cboe website for hours of operation.

All orders are live upon acceptance by Cboe. Orders are rejected if they are received outside the hours Cboe is available for trading. Cboe does not support maintaining orders for multiple days (GTC orders). All open orders are canceled on close of the market. Participants will receive an execution report for every open order with *ExecType* (150) = 4 (Canceled).

1.2 Timestamps

All FIX timestamps are GMT as per the FIX standard. Participants are expected to synchronise their clocks with an external time source.

1.3 Symbology

Cboe accepts three symbologies: Uniform Symbology, RIC and ISIN. Different symbologies may be used on different orders, but it is recommended that Participants use the same symbology for all orders.

If using Uniform Symbology to identify a stock, the Participant:

- **must** set *Symbol* (55) to the Uniform Symbology symbol;
- *may optionally* set the *SecurityExchange* (207); and,
- *may optionally* set the *Currency* (15).

If using ISIN to identify a stock, the Participant:

- **must** set *IDSource* (22) to ISIN (4);
- **must** set *SecurityID* (48) to the ISIN;
- **must** set *SecurityExchange* (207) to note the market in which the ISIN trades;
- **must** set the *Currency* (15) field to identify the currency in which the stock is traded; and,
- *may optionally* set the *Symbol* (55).

If using RIC to identify a stock, the Participant:

- **must** set *IDSource* (22) to RIC (5);
- **must** set *SecurityID* (48) to the RIC;
- *may optionally* set the *SecurityExchange* (207);
- *may optionally* set the *Currency* (15) field; and,
- *may optionally* set the *Symbol* (55).

If using ISIN or RIC to identify a stock in *SecurityID* (48), and opting to also send *Symbol* (55), the *Symbol* (55) may be specified as the Uniform Symbology symbol, the *SecurityID* (48), the RIC or the Ticker code.

A RIC in either *SecurityID* (48) or *Symbol* (55), may be supplied as either a Cboe or primary market RIC listed in the Cboe symbol reference files for that day. Where a Cboe RIC is used, it must be applicable for the venue (UK or EU) the order or TCR is being submitted to, otherwise it will be rejected i.e. .CHI for CXE, .BS for BXE or .DXE for DXE.

When specifying an optional value as noted above, the value specified must match the value in the Cboe symbol database. Otherwise, the order will be rejected.

Execution reports will always respond with the same symbology as was sent in the corresponding New Order Single message.

1.4 Tick Sizes

The minimum price increment, or tick size, is generally the same as that on the primary market. Tick size is subject to change. Orders entered which violate the tick size will be rejected. Midpoint peg orders are not tick size validated and may execute at a price that is one-half the tick size. Where the mathematical mid price has more than four decimal places then the match mid price will be rounded up to four decimal places.

Trade Capture Reports do not need the reported or settlement price to be on a tick size boundary. Should the price specified exceed seven decimal places, it will be truncated to such.

1.5 Hidden Orders

Cboe allows Participants to place hidden orders which are not represented on its market data feed. Hidden orders include pegged orders and orders which have the *DisplayIndicator* (9479) = `I` (Invisible). MiFID regulations require that orders with a notional value less than the Large In Scale (LIS) *must* be displayed unless routed to the Cboe Dark Pool.[1]

Cboe will reject any order submitted which violates this regulation. Where no LIS value is specified in the MiFID regulations, Participants may only submit hidden orders with a notional value greater than the value specified by Cboe in its reference data unless routed to the Cboe Dark Pool. Cboe will reject any order submitted which violates this.

Notional value is calculated differently depending upon order type:

Limit:	*Price* (44) × *OrderQty* (38)
Buy primary peg, sell market peg:	(PBBO bid + *PegDifference* (211)) × *OrderQty* (38)
Sell primary peg, buy market peg:	(PBBO ask + *PegDifference* (211)) × *OrderQty* (38)
Midpoint peg:	PBBO midpoint × *OrderQty* (38)

More information on the MiFID ESMA database can be found at the ESMA MiFID website.

1.6 Cboe Dark Pool

The Cboe Dark Pool is a separate book which allows matching of dark liquidity based at a midpoint reference price. Separate Cboe Dark Pools are operated for BXE, CXE and DXE ports. Orders placed into the Cboe Dark Pool do not need to be Large In Scale (LIS). Cboe Dark Pool orders only interact with other Cboe Dark Pool orders. Quotes for Cboe Dark Pool orders are not represented on any market data feed. Matches in the Cboe Dark Pool are represented as trades on the Cboe market data feeds and may be differentiated.

Orders destined for the Cboe Dark Pool must be midpoint peg orders[2]. The *RoutingInst* (9303) must be set to one of the following:

`BD`: Routes only to the Cboe Dark Pool.

[1] Field *RoutingInst* (9303) = BD or BA.
[2] *ExecInst* (18) = M.

BA: Routes to the Cboe Dark Pool if midpoint and not LIS, or to the integrated book otherwise. Non-midpoint peg orders may also be sent with BA and will always route to the integrated book.

BH: Routes initially to the Cboe Dark Pool, with any residual routed to the periodic auction book.

A limit price may be specified on a Cboe Dark Pool order using the *Price* (44) field. If set, execution still only occurs at the midpoint. When the midpoint is a more aggressive price than the limit price, the order will not be available for execution.

1.7 Fee Codes Returned on Execution Reports

The Fee Code returned on an execution report indicates which category of fee is applicable to the trade. Details on the fee codes returned (including the rate charged and description of the category of trade) are available at:

BXE: http://www.bats.com/europe/equities/participation/fee_schedule/bxe/

CXE: http://www.bats.com/europe/equities/participation/fee_schedule/cxe/

DXE: http://www.bats.com/europe/equities/participation/fee_schedule/dxe/

TRF: http://www.bats.com/europe/equities/participation/fee_schedule/trf/

All fee codes map directly to a tariff on the Cboe Europe trading price list, available at http://www.bats.com/europe/equities/support/price_lists/ under Trading Price List. Participants should program their systems to read, validate and pass along this field in order to avoid making software changes to their system when the Cboe fee schedule changes.

1.8 Service Bureau Configuration

Service Bureaus require special configuration. *OnBehalfOfCompID* (115) must be set for Order, Cancel, and Cancel/Replace messages sent to Cboe. Orders with an unknown *OnBehalfOfCompID* (115) will be rejected. *ClOrdID* (11) values are required to be unique only within a given *OnBehalfOfCompID* (115). Execution Reports, Cancel Rejects and trade capture related messages sent by Cboe will have the *DeliverToCompID* (128) set. **Orders must be canceled or replaced using the same OnBehalfOfCompID (115) as was sent on the original order.**

1.9 Order Price Collars

Cboe uses market data from the primary exchange for each symbol. This primary best bid and offer (PBBO) is used to create an order price collar. Executions will not be allowed to occur a set percentage above the PBBO best ask quote or a set percentage below the PBBO best bid quote. If an order matches against a resting order, but is outside the price collar, the incoming order will be rejected with reason "price exceeds cross range".

If the primary exchange is not in continuous trading (e.g., is in auction or has closed), the order price collar will be a percentage of the last regular on-book trade price on the primary exchange. If there has not yet been a trade on the primary exchange today, then no collar is in effect.

The percentage value(s) used for order price collars is/are defined in the Participant Manual.

1.10 Reserve (Iceberg) Orders

Cboe allows the use of *MaxFloor* (111) for entering reserve (iceberg) orders. On reload, the Cboe market data feeds show a new OrderID in order to hide the fact that the order is a reserve order.

A port- or firm-level attribute may be enabled which will cause an execution report with *ExecType* (150) = D (Restated) to be sent each time an order is reloaded, specifying the new OrderID as will be seen on the market data feeds in *SecondaryOrderID* (198).

The *OrderID* (37) remains constant for the lifetime of the order. Only the *SecondaryOrderID* (198) will change.

Here is a timeline showing an order and its OrderIDs as seen by the Participant and the market data feeds.

#	OrderID (37)	SecondaryOrderID (198)	Displayed Order ID	Notes
1	1C3M03000008	—	1C3M03000008	Initial order entry.
2	1C3M03000008	1C3M0300000E	1C3M0300000E	Order reloaded.
3	1C3M03000008	1C3M0300000Q	1C3M0300000Q	Order reloaded again.

1.11 Peg Order Pricing

Pegged orders are priced using the primary best bid and offer (PBBO). If the primary exchange is not in continuous trading (e.g., is in auction or is not currently open), resting pegged orders are temporarily suspended from trading. They will become executable again as soon as the primary exchange is back in continuous trading. Newly entered pegged orders will go into a suspended state immediately.

1.12 Interbook Order Types

Dark Sweep and Lit Sweep are interbook order types designed to be a simple solution for multi-book access on the UK venue only between BXE and CXE.

Orders marked *RoutingInst* (9303) = U will be eligible for dark book sweeping. Dark Sweep orders are first sent to the dark book associated with the book of entry before moving on to the other book if there is a partial execution.

Orders marked *RoutingInst* (9303) = W will be eligible for lit book sweeping. Lit Sweep orders are sent to the lit book with the best price available. Orders work between books and associated price levels. In the event that the same price is available on both order books, the order will be sent to the book of entry first.

Orders marked *RoutingInst* (9303) = X will be eligible for sequential lit book sweeping. The order executes on the book of entry up to the order limit, any residual is sent to the other book and continues to be filled up to the limit. Any un-filled portion is cancelled back or posted on the book of entry.

1.13 Dark Lit Orders

Orders marked *RoutingInst* (9303) = u[3] will match on both the lit and dark books on a price/priority basis. Any residual quantity will rest on the lit book (for limit orders) or be cancelled (immediate or cancel orders). There may be limitations as to the markets supported; refer to the Participant Manual or contact the Cboe Trade Desk for details.

1.14 Dark Periodic Orders

Orders marked *RoutingInst* (9303) = BH[4] will match first on the dark book and then on the periodic auction book. All dark periodic orders will be midpoint peg. IOC orders will be valid as good for auction with minimum quantities honoured in both books; refer to the Participant Manual or contact the Cboe Trade Desk for details.

1.15 Account Field

This field can carry two pieces of information.

Firstly a Central Counterparty (CCP) Account Type prefix.

[3] See *RoutingInst* (9303) = u for details.
[4] See *RoutingInst* (9303) = BH for details.

If this field begins with **H:**, allocate to the house account at the CCP.

If this field begins with **C:**, allocate to the client account at the CCP.

As of the Q2 2017 release, when an Account Type Prefix is not supplied, the *OrderCapacity* (47) or *OrderCapacity* (528) will no longer be used to determine which CCP account to use. The CCP account can be defaulted at a port level if required. Non-prefixed or absent accounts would be allocated to *House* account. All CCPs support this feature.

Secondly is a trading account name/number.

With configuration, this can passed to CCPs which support this feature (LCH.Clearnet currently does not). This part must be 16 characters or less. The trading account is configurably available via Drop.

1.16 Trading Venue Transaction Identification Code (TVTIC)

A TVTIC is needed when using Cboe Transaction Reporting. TVTIC can be derived from *TradeID* (1003) as returned in a Trade Capture Report or *ExecID* (17) as returned in an Execution Report. Please refer to the Cboe Participant Manual on how to construct TVTIC.

1.17 New Trade Type Flags (Effective 1 Jan 2024)

Effective 1 Jan 2024, Cboe will support new Trade Type values, resulting from ESMA's review of RTS 1 (Equity Transparency) and RTS 2 (Non-equity Transparency).

Effective 29 Apr 2024, Cboe will additionally support CLSE on BXE and CXE, resulting from FCA's PS 23/4.

Currently Cboe supports the following:

TrdType (828) = 0 = Regular Trade, and,

SecondaryTrdType (855) = 64 = Benchmark

Cboe will introduce support for the following new input Trade Type values:

`PORT`: Portfolio Trade = '50'

`CONT`: Contingent Transaction = '65'

`CLSE`: Benchmark Transactions executed at the market close price = '67' (from 29 Apr 2024)

`GIVE`: Give-up / Give-in trade (Request for Market data (RFMD) Give-up) = '61'

The following values will also be introduced, effective 1 Jan 2024, solely to be used in *TrdType* (828), and soley for RTS 2 non-equities:

TrdType (828) = 2 = Exchange For Physicals

TrdType (828) = 65 = Package Trade which are not Exchange for Physicals. Use 2 if it's Exchange for Physicals.

Cboe will expand Trade Type values support by adding a new FIX tag *TertiaryTrdType* (2896). This allows up to three trade type flags to be specified on single report using a combination of *TrdType* (828), *SecondaryTrdType* (855), and *TertiaryTrdType* (2896). *TertiaryTrdType* (2896) should not be used when only one additional trade type needs to be assigned.

The following Secondary Trade Type values are hence supported:

`PORT`: Portfolio Trade = '50'

`CONT`: Contingent Transaction = '65'

GIVE: Give-up / Give-in trade (Request for Market data (RFMD) Give-up) = '61'

BENC: Benchmark = '64'

CLSE: Benchmark Transactions executed at the market close price = '67' (from 29 Apr 2024)

In most cases, TCRs can have a valid combination of only one or two of the secondary trade types listed above.

BENC cannot be combined with CLSE. They are mutually exclusive, and only one of them can be used for a particular trade report.

When there is just one secondary trade type, specify it in *SecondaryTrdType* (855).

To specify two secondary trade types, use both *SecondaryTrdType* (855) and *TertiaryTrdType* (2896).

In both cases, *TrdType* (828) should continue to be set to '0' (Regular Trade).

There is one exception case where a trade is a Benchmark, Portfolio Trade as well as a Contingent Transaction (BENC, PORT, CONT). In this case, *TrdType* (828) should be set to 50 (Portfolio Trade). *SecondaryTrdType* (855) and *TertiaryTrdType* (2896) can then be used to set the other two values of BENC and CONT.

Here's the list of valid combinations of 2 or more trade types:

BENC & PORT

BENC & CONT

CONT & PORT

PORT & BENC & CONT

BENC & GIVE

CLSE & PORT (from 29 Apr 2024)

The order they are specified does not matter to Cboe, but the order in which they are returned in Trade Capture Report message will be as above.

For example, in the first case of BENC & PORT, it's acceptable to specify either

SecondaryTrdType (855) = 64 (BENC) and *TertiaryTrdType* (2896) = 50 (PORT), or

SecondaryTrdType (855) = 50 (PORT) and *TertiaryTrdType* (2896) = 64 (BENC).

In either case, Cboe will return *SecondaryTrdType* (855) = 64 (BENC) and *TertiaryTrdType* (2896) = 50 (PORT).

2 Cboe Specific FIX Fields

The following FIX fields are specific to Cboe:

Tag	Name	Description
6655	*CorrectedSize*	UCC trade correction message, this holds the corrected size.
7570	*RptTime*	Indicates the time at which a deferred trade report will be automatically published. Where *RptTime* falls outside of the systems operating time, the report will be published during operating hours on the next trading day. When no deferral is requested, or when the trade does not qualify for a deferral, any time returned will match *TransactTime* (60). Microsecond level precision.
7692	*RiskReset*	For use by customers to reset or release clearing firm, symbol or *CustomGroupID* (7699) level lockout conditions resulting from self-imposed lockouts issued via `Purge Request messages`. **Single Character Values**: `S` = Symbol level lockout reset `F` = Clearing firm level lockout reset `C` = CustomGroupID lockout reset Values may be combined together to allow for resets of multiple self-imposed lockouts in a single message. For example, FS, SC, FC, and SFC are all acceptable values. If orders have been locked out at any level, inbound orders for the locked *Symbol* (55), clearing firm, or *CustomGroupID* (7699) will be rejected until this field is filled with the appropriate value on a `New Order Single message`.
7698	*CustomGroupIDCnt*	Number of custom group IDs. Must be between 1 and 10.
7699	*CustomGroupID*	Custom identifier for a group of orders.
7772	*CentralCounterparty*	Only present on trades. The CCP handling the trade: EMCF = Cboe Clear LCHL = LCH Ltd (London) LCHS = LCH SA (Paris) XCLR = SIX x-clear NONE = Clearing Suppressed Returned on trades if the participant has selected a Preferred CCP. The FIX port can be configured to always return this optional field.

7928	*PreventParticipant Match*	Participant Trade Prevention: 3 characters (not space separated): 1st character - PTP Modifier: `N` = Cancel Newest `O` = Cancel Oldest `B` = Cancel Both `D` = Decrement Larger[5] / Cancel Smaller `d` = Same as `D` above, but only decrement *LeavesQty* (151). Do not restate *OrderQty* (38). 2nd character - Unique ID Level: `N` = Do not prevent (Default value if not specified) `F` = Prevent Match at Participant Level `M` = Prevent Match at Trading Firm Level `P` = Prevent Match at Port Owner Level 3rd character - Trading Group ID (optional): Member specified alphanumeric value 0-9, A-Z, or a-z. The Unique ID level (character 2) of both orders must match to prevent a trade. If specified on both orders, Trading Group ID (character 3) must match to prevent a trade. The PTP Modifier (character 1) of the inbound order will be honored, except that if the inbound order specified Decrement and the resting order does not, and the resting order is larger, then both orders will be canceled. This exception is to protect the order entry software for the resting order from receiving an unexpected restatement message.

[5] Users of PTP modifier `D` must be prepared to receive a FIX Restatement execution report (*ExecType* (150) = `D`) that includes both *OrderQty* (38) and *LeavesQty* (151).

9303	*RoutingInst*	Up-to 4 characters (not space separated): `B` = Cboe Only **(default)** `P` = Cboe Only — Post Only (will reject rather than remove visible liquidity) `U` = Dark Sweep (interbook) `u` = Dark Lit Sweep (best-price) `W` = Lit Sweep (interbook, best-price) `X` = Lit Sweep (interbook, sequential) `BD` = Cboe Dark Book Only (hidden midpoint peg orders only) `BA` = Cboe Automatic Dark Routed (routes to Cboe Integrated Book if order is Large In Scale (LIS) or is not a midpoint order, otherwise routes midpoint non-LIS orders to Cboe Dark Book) `BP` = Cboe Periodic Auction Book `BH` = Dark Periodic Sweep (midpoint peg orders) `BU` = Cboe Closing Cross `BL` = Periodic Lit Sweep `BM` = Dark Periodic Lit Sweep An optional 3^{rd} character can be used to specify Price Improvement for Cboe Periodic Auction Book (`BP`) limit orders only: `N` = None (implied if 3^{rd} char is not specified) `L` = Limit (1-tick inside the limit price) `T` = Far-touch (least aggressive of 1-tick inside the EBBO far-touch or limit price) `B` = Both (least aggressive of 1-tick inside far-touch or 1-tick inside limit) An optional 4^{th} character can be used to specify Dark-on-Expiry for Periodic Auction Book and Dark Periodic Sweep orders. Only applicable when *TimeInForce* (59) = `8` (Good-For-Auction) or *TimeInForce* (59) = `3` (AOC) and the order is deemed marketable (ignored for other order types or *TimeInForce* values): `N` = None (implied if 4^{th} char is not specified) `D` = Dark-on-Expiry Post Only does not mix with *TimeInForce* (59) = `3` (IOC). If a *RoutingInst* is not specified a default value of `B` is implied (Cboe Only).
9416	*ExtExecInst*	`N` = None `G` = All Or None The all or none instruction applies to periodic auctions only.
9479	*DisplayIndicator*	`X` = Displayed Order (default) `I` = Invisible Invisible orders must meet the MiFID requirements for Large in Scale (LIS) unless routed to the Cboe Dark Book.
9617	*ModifySequence*	**Drop only.** Base 36 number of times order has been replaced.
9619	*CancelOrigOnReject*	`N` = Leave original order alone (default) `Y` = Cancel original order if replacement fails Default may be configured per port.
9620	*CorrectedPrice*	UCC trade correction message, this holds the corrected price.

9688	*OrigCompID*	**Drop only.** *TargetCompID* (56) of original FIX execution report. Drop port must be configured to send this optional field.
9689	*OrigSubID*	**Drop only.** *TargetSubID* (57) of original FIX execution report. Drop port must be configured to send this optional field.
9730	*TradeLiquidity Indicator*	Only present on trades. `A` = Added Liquidity `R` = Removed Liquidity `AD` = Added Liquidity for the Cboe Dark Pool `RD` = Removed Liquidity from the Cboe Dark Pool `RT` = Removed Liquidity from the Cboe Dark Pool by IOC order `AI` = Added Hidden Liquidity that was price improved `AK` = Added Liquidity from the hidden (reserve) portion of an iceberg order `AS` = Added Liquidity A-LPS BBO Setter `RS` = Removed Liquidity from A-LPS BBO Setter `X` = Routed to Another Market `C` = Auction `CC` = Cboe Closing Cross `P` = Periodic Auction `S` = Self Match (opt-in) `SD` = Self Match from the Cboe Dark Pool (opt-in) To allow for future expansion of this field, please ignore values with an unknown character in the 2nd position.
9882	*FeeCode*	Specific fee code associated with the trade. See the Fee Schedule for the respective market for possible values.

3 FIX Session Protocol

Cboe uses the FIX 4.2 session protocol. FIX 4.4 is possible should the firm dedicate the port for trade reporting purposes. The Participant will be provided with a *SenderCompID* (49) and *SenderSubID* (50) that must be sent on every message. The *TargetCompID* (56) for all messages the Participant sends will be `BATS` on BXE sessions, `CHIX` on CXE sessions or `CEUX` on DXE sessions. The *TargetSubID* (57) is `TEST` for the Cboe test system and `PROD` for the Cboe production system. All messages the Participant receives will have the sender and target fields swapped, as per the FIX specification.

The following session messages are supported in both directions:

Message	Type	Comment
Logon	`A`	Begin session (or resume a broken session)
Heartbeat	0	
Test Request	1	
Resend Request	2	
Reject	3	Malformed message or improper session level handling
Sequence Reset	4	Both Gap Fill (*GapFillFlag* (123) = `Y`) and Reset
Logout	5	used to gracefully close session

3.1 Sequence Numbers

Sequence numbers, both inbound and outbound, will be reset to one each night during the down time.

Messages are processed in sequence order. Behind sequence messages (other than Sequence Reset — Reset) cause immediate logout. Ahead of sequence messages (other than a Resend Request) trigger a message recovery via a Resend Request.

3.2 Logon

The logon must be the first message sent by the Participant after the TCP connection is established. *EncryptMethod* (98) is ignored (FIX level encryption is not supported).

The IP address of the Participant, the *SenderCompID* (49), *SenderSubID* (50), *TargetCompID* (56) (`BATS`, `CHIX`, or `CEUX`) and *TargetSubID* (57) (`TEST` or `PROD`) will be validated. If validation fails, the connection will be dropped without a reject (to avoid corrupting the Participant's sequence in the case that the Participant merely mistakenly connected to the wrong port).

If the connection is unexpectedly broken, upon reconnection, the Participant may receive a login reply with a sequence number greater than expected. This means that in-flight messages were missed (likely important execution reports). The Participant should issue a Resend request to retrieve the missed messages.

Similarly, Cboe will issue a Resend Request to the Participant for messages that it missed. The Participant may wish to send gap fill messages in place of new orders to avoid submission of potentially stale orders.

HeartBtInt (108) must be specified by the Participant in the Logon message. This value will be clamped between five and 300 seconds and returned in the Logon reply message. We recommend using as low a value as the reliability and latency of your telecommunications channel will allow.

3.3 Heartbeat

A Heartbeat message should be sent if the agreed upon *HeartBtInt* (108) has elapsed since the last message sent. If any message has been sent during the preceding *HeartBtInt* (108), a Heartbeat message need not be sent.

3.4 Test Request

If *HeartBtInt* + 1 seconds have elapsed since the last message received, a Test Request should be issued. If another *HeartBtInt* + 1 seconds go by without receiving a message, the TCP connection should be dropped. This ensures that a broken TCP connection will be detected even if the TCP stack doesn't notice (this has been observed to happen in WAN environments, particularly when a VPN is involved).

3.5 Resend Request

A Resend Request message should be processed even if it is received ahead of sequence. Only after resending the requested range (all marked *PossDup* (43) = `Y`), including any gap fills) should Resend Request be issued in the opposite direction.

As discussed in the FIX 4.2 specification, it is possible to send an open or closed sequence range in a Resend Request (an open range uses sequence zero as the *EndSeqNo* (16)). Cboe will honor either type of request, but will always issue Resend Requests with a closed sequence range.

3.6 Reject

Session level rejects (*MsgType* (35) = 3) are used to indicate violations of the session protocol, or missing (or mangled) fields. These are to be expected during development and certification while the Participant's systems are being adapted for Cboe, but should be extremely rare in production. Application layer rejects (like Order Reject, Cancel Reject, Trade Capture Reject) are normal and should be handled separately. See FIX Application Messages - Cboe to Participant (p. 36) for more details.

3.7 Sequence Reset

Sequence Reset — Gap Fill messages (*GapFillFlag* (123) = `Y`) must be received in sequence. Any messages (including Gap Fills) sent in response to a Resend Request should have *PossDup* (43) = `Y`.

Sequence Reset — Reset (*GapFillFlag* (123) ≠ `Y`) is used only as a last resort, and always by human intervention, to allow an otherwise hopelessly confused session to be resumed. In these cases, all chances at automatic message recovery are lost.

3.8 Logout

Either side may issue a logout to gracefully close the session. The side that issues the logout should process messages normally until it sees the logout reply, and then break the TCP connection. Cboe will typically only request logout after the scheduled end of FIX session.

4 Standard FIX Message Header and Trailer

4.1 Header

Tag	Name	Description
8	*BeginString*	`FIX.4.2` or `FIX.4.4` Must be the first field in the message. FIX 4.4 only available if the port is dedicated for trade reporting purposes.
9	*BodyLength*	Length of message following *BodyLength* field up to and including the delimiter preceding the *CheckSum* (10) field. Must be the second field in the message.
35	*MsgType*	Must be the third field in the message.
34	*MsgSeqNum*	Sequential sequence number for the session.
43	*PossDupFlag*	Indicates a message resent from the admin level (has a duplicate sequence number). Defaults to `N`.
49	*SenderCompID*	ID of sender. Assigned by Cboe for messages sent to Cboe. (*TargetCompID* (56) for messages from Cboe.)
52	*SendingTime*	GMT date and time that message was sent. Microsecond level resolution.
56	*TargetCompID*	ID of destination. `BATS` for messages sent to BXE ports. `CHIX` for messages sent to CXE ports. `CEUX` for messages sent to DXE ports. (*SenderCompID* (49) for messages from Cboe.)
57	*TargetSubID*	Sub ID of destination. `TEST` for messages sent to the Cboe test system. `PROD` for messages sent to the Cboe production system. (*SenderSubID* (50) for messages from Cboe.)
97	*PossResend*	Possible resend flag. Cboe has special handling for the *PossResend* for New Order Single messages. See New Order — Single below.
115	*OnBehalfOfCompID*	Service bureau use. Identifies end-client on messages **to** Cboe. Must be identifier known to Cboe. May be used by non-service bureau to specify which clearing arrangement to use if multiple are configured.
116	*OnBehalfOfSubID*	End-client sub identifier. Four characters, alphanumeric, otherwise not validated. Recorded and returned in *DeliverToSubID* (129). Available via Drop.
122	*OrigSendingTime*	For messages with *PossDupFlag* (43) = `Y`, indicates time that message was first sent. Microsecond level resolution.
128	*DeliverToCompID*	Service bureau use. Identifies end-client on message **from** Cboe. Must be Cboe approved identifier.
129	*DeliverToSubID*	Returns *OnBehalfOfSubID* (116) optionally sent by client.

Note: *OnBehalfOfSubID* (116) and *DeliverToSubID* (128) are currently only applicable to order-related messages, and are unimplemented for Trade Capture Reports.

4.2 Trailer

Tag	Name	Description
10	*CheckSum*	Modulo 256 checksum of all characters in the message up to and including the delimiter preceding the *CheckSum* field. Three digits with leading zeroes if necessary.

5 FIX Application Messages — Participant to Cboe

5.1 New Order — Single

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = D
97	*PossResend*	N = indicates a new order (default) Y = indicates an application level resend and **is not supported** For reasons of economy, Cboe does not track (in primary storage), the *ClOrdID* (11) values of orders that are no longer live. For reasons of performance, Cboe does not access secondary storage to enforce unique *ClOrdID* (11) values against orders that are no longer live. Without full duplicate *ClOrdID* (11) value enforcement, it is not possible to safely implement the full behavior specified in the FIX 4.2 protocol for *PossResend* = Y. To remain economical, fast, *and* safe, all New Order — Single and Trade Capture Report messages with *PossResend* = Y will be simply ignored.
1	*Account*	Optional. Returned on execution reports associated with this order. 16 characters or less (ASCII 33–126). **H:** and **C:** prefix can be used to specify which CCP Account to use. **If configured by Cboe**: *House* or *Client* CCP account can be defaulted. *OrderCapacity* (47) is no longer used to determine which CCP account to use. The value supplied can be passed to the CCP and made available on the Drop feed.
11	*ClOrdID*	Day-unique ID chosen by client. 20 characters or less. Characters in ASCII range 33–126 are allowed, except for comma, semicolon, and pipe. If the *ClOrdID* matches a live order, it will be rejected as duplicate (unless *PossResend* (97) = Y; see above). **Note: Cboe only enforces the uniqueness of ClOrdID values among currently live orders. However, we strongly recommend that you keep your ClOrdID values day unique.**
15	*Currency*	Required if *IDSource* (22) = 4 (ISIN). If *Currency* (15) is included when other symbology is used, it must match the currency expected by Cboe for the given symbol.

18	*ExecInst*	**Single value only (with no trailing space).** P = Market Peg (peg buy to PBBO offer, peg sell to PBBO bid) R = Primary Peg (peg buy to PBBO bid, peg sell to PBBO offer) M = Midpoint (peg to PBBO midpoint) L = Alternate Midpoint (less aggressive of midpoint and 1 tick inside PBBO) for Periodic Auction Orders:[6] M = Midpoint (peg to Cboe EBBO midpoint) G = Guarded Midpoint (peg to Cboe EBBO midpoint but suspend order if primary market quote becomes one-sided or disappears) R = Near-touch peg (peg buy to EBBO bid, peg sell to EBBO offer) with offset specified in *PegDifference* (211). For Periodic Auction orders, this offset can be aggressive. **NOTE: Values L and G differ in meaning from standard FIX 4.2.** All other values are ignored. Midpoint pegged orders (M and L) are implicitly hidden (*DisplayIndicator* (9479) = I). Rejected when used for Dark/Lit sweep orders.
22	*IDSource*	Values supported by Cboe: 4 = ISIN 5 = RIC Required if *Symbol* (55) is not set.
38	*OrderQty*	Number of shares for the order. System-wide limit is 99,999,999 shares.
40	*OrdType*	1 = Market 2 = Limit P = Pegged During continuous trading Market (1) implies *TimeInForce* (59) = 3 (IOC). Pegged requires *ExecInst* (18) = L, M, P, or R. Pegged orders may not be routable.
44	*Price*	Limit price.
47	*OrderCapacity*	A = Agency (maps to 'AOTC') P = Principal (maps to 'DEAL') R = Riskless (maps to 'MTCH')
48	*SecurityID*	ISIN, or RIC if *IDSource* (22) is set.
54	*Side*	1 = Buy 2 = Sell 5 = Sell Short 6 = Sell Short Exempt H = Sell Undisclosed
55	*Symbol*	Security symbol. See Symbology (p. 5) for additional notes.

[6] *RoutingInst* (9303)=BP

59	*TimeInForce*	0 = Day 1 = GTC (allowed, but treated as Day) 2 = AtTheOpen 3 = IOC (treated as Accept-Or-Cancel for *RoutingInst* (9303) = BP or BH) 6 = GTD (expires at earlier of specified *ExpireTime* (126) or end of day) 7 = AtTheClose 8 = Good For Auction (only valid if *RoutingInst* (9303) = BP or BU)
60	*TransactTime*	Time order initiated/released. Required by FIX 4.2 but not used by Cboe. Microsecond level resolution.
110	*MinQty*	**Optional** minimum[7] fill quantity for Cboe Only hidden, Cboe Dark Pool, Cboe Dark Periodic Sweep, Cboe Periodic Auction Book, Cboe Closing Cross or IOC orders. **Rejected** for Dark, Lit and Dark Lit Sweep Types. **Ignored** for other orders. **Default** is zero.
111	*MaxFloor*	Portion of *OrderQty* (38) to display. The balance is reserve. 0 displays entire quantity (default). The displayed quantity of each order at a price level is decremented first. When displayed quantity is fully decremented, it is reloaded up to *MaxFloor* from reserve. May opt-in at the firm or port level to receive a restatement execution report on each reserve reload, allowing a Participant to know the new OrderID as represented on the Cboe market data feeds.
126	*ExpireTime*	Required for *TimeInForce* (59) = 6 (GTD) orders, specifies the date and time (in GMT) that the order expires.
207	*SecurityExchange*	Required when *IDSource* (22) = 4 (ISIN).
211	*PegDifference*	**Optional signed** value up to four decimal places[8] is **added** to the result of peg calculation. Default is 0. Must be ≥ 0 for sell orders. Must be ≤ 0 for buy orders. Must be zero (or not specified) for midpoint peg or non-pegged orders. For periodic auction orders where *ExecInst* (18) = R (Near-touch peg), the value can be > 0 for buy orders,or < 0 for sell orders. This indicates an aggressive offset for the near-touch peg. Please be aware that the peg price is compared against the mid-point of the EBBO, with the least aggressive price chosen as the working price of the order.
439	*ClearingFirm*	Firm that will clear trade. Optional. Note: shares storage with *OnBehalfOfCompID* (115). If both fields are set, they must be equal.
440	*ClearingAccount*	Supplemental identifier. Optional. Recorded and returned in execution reports. Available via Drop. Note: shares storage with *OnBehalfOfSubID* (116). If both fields are set, then *OnBehalfOfSubID* (116) takes precedence for Service Bureau connections and *ClearingAccount* takes precedence for other connections.

[7] Once resting all fills will exceed the minimum. On entry and user modification, the behaviour is configurable on the port and can apply to the **total** fill size, which may be made up of several **consecutive** smaller fills.
[8] *PegDifference* is rounded (down for buy, up for sell) to fit the tick size.

1724	*OrderOrigination*	5 = DEA. Indicate DEA activity (as defined by MiFID II) is involved in the order. 0 = Non-DEA. (default) Other values are unsupported and will be rejected.
7928	*PreventParticipant Match*	Participant Trade Prevention: 3 characters (not space separated): 1st character - PTP Modifier: N = Cancel Newest O = Cancel Oldest B = Cancel Both D = Decrement Larger[9] / Cancel Smaller d = Same as D above, but only decrement *LeavesQty* (151). Do not restate *OrderQty* (38). 2nd character - Unique ID Level: N = Do not prevent (Default value if not specified) F = Prevent Match at Participant Level M = Prevent Match at Trading Firm Level P = Prevent Match at Port Owner Level 3rd character - Trading Group ID (optional): Member specified alphanumeric value 0-9, A-Z, or a-z. The Unique ID level (character 2) of both orders must match to prevent a trade. If specified on both orders, Trading Group ID (character 3) must match to prevent a trade. The PTP Modifier (character 1) of the inbound order will be honored, except that if the inbound order specified Decrement and the resting order does not, and the resting order is larger, then both orders will be canceled. This exception is to protect the order entry software for the resting order from receiving an unexpected restatement message.
8015	*OrderAttributeTypes*	Optional. This FIX tag can contain multiple values. If more than one value is present, they must be separated by spaces. The presence of a value means, for example, the order is an algorithmic order. The absence of a value indicates otherwise. Cboe supports the following values: 2 = Liquidity Provision activity order. This indicates the order is related to any sort of liquidity provision activity, as defined by MiFID II. This flag is mandatory for orders which are part of a liquidity provision activity. Absence of this value indicates otherwise. 4 = Algorithmic order. This indicates that the order was placed as a result of an investment firm engaging in algorithmic trading. Absence of this value indicates otherwise.

[9] Users of PTP modifier D must be prepared to receive a FIX Restatement execution report (*ExecType* (150) = D) that includes both *OrderQty* (38) and *LeavesQty* (151).

9303	*RoutingInst*	Up-to 4 characters (not space separated): `B` = Cboe Only **(default)** `P` = Cboe Only — Post Only (will reject rather than remove visible liquidity) `U` = Dark Sweep (interbook) `u` = Dark Lit Sweep (best-price) `W` = Lit Sweep (interbook, best-price) `X` = Lit Sweep (interbook, sequential) `BD` = Cboe Dark Book Only (hidden midpoint peg orders only) `BA` = Cboe Automatic Dark Routed (routes to Cboe Integrated Book if order is Large In Scale (LIS) or is not a midpoint order, otherwise routes midpoint non-LIS orders to Cboe Dark Book) `BP` = Cboe Periodic Auction Book `BH` = Dark Periodic Sweep (midpoint peg orders) `BU` = Cboe Closing Cross `BL` = Periodic Lit Sweep `BM` = Dark Periodic Lit Sweep An optional 3^{rd} character can be used to specify Price Improvement for Cboe Periodic Auction Book (`BP`) limit orders only: `N` = None (implied if 3^{rd} char is not specified) `L` = Limit (1-tick inside the limit price) `T` = Far-touch (least aggressive of 1-tick inside the EBBO far-touch or limit price) `B` = Both (least aggressive of 1-tick inside far-touch or 1-tick inside limit) An optional 4^{th} character can be used to specify Dark-on-Expiry for Periodic Auction Book and Dark Periodic Sweep orders. Only applicable when *TimeInForce* (59) = `8` (Good-For-Auction) or *TimeInForce* (59) = `3` (AOC) and the order is deemed marketable (ignored for other order types or *TimeInForce* values): `N` = None (implied if 4^{th} char is not specified) `D` = Dark-on-Expiry Post Only does not mix with *TimeInForce* (59) = `3` (IOC). If a *RoutingInst* is not specified a default value of `B` is implied (Cboe Only).
9416	*ExtExecInst*	`N` = None `G` = All Or None The all or none instruction applies to periodic auctions only.
9479	*DisplayIndicator*	`X` = Displayed Order (default) `I` = Invisible Invisible orders must meet the MiFID requirements for Large in Scale (LIS) unless routed to the Cboe Dark Book.
9688	*OrigCompID*	**Drop only.** *TargetCompID* (56) of original FIX execution report. Drop port must be configured to send this optional field.
9689	*OrigSubID*	**Drop only.** *TargetSubID* (57) of original FIX execution report. Drop port must be configured to send this optional field.

7692	*RiskReset*	For use by customers to reset or release clearing firm, symbol or *CustomGroupID* (7699) level lockout conditions resulting from self-imposed lockouts issued via `Purge Request messages`. **Single Character Values**: `S` = Symbol level lockout reset `F` = Clearing firm level lockout reset `C` = CustomGroupID lockout reset Values may be combined together to allow for resets of multiple self-imposed lockouts in a single message. For example, FS, SC, FC, and SFC are all acceptable values. If orders have been locked out at any level, inbound orders for the locked *Symbol* (55), clearing firm, or *CustomGroupID* (7699) will be rejected until this field is filled with the appropriate value on a `New Order Single message`.
7699	*CustomGroupID*	Custom identifier for a group of orders.
453	*NoPartyIDs*	Indicates the number of instances of the repeating group *NewOrderPtyRptGrp* to follow. Defaults to zero.
Repeating Group *NewOrderPtyRptGrp* must occur the number of times specified in *NoPartyIDs* (453)		
448	*PartyID*	The short code representing the client or decision maker represented by this block. Unsigned numerical only. Data corresponding to this short code must have been previously supplied, or will be supplied by the end of the calendar day, per our Rules. For clients, the following values are reserved for applicable use: Applicable to PartyRole value 3: 0 = NONE (No Client for this order) 1 = AGGR (An aggregation of multiple client orders) 2 = PNAL (Clients are pending allocation) Applicable to PartyRole value 12: 3 = NORE (Timing and location of the execution determined by the client of the Participant)
447	*PartyIDSource*	Must always be `P` (Short code identifier)
452	*PartyRole*	Specifies the role of the party to the trade. At this time, only the following values are valid: 3 = Client ID 12 = Executing Trader (the Executing Decision Maker) 122 = Investor ID (the Investment Decision Maker)
2376	*PartyRoleQualifier*	Provides further qualification of the PartyRole value. Valid values are: 0 = None (applicable only for the reserved Party IDs) 22 = Algorithm (applicable to PartyRole values 12 or 122) 23 = Firm or legal entity (LEI) (applicable to PartyRole value 3) 24 = Natural person (applicable to PartyRole values 3, 12 and 122)
	Standard Message Trailer	

5.1.1 Notes on Pegged Orders

Midpoint pegged orders (*ExecInst* (18) = M or L) are implicitly hidden. Midpoint peg orders may execute between the minimum price increment of a stock, except for those stocks which are quoted at a 0.0001 increment. In that case, the peg price will be the less aggressive rounded price (rounded down for buys, rounded up for sells). Midpoint pegs may not use *PegDifference* (211).

Peg orders are prioritised behind non-pegged orders at each price and display level. With regard to hidden peg orders, regular peg orders (*ExecInst* (18) = R or P) have a higher priority than midpoint peg orders ranked at the same price.

Pegged orders will be automatically suspended from execution if Cboe loses receipt of market data for any reason or if the primary exchange halts the symbol (including for non-regulatory reasons, such as a volatility interrupt).

5.1.2 MiFID II Short Code Identifier Ranges

Cboe supports six separate ranges of short codes listed below. A range is provided for each valid combination of *PartyRole* (452) and *PartyRoleQualifier* (2376). These tags are used to fully qualify the type of short code in *PartyID* (448).

- Client (Person) - *PartyRole* (452) = 3 and *PartyRoleQualifier* (2376) = 24
- Client (Entity) - *PartyRole* (452) = 3 and *PartyRoleQualifier* (2376) = 23
- Investment Decision Maker (Person) - *PartyRole* (452) = 122 and *PartyRoleQualifer* (2376) = 24
- Investment Decision Maker (Algorithm) - *PartyRole* (452) = 122 and *PartyRoleQualifier* (2376) = 22
- Execution Decision Maker (Person) - *PartyRole* (452) = 12 and *PartyRoleQualifier* (2376) = 24
- Execution Decision Maker (Algorithm) - *PartyRole* (452) = 12 and *PartyRoleQualifier* (2376) = 22

Each range is four bytes in length. Participants can use numbers 4 through to 4,294,967,295 as short codes. Values 0, 1, 2 and 3 are reserved for applicable use as per *PartyID* (448) tag definition on pg 21.

5.2 Order Cancel Request

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = F
97	*PossResend*	N = indicates a new cancel (default) Y = indicates an application level resend. If *ClOrdID* (11) has not yet been seen, the cancel is treated as normal. If *ClOrdID* (a)lready exists, the resent cancel is ignored.
1	*Account*	Optional. Reflected back on Pending Cancel Execution Report or Cancel Reject associated with this cancel. 16 characters or less (ASCII 33–126). Configurably available via Drop.
11	*ClOrdID*	Day-unique cancel ID chosen by client. 20 characters or less. Characters in ASCII range 33–126 are allowed, except for commma, semicolon, and pipe. Duplicate *ClOrdID*s will be rejected (or ignored if *PossResend* (97) = Y.
15	*Currency*	Required if *IDSource* (22) = 4 (ISIN). If *Currency* (15) is included when other symbology is used, it must match the currency expected by Cboe for the given symbol.
22	*IDSource*	Values supported by Cboe: 4 = ISIN 5 = RIC Required if *Symbol* (55) is not set.
37	*OrderID*	Order identifier supplied by Cboe on the order acknowledgement. (Optional, but recommended for performance.)
38	*OrderQty*	Number of shares for the order. Must match original order.
41	*OrigClOrdID*	*ClOrdID* of the order to cancel.
48	*SecurityID*	ISIN, or RIC if *IDSource* (22) is set.
54	*Side*	1 = Buy 2 = Sell 5 = Sell Short 6 = Sell Short Exempt H = Sell Undisclosed
55	*Symbol*	Security symbol. See Symbology (p. 5) for additional notes.
60	*TransactTime*	Time cancel initiated/released. Required by FIX 4.2 but not used by Cboe. Microsecond level resolution.
207	*SecurityExchange*	Required when *IDSource* (22) = 4 (ISIN).
	Standard Message Trailer	

5.3 Order Cancel/Replace Request

Only *Price* (44), *OrderQty* (38), and *OrdType* (40) may be adjusted. Any change in *Price* or increase in *OrderQty* will result in the order losing its time priority. *OrdType* may be adjusted from Limit to Market (but not from Limit to Peg or Peg to Limit).

Other fields (including ExecInst (18)) will be ignored, and the value from the original order will be reused.

Changes in *OrderQty* result in an adjustment of the current order's *OrderQty*. The new *OrderQty* does **not** directly replace the current order's *LeavesQty* (151). Rather, a delta is computed from the current *OrderQty* and the replacement *OrderQty*. This delta is then applied to the current *LeavesQty*. If the resulting *LeavesQty* is less than or equal to zero, the order is canceled. This results in safer behavior when the replace request overlaps partial fills for the current order, leaving the Participant in total control of the share exposure of the order.

MaxFloor (111) is preserved from the original order and applied to the new size.

A Cancel/Replace should not be issued until the acknowledgment for the previous Cancel/Replace has been received for that order (or the New Order acknowledgment for the first Cancel/Replace). The FIX handler will reject a new Cancel/Replace if it has not seen the prior Cancel/Replace from the Matching Engine.

Cancel/Replace requests that merely reduce *OrderQty* may be overlapped if the existing *ClOrdID* is reused, as long as the trading identifier has not been opted-in to daily limit trading risk controls. This is the only case where reuse of the existing *ClOrdID* is allowed.

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = `G`
97	*PossResend*	`N` = indicates a new cancel/replace (default) `Y` = indicates an application level resend. If the *ClOrdID* does not indicate an already pending cancel/replace, the cancel/replace is treated as normal. If *ClOrdID* does indicate an already pending cancel/replace, then the resent cancel/replace is ignored.
1	*Account*	Optional. Returned on execution reports associated with this order. 16 characters or less (ASCII 33–126). **H:** and **C:** prefix can be used to specify which CCP Account to use. **If configured by Cboe**: *House* or *Client* CCP account can be defaulted. *OrderCapacity* (47) is no longer used to determine which CCP account to use. The value supplied can be passed to the CCP and made available on the Drop feed.
11	*ClOrdID*	Day-unique ID chosen by client. 20 characters or less. Characters in ASCII range 33–126 are allowed, except for comma, semicolon, and pipe. If the *ClOrdID* matches a live order, it will be rejected as duplicate (unless *PossResend* (97) = `Y`; see above). **Note: Cboe only enforces the uniqueness of ClOrdID values among currently live orders. However, we strongly recommend that you keep your ClOrdID values day unique.**
15	*Currency*	Required if *IDSource* (22) = `4` (ISIN). If *Currency* (15) is included when other symbology is used, it must match the currency expected by Cboe for the given symbol.
22	*IDSource*	Values supported by Cboe: `4` = ISIN `5` = RIC Required if *Symbol* (55) is not set.
37	*OrderID*	Order identifier supplied by Cboe on the order acknowledgement. In the case of multiple changes to a single order, this should be the *OrderID* from the most recent acknowledgement.
38	*OrderQty*	Number of shares for the order. This will modify the *OrderQty* of the current order, it does **not** directly set the remaining quantity.

40	*OrdType*	1 = Market 2 = Limit P = Pegged During continuous trading Market (1) implies *TimeInForce* (59) = 3 (IOC). Pegged requires *ExecInst* (18) = L, M, P, or R. Pegged orders may not be routable.
41	*OrigClOrdID*	*ClOrdID* of the order to replace. In the case of multiple changes to a single order, this will be the *ClOrdID* of the most recently accepted change.
44	*Price*	Limit price.
48	*SecurityID*	ISIN, or RIC if *IDSource* (22) is set.
54	*Side*	1 = Buy 2 = Sell 5 = Sell Short 6 = Sell Short Exempt H = Sell Undisclosed
55	*Symbol*	Security symbol. See Symbology (p. 5) for additional notes.
60	*TransactTime*	Time cancel/replace initiated/released. Required by FIX 4.2 but not used by Cboe. Microsecond level resolution.
207	*SecurityExchange*	Required when *IDSource* (22) = 4 (ISIN).
9619	*CancelOrigOnReject*	N = Leave original order alone (default) Y = Cancel original order if replacement fails Default may be configured per port.
	Standard Message Trailer	

5.4 Purge Request

Request to cancel a group of orders across all of the firms sessions. A Purge Request is accepted only on dedicated FIX Purge Ports.

A firm may choose to implement one or more filters:

- MPID Filter - optionally cancel based on MPID. This is required for any self-imposed lockouts or for service bureaus. Set using first character of *MassCancelInst* (7700) and sending *OnBehalfOfCompId* (115).
- Symbol Filter - optionally cancel based on symbol. Set by sending a valid symbol in the *Symbol* (55) field. Cannot be combined with CustomGroupID filter.
- CustomGroupID Filter - optionally cancel based on *CustomGroupID* (7699). A maximum of 10 custom group IDs may be included on a single Purge Request message. Set by populating *CustomGroupIDCnt* (7698) to a non-zero value. Cannot be combined with Symbol filter.

A firm may use the second character of *MassCancelInst* (7700) to set the acknowledgement style. If a single Purge Acknowledgement is selected, then *MassCancelID* (7695) must be sent. A firm may also impose a lockout using the third character of *MassCancelInst* (7700), which cancels any open orders and causes inbound orders received after the lockout to be rejected. A self-imposed lockout requires an MPID (115) to be sent. The firm may also choose to lockout by *Symbol* (55) or *CustomGroupID* (7699) but not by both in the same message.

The system limits the rate at which identical Purge Request messages can be submitted to the system. Requests are restricted to twenty (20) messages per second per port.

An identical purge message is defined as a message having all of the same *CustomGroupID* (7699), *Symbol* (55), *OnBehalfOfCompId* (115), and Lockout Instruction field values, as a previously received message.

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = `F`
97	*PossResend*	`N` = indicates a new cancel (default) `Y` = indicates an application level resend. If *ClOrdID* (11) has not yet been seen, the cancel is treated as normal. If *ClOrdID* (a)lready exists, the resent cancel is ignored.
60	*TransactTime*	Time cancel initiated/released. Required by FIX 4.2 but not used by Cboe. Microsecond level resolution.
7700	*MassCancelInst*	At least one character must be provided (clearing firm Filter). Contiguous characters must be specified up to total length. Truncated (unspecified) characters will default to values indicated below. **1st Character : clearing firm Filter** `A` = No filtering by clearing firm is performed. `F` = All orders that were sent under the clearing firm specified in *OnBehalfOfCompId* (115) will be cancelled. **2nd Character : Acknowledgement Style** `M` = (Default) Individual Execution Reports are sent for each cancelled order. `S` = Single Purge Acknowledgement sent once all cancels have been processed. Single Purge Acknowledgement will contain *MassCancelId* (7695) and *CancelledOrderCount* (7696). *MassCancelId* (7695) must be specified or the Purge Request will be rejected. `B` = Both individual Execution Reports and single Purge Acknowledgement. Also requires *MassCancelId* (7695) to be specified or the Purge Request will be rejected. **3rd Character : Lockout Instruction** `N` = (Default) No lockout `L` = Lockout until corresponding Risk Reset received. Lockout can be used only with clearing firm Filter set to F, otherwise the Purge Request will be rejected. Lockout will apply to all new orders and cancel/replace orders for the clearing firm (and symbol or *CustomGroupId* (7699), if specified). A self-imposed lockout can be released using the *RiskReset* (7692) field of the New Order Single message.
7695	*MassCancelID*	This field will be echoed back in the resulting `Purge Acknowledgement` when the second character of *MassCancelInst* (7700) is set to "S" or "B". Purge requests containing a *MassCancelID* that is currently outstanding will be rejected.
48	*SecurityID*	ISIN, or RIC if *IDSource* (22) is set.
22	*IDSource*	Values supported by Cboe: `4` = ISIN `5` = RIC Required if *Symbol* (55) is not set.
55	*Symbol*	Security symbol. See Symbology (p. 5) for additional notes.
207	*SecurityExchange*	Required when *IDSource* (22) = `4` (ISIN).
15	*Currency*	Required if *IDSource* (22) = `4` (ISIN). If *Currency* (15) is included when other symbology is used, it must match the currency expected by Cboe for the given symbol.
7698	*CustomGroupIDCnt*	Number of custom group IDs. Must be between 1 and 10.

Repeating Group *CustomGroupIDs* must occur the number of times specified in *CustomGroupdIDCnt* (7698)		
7699	*CustomGroupID*	Custom identifier for a group of orders.
	Standard Message Trailer	

5.5 Trade Capture Report

The Trade Capture Report is used to submit a Negotiated Trade. The report must contain both sides of the trade (*NoSides* (552) = 2).

The models supported are as described in the FIX 5.0 (SP2) specification in the *Two-Party Reporting*, *One-Party Report for Matching* and *Confirmed Trade Reporting* workflow diagrams of the Trade Capture Reporting section.

Whilst we make use of FIX 4.4 and FIX 5.0 messages/tags, these are handled as extensions and operate over a FIX 4.2 session. If a port is dedicated to the use of trade reporting, there is an option to utilise a FIX 4.4 session.

When a new trade report is accepted, a *TradeID* is returned in the corresponding Trade Capture Report confirmation messages. Where applicable, such trade reports may be cancelled, amended or released by specifying the *TradeID*.

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = AE
15	*Currency*	Required if *IDSource* (22) = 4 (ISIN). If *Currency* (15) is included when other symbology is used, it must match the currency expected by Cboe for the given symbol.
22	*IDSource*	Values supported by Cboe: 4 = ISIN 5 = RIC Required if *Symbol* (55) is not set.
31	*LastPx*	Price of this fill. May be excluded if using *GrossTradeAmt*.
32	*LastShares*	Quantity of shares traded on this fill.
48	*SecurityID*	ISIN, or RIC if *IDSource* (22) is set.
55	*Symbol*	Security symbol. See Symbology (p. 5) for additional notes.
60	*TransactTime*	Optional, when *TradeReportTransType* (487) = 0. Microsecond level resolution.
63	*SettlType*	Optional. Indicates settlement period. If present, *SettlDate* (64) overrides this field. If *SettlType* (63) = 6, then *SettlDate* (64) is required. If both *SettlType* (63) and *SettlDate* (64) are omitted, the default value is 0 (Regular). Supported values: 0 = Regular (default) 6 = Future This field may not be amended. If TradeHandlingInstr (1123) = 2 (One Party Report for Matching), SettlType must be set to 0 (Regular)
64	*SettlDate*	Optional. Specifies the date on which the trade is desired to settle. If present, this field overrides *SettlType* (63). This field is required if the value of *SettlType* (63) = 6 (Future). May only be specified on new trade reports. This field may not be amended using our standard interface, but the actual settlement date may be varied by the central counterparties (CCPs) due to operational requirements (eg. for symbols in a conditional trading status).
75	*TradeDate*	Optional. Specifies the date on which the trade was arranged. If present, must match the date component of *TransactTime* (60).
120	*SettlCurrency*	Optional. Currency in which the trade should settle. Must be USD or EUR. This field may not be amended.
150	*ExecType*	Must be F = Trade
207	*SecurityExchange*	Required when *IDSource* (22) = 4 (ISIN).
381	*GrossTradeAmt*	Total amount traded, expressed in units of currency. Only considered when *LastPx* is not specified. Supports a maximum of 9 digits before decimal.
487	*TradeReportTransType*	Specifies whether this trade report is new, a cancellation, an amendment or a release. Unilateral trades reported under the service may be cancelled or amended by the participant. Trades that are currently being delayed from publication can be released for immediate publication. Defaults to 'new' if unspecified. 0 = New 1 = Cancel 2 = Replace 3 = Release
571	*TradeReportID*	Day-unique ID chosen by client. Cboe will enforce port level day-uniqueness. 20 characters or less. Characters in ASCII range 33–126 are allowed, except for comma, semicolon, and pipe. If the *TradeReportID* matches a live trade report (one that has been acked, but not confirmed or declined), it will be rejected as duplicate.
572	*TradeReportRefID*	Contains the *TradeReportRefID* (572) of the trade capture report ack that should now be withdrawn

574	*MatchType*	Where *VenueType* (1430) = 0 (off book), this field models the MMT Level 2 'Trading Mode' field, and must be 3 = Trade Reporting (On-Exchange).
730	*SettlPrice*	Optional, mandatory when *SettlCurrency* (120) is used. Price at which the trade should settle. If specified, any risk controls will be applied against this price. This field may not be amended.
820	*TradeLinkID*	Third Party Trade Identifier used for optional matching with counterparty. 30 characters or less. Characters in ASCII range 33–126 are allowed, except for comma, semicolon, and pipe.
828	*TrdType*	This field corresponds to the MMT Level 3.1 field 'Transaction Category'. 0 = Regular Trade 2 = Exchange For Physicals (XFPH), for RTS 2 non-equities only (Effective 1 Jan 2024) 50 = Portfolio Trade (Effective 1 Jan 2024). Refer to p. 9 for more details. This corresponds to MMT Level 3.11 field 'Portfolio', PORT. Should only be set to this value when used in conjunction with *SecondaryTrdType* (855) = 64 (BENC) and *TertiaryTrdType* (2896) = 65 (CONT). 65 = Package Trade (TPAC) which are not Exchange for Physicals (XFPH), for RTS 2 non-equities only. Use 2 if it's Exchange for Physicals (XFPH) (Effective 1 Jan 2024).
829	*TrdSubType*	This optional field corresponds to the MMT Level 3.3 field 'Agency Cross Trade Indicator'. Agency Cross trades may be indicated by setting *TrdSubType* (829) = 37. Effective 29 April 2024, this is deprecated for BXE and CXE. Other values are invalid.
855	*SecondaryTrdType*	This optional field corresponds to the MMT Level 3.5 field 'Benchmark Indicator'. Benchmark trades may be indicated by setting *SecondaryTrdType* (855) = 64. Other values are invalid. Effective 1 Jan 2024, valid values are as follows. Refer to p. 9 for more details 50 = Portfolio Trade (MMT Level 3.11 field 'Portfolio', PORT) 61 = RFMD Give-up Trade (MMT Level 3.13 field 'Give-up', GIVE) 64 = Benchmark (MMT Level 3.5 field 'Benchmark Indicator', BENC) 65 = Contingent Transaction (Package Trade (TPAC)) (Only Supported in DXE, RTS 1 equities only) (MMT Level 3.12 field 'Contingent', CONT) 67 = Benchmark Transactions Executed At the Market Close Price (MMT Level 3.5 field 'Benchmark At Market Close Price', CLSE) (Supported in BXE and CXE, for RTS 1 equities only, from 29 April 2024)
856	*TradeReportType*	This field controls pending state of the trade report. 0 = (Submit) for all new trade reports 6 = (Trade Report Cancel) to cancel any acknowledged, but not confirmed trade reports entered where *TradeHandlingInst* (1123) = 2
1003	*TradeID*	Used to specify a previously reported trade to be amended or cancelled. Mandatory when *TradeReportTransType* (487) = 1, 2 or 3, must be absent when *TradeReportTransType* (487) = 0.
1115	*OrderCategory*	This field corresponds to the MMT Level 3.2 field 'Negotiated Transaction Indicator', and is used by the participant to indicate that the trade was a Negotiated Transaction as per the Cboe Rules. For all trade reports reported on-exchange, the value must be 3. 3 = Privately Negotiated Trade
1116	*NoRootPartyIDs*	Must be 1 or 2

Repeating Group *RootParties* must occur the number of times specified in *NoRootPartyIDs* (1116)		
1117	*RootPartyID*	Either an identifier for the third party submitting the trade, (4 uppercase letters) known to Cboe, when *TradeHandlingInst* (1123) = 0 and *RootPartyRole* (1119) = 6. Or EB (Euroclear Bank) when *RootPartyRole* (1119) = 10.
1118	*RootPartyIDSource*	Acceptable values: D = Proprietary / Custom Code
1119	*RootPartyRole*	Specifies the role of the given *RootPartyID*. Acceptable values: 6 = IntroducingFirm, when *TradeHandlingInst* (1123) = 0 10 = SettlementLocation
1123	*TradeHandlingInstr*	Used to specify the trade reporting model used. 0 (Confirmed Trade) 1 (Two-Party Report) 2 (One Party Report for Matching)
1390	*TradePublishIndicator*	This field corresponds to the MMT Level 4.1 field 'Publication Mode', and is used by the participant to request that the publication be delayed. In order for RTS 1 based instruments to be considered for a deferral, *OrderCapacity* (528) = P (maps to 'DEAL') must be set. For RTS 1 and RTS 2 instruments, delayed publication/deferrals are ignored if the trade does not qualify for delayed publication. This field may not be amended, however trades currently being delayed may be released prior to their maximum delay duration using *TradeReportTransType* (487) = 3. Supported values: 0 = Do Not Publish. Deprecated from 4th December 2017 in Certification and 2nd January 2018 in Production. Any requests to publish a trade using this indicator will not be honoured and will instead be published immediately. 1 = Publish Trade Immediately 2 = Deferred Publication
1430	*VenueType*	Must be 0 = Off Book. This field models the MMT Level 1 field 'Market Mechanism'.
1838	*NoTradePriceConditions*	Optional. If present, indicates the number of *TradePriceCondition* (1839) fields.
Repeating Group *TradePriceConditionGrp* must occur the number of times specified in *NoTradePriceConditions* (1838)		
1839	*TradePriceCondition*	Optional. Used to indicate values in MMT v3/v4.1 levels 3.2, 3.6 and 3.8 For MMT Level 3.2 'Negotiation Indicator', supported values are: 16 = Negotiated Trade Subject to Conditions Other than The current Market Price (PRIC) Note that this is no longer required or supported on BXE or CXE from 29 Apr 2024. For MMT Level 3.6 'Special Divided Indicator', supported values are: 13 = Special Dividend (SDIV) For MMT Level 3.8 'Contribution to Price Formation or the Price Discovery Process', supported values are: 15 = Non-Price Forming Trade (NPFT)

2373	*IntraFirmTradeIndicator*	This field is valid from 1 Jan 2024, and indicates if a transaction is between entities within the same group carried out for intra group risk management purposes to transfer risk accumulated as a result of previously executed transactions or where necessary to comply with regulatory requirements. N = Not an intra-group transaction (default) Y = Is an intra-group transaction
2405	*ExecMethod*	Optional. Used to indicate that the method by which the trade was executed. This field corresponds to the proposed MMT Level 3.7 'Offbook Automated Liquidity Indicator'. The following values are supported: 0 = Unspecified (default) 1 = Manual 2 = Automated
2667	*AlgorithmicTrade Indicator*	Indicates that the submitted trade was a result of an investment firm engaging in algorithmic trading. Optional. 0 = No algorithm was involved (the default). 1 = The trade was an algorithmic trade (ALGO).
2896	*TertiaryTrdType*	This field is valid from 1 Jan 2024, and valid values are as follows. Refer to p. 9 for more details 50 = Portfolio Trade (MMT Level 3.11 field 'Portfolio', PORT) 61 = RFMD Give-up Trade (MMT Level 3.13 field 'Give-up', GIVE) 64 = Benchmark (MMT Level 3.5 field 'Benchmark Indicator', BENC) 65 = Contingent Transaction (Package Trade (TPAC)) (Only Supported in DXE, RTS 1 equities only) (MMT Level 3.12 field 'Contingent', CONT) 67 = Benchmark Transactions Executed At the Market Close Price (MMT Level 3.5 field 'Benchmark At Market Close Price', CLSE) (Supported in BXE and CXE, for RTS 1 equities only, from 29 April 2024)
9128	*Tolerance*	Maximum allowed delta (in terms of consideration, expressed in the traded currency), that the trade is prepared to match against counterparty. The tolerance should be specified by the buyer in the traded currency of the stock and capped at 5,000 currency units (e.g. 5000 GBX). If the tolerance selected allows for the price submitted by the selling Participant, then it is the price provided by the selling Participant that will be applied to the transaction. Sub-decimal tolerance is not allowed (e.g. 50.20).
552	*NoSides*	Must always be 2

Repeating Group *TrdCapRptSideGrp* must occur the number of times specified in *NoSides* (552)

54	*Side*	Must be first field in repeating-group 1 = Buy 2 = Sell 5 = Sell Short 6 = Sell Short Exempt H = Sell Undisclosed
1	*Account*	Optional. Reflected back on Trade Capture Report confirmed and declined messages. **H:** and **C:** prefix can be used to specify which CCP Account to use. 16 characters or less (ASCII 33–126).
528	*OrderCapacity*	Optional. A = Agency (maps to 'AOTC') P = Principal (maps to 'DEAL') R = Riskless (maps to 'MTCH')
625	*TradingSessionSubID*	This field must not be supplied on trade reports to Cboe. This field models the MMT Level 2 'Trading Mode' field for scenarios where *VenueType* (1430) is not '0' (off book), which is not permitted.
453	*NoPartyIDs*	Must always be 1.

Repeating Group *Parties* must occur the number of times specified in *NoPartyIDs* (453)

448	*PartyID*	The end-client responsible for the trade. Must be an identifier (4 uppercase letters) known to Cboe.
447	*PartyIDSource*	Must always be D (Proprietary / Custom Code)
452	*PartyRole*	Specifies the role of the party to the trade. At this time, only the following values are valid: 1 = ExecutingFirm (if used, must be set on both sides. Is not permitted for bilateral trades, except for Trade Confirmations) 3 = Client ID (**Drop only**) 6 = IntroducingFirm (**Drop only**) 7 = EnteringFirm (the party reporting the trade. Should not be used in the second leg, except when *TradeHandlingInst* (1123) = 0) 14 = GiveupClearingFirm (**Drop only**) 17 = ContraFirm (the party the trade is alleged against)

	Standard Message Trailer	

6 FIX Application Messages — Cboe to Participant

6.1 Execution Report

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = 8
1	*Account*	Copied from order, if present. (Available on Drop via optional port configuration)
6	*AvgPx*	Average fill price.
11	*ClOrdID*	*ClOrdID* of the order being accepted, executed, or rejected. –or– *ClOrdID* of the cancel or replace **request**. –or– *ClOrdID* of the order subject to unsolicited cancel (*OrigClOrdID* (41) will not be present).
14	*CumQty*	Cumulative quantity of shares executed for this order.
15	*Currency*	Copied from order, if present.
17	*ExecID*	Day-unique ID of execution message.
18	*ExecInst*	Copied from order, if present.
20	*ExecTransType*	0 = New
22	*IDSource*	Copied from order, if present.
29	*LastCapacity*	Broker capacity in order execution. 1 = Agent (maps to 'AOTC') 3 = Cross as Principal (maps to 'MTCH') 4 = Principal (maps to 'DEAL')
30	*LastMkt*	Populated with the segment MIC of this fill.
31	*LastPx*	Price of this fill (zero for non-fills).
32	*LastShares*	Quantity of shares traded on this fill (zero for non-fills).
37	*OrderID*	Order identifier supplied by Cboe.
38	*OrderQty*	Copied from order.
39	*OrdStatus*	State of order. 0 = New 1 = Partially Filled 2 = Filled 4 = Canceled 5 = Replaced 6 = Pending Cancel 8 = Rejected A = Pending Ack E = Pending Replace
41	*OrigClOrdID*	*ClOrdID* of the order being canceled or replaced (for a solicited cancel or cancel/replace, otherwise not present).
44	*Price*	Copied from order.
48	*SecurityID*	Copied from order, if present.
54	*Side*	Copied from order or trade report.
55	*Symbol*	Copied from order, if present.

58	*Text*	If present, indicates reason for the message. Format is one letter reason code followed by colon and space followed by free form text message. Reason codes are: A = Admin D = Duplicate *ClOrdID* H = Halted K = Order Rate Threshold Exceeded k = Pending periodic auction (BXE and DXE only) L = Price Exceeds Cross Range N = Ran Out of Liquidity to Execute Against O = *ClOrdID* Doesn't Match a Known Order P = Can't Modify an Order That is Pending Fill Q = Waiting For First Trade R = Routing Unavailable T = Routing order would trade through an away destination U = User Requested V = Would Wash W = Add Liquidity Only Order Would Remove X = Order Expired Y = Symbol Not Supported Z = Unforseen Reason l = Large in Scale r = Reserve Reload m = Market Access Risk Limit Exceeded o = Max Open Orders Count Exceeded p = Static Collar Breach s = Symbol Level Risk Management v = MiFID II Double Cap related
59	*TimeInForce*	Copied from order.
60	*TransactTime*	Time transaction occurred. Microsecond level resolution.
63	*SettlType*	**Drop only.** Only relevant if *ExecType* (150) = F and the original trade report utilised a non-standard settlement date. In this case, the relevant date will also be included in *SettlDate* (64).
64	*SettlDate*	**Drop only.** Only relevant if *ExecType* (150) = F and the original trade report utilised *SettlType* (63) ≠ 0, in which case, the relevant date will be included in *SettlDate* (64); or, if the original trade report specified a *SettlDate* (64).
103	*OrdRejReason*	Optionally set when *ExecType* (150) = 8 (Rejected). 0 = Broker Option 1 = Unknown Symbol 2 = Exchange Closed 3 = Order Exceeds Limit 5 = Unknown Order 6 = Duplicate Order 8 = Stale Order
111	*MaxFloor*	Copied from order.
126	*ExpireTime*	Copied from order if *TimeInForce* (59) = 6 (GTD).

150	*ExecType*	Reason for this execution report. 0 = New (acknowledgement of new order) 1 = Partial Fill 2 = Fill 4 = Canceled 5 = Replaced 8 = Rejected D = Restated For **Drop only**, if optionally configured, the following value is also supported: F = Trade
151	*LeavesQty*	Quantity of shares still open for further execution. Will be zero if order is dead, otherwise will be *OrderQty* − *CumQty*. **Note**: It is possible for *LeavesQty* to be zero when *ExecType* (150) = 5 indicating that the order is dead.
198	*SecondaryOrderID*	Must request opt-in at firm or port level to receive this field. Present on Prevent Participant Match triggered cancel/restatement. Denotes the Cboe *OrderID* (37) of contra side of prevented match. Present on a restatement execution report for reload of a reserve (iceberg) order. Denotes the new Cboe OrderID which will be present on the Cboe market data feeds.
207	*SecurityExchange*	Copied from order, if present.
375	*ContraBroker*	Only present on trades. Indicates the market of execution. Markets are identified by their ISO Market Identification Code (MIC)[10][11]
378	*ExecRestatement Reason*	Required when *ExecType* (150) = D (Restated). 4 = Broker option; optionally sent during reload of a reserve (iceberg) order. 5 = Partial decline of *OrderQty*
382	*NoContraBrokers*	Only present on trades. Always 1.
439	*ClearingFirm*	Copied from order, if present.
440	*ClearingAccount*	Copied from order, if present.

[10] ISO 10383, see http://www.iso15022.org/MIC/homepageMIC.htm for details

[11] for historical reasons a local execution for an order entered into the BXE book is identified with a value of BATS

Repeating Group *NewOrderPtyRptGrp* must occur the number of times specified in *NoPartyIDs* (453)

448	*PartyID*	The short code representing the client or decision maker represented by this block. Unsigned numerical only. Data corresponding to this short code must have been previously supplied, or will be supplied by the end of the calendar day, per our Rules. For clients, the following values are reserved for applicable use: Applicable to PartyRole value 3: 0 = NONE (No Client for this order) 1 = AGGR (An aggregation of multiple client orders) 2 = PNAL (Clients are pending allocation) Applicable to PartyRole value 12: 3 = NORE (Timing and location of the execution determined by the client of the Participant)
447	*PartyIDSource*	Must always be `P` (Short code identifier)
452	*PartyRole*	Specifies the role of the party to the trade. At this time, only the following values are valid: 3 = Client ID 12 = Executing Trader (the Executing Decision Maker) 122 = Investor ID (the Investment Decision Maker)
2376	*PartyRoleQualifier*	Provides further qualification of the PartyRole value. Valid values are: 0 = None (applicable only for the reserved Party IDs) 22 = Algorithm (applicable to PartyRole values 12 or 122) 23 = Firm or legal entity (LEI) (applicable to PartyRole value 3) 24 = Natural person (applicable to PartyRole values 3, 12 and 122)

730	*SettlPrice*	**Drop only.** Only relevant if *ExecType* (150) = `F` and the original trade report utilised *SettlPrice* (730). This represents the price at which the trade will settle.
1724	*OrderOrigination*	`5` = DEA. Indicate DEA activity (as defined by MiFID II) is involved in the order. `0` = Non-DEA. (default) Other values are unsupported and will be rejected.
7772	*CentralCounterparty*	Only present on trades. The CCP handling the trade: EMCF = Cboe Clear LCHL = LCH Ltd (London) LCHS = LCH SA (Paris) XCLR = SIX x-clear NONE = Clearing Suppressed Returned on trades if the participant has selected a Preferred CCP. The FIX port can be configured to always return this optional field.

8013	*TrdRegPublication Reasons*	If present, indicates the post-trade waiver derived by Cboe for this trade. Valid values are: 3 = Reference Price (Dark Book) (RFPT) 9 = Large In Scale (Pre-Trade Transparency Waiver) 10 = Order Management Facility (Iceberg) (Pre-Trade Transparency Waiver)
8015	*OrderAttributeTypes*	Optional. This FIX tag can contain multiple values. If more than one value is present, they must be separated by spaces. The presence of a value means, for example, the order is an algorithmic order. The absence of a value indicates otherwise. Cboe supports the following values: 2 = Liquidity Provision activity order. This indicates the order is related to any sort of liquidity provision activity, as defined by MiFID II. This flag is mandatory for orders which are part of a liquidity provision activity. Absence of this value indicates otherwise. 4 = Algorithmic order. This indicates that the order was placed as a result of an investment firm engaging in algorithmic trading. Absence of this value indicates otherwise.
9416	*ExtExecInst*	Copied from order, if present.
9730	*TradeLiquidity Indicator*	Only present on trades. A = Added Liquidity R = Removed Liquidity AD = Added Liquidity for the Cboe Dark Pool RD = Removed Liquidity from the Cboe Dark Pool RT = Removed Liquidity from the Cboe Dark Pool by IOC order AI = Added Hidden Liquidity that was price improved AK = Added Liquidity from the hidden (reserve) portion of an iceberg order AS = Added Liquidity A-LPS BBO Setter RS = Removed Liquidity from A-LPS BBO Setter X = Routed to Another Market C = Auction CC = Cboe Closing Cross P = Periodic Auction S = Self Match (opt-in) SD = Self Match from the Cboe Dark Pool (opt-in) To allow for future expansion of this field, please ignore values with an unknown character in the 2nd position.
9882	*FeeCode*	Specific fee code associated with the trade. See the Fee Schedule for the respective market for possible values.
	Standard Message Trailer	

6.2 Cancel Reject

Rejects a Cancel or Cancel/Replace request.

When a Cancel/Replace is rejected, by default, the original order is left alive. A Cancel Reject should not be used as a sign that the original order has been canceled. Even if the CancelOrigOnReject (9619) = Y option is being used, a separate "unsolicited" cancel will be sent to close out the original order.

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = 9
1	*Account*	Copied from Cancel or Cancel/Replace request.
11	*ClOrdID*	*ClOrdID* from the Cancel or Cancel/Replace request.
37	*OrderID*	*OrderID* of the order that failed to be canceled or replaced. NONE if *CxlRejReason* (102) = 1 (Unknown Order).
39	*OrdStatus*	State of order that failed to be canceled or replaced.
41	*OrigClOrdID*	*ClOrdID* of the order that failed to be canceled or replaced.
58	*Text*	Free-form text message.
102	*CxlRejReason*	0 = Too Late to Cancel 1 = Unknown Order 3 = Already Pending Cancel or Pending Replace
434	*CxlRejResponseTo*	1 = Cancel 2 = Cancel/Replace
	Standard Message Trailer	

6.3 Trade Cancel/Correct

Trade Cancel/Correct (UCC) is an optional message that must be enabled at the port level. It may be enabled for current-day only or for all cancels and corrections. Only the price and size of a trade may be corrected, all other details remain the same. Trade cancels and corrections do not alter live order state.

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = `UCC`
11	*ClOrdID*	*ClOrdID* of the order whose trade is being canceled or corrected.
15	*Currency*	Copied from order if *IDSource* (22) = 4 (ISIN).
17	*ExecID*	Day-unique ID of execution message.
19	*ExecRefID*	Refers to the *ExecID* (17) of the execution being canceled or corrected.
20	*ExecTransType*	1 = Cancel 2 = Correct
22	*IDSource*	Copied from order being canceled or corrected, if present.
30	*LastMkt*	*LastMkt* on the original trade being canceled or corrected, if the port is configured to send this tag.
31	*LastPx*	Price on the original trade being canceled or corrected.
32	*LastShares*	Quantity of shares on the original trade being canceled or corrected.
37	*OrderID*	*OrderID* of the order whose trade is being canceled or corrected.
42	*OrigTime*	Date and time of the original trade, in GMT. Microsecond level resolution.
48	*SecurityID*	Copied from original order being canceled or corrected if *IDSource* (22) = 4 (ISIN), or 5 (RIC) was used.
54	*Side*	Copied from trade being canceled or corrected.
55	*Symbol*	Copied from original order being canceled or corrected.
60	*TransactTime*	Date and time of the cancel or correction. Microsecond level resolution.
63	*SettlType*	**Drop only.** *SettlType* on the original trade being canceled or corrected, if relevant.
64	*SettlDate*	**Drop only.** *SettlDate* on the original trade being canceled or corrected, if relevant.
207	*SecurityExchange*	Copied from order being canceled or corrected if *IDSource* (22) = 4 (ISIN) was used.
439	*ClearingFirm*	Copied from trade being canceled or corrected, if present.
440	*ClearingAccount*	Copied from trade being canceled or corrected, if present.
730	*SettlPrice*	**Drop only.** *SettlPrice* on the original trade being canceled or corrected, if relevant.
6655	*CorrectedSize*	The corrected size of the trade. Only set if *ExecTransType* (20) = 2 (Trade Correct).
9620	*CorrectedPrice*	The corrected price of the trade. Only set if *ExecTransType* (20) = 2 (Trade Correct).
9730	*TradeLiquidity Indicator*	Copied from trade being canceled or corrected.
	Standard Message Trailer	

6.4 Purge Acknowledgement

A response to a Purge Request will only be sent when the *MassCancelID* (7695) is populated on a Purge Request. This includes cases where the Acknowledgement Style of *MassCancelInst* is S or B.

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = 8
20	*ExecTransType*	3 = Status
150	*ExecType*	M = Mass Cancel Complete
7695	*MassCancelID*	Copied from the incoming Purge Request.
7696	*CancelledOrderCount*	Number of orders cancelled from a Purge Request with the specified *MassCancelID*
	Standard Message Trailer	

6.5 Purge Reject

Rejects a Purge Request.

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = 9
39	*OrdStatus*	8 = Rejected
434	*CxlRejResponseTo*	1 = Cancel
102	*CxlRejReason*	2 = Broker Option
58	*Text*	Free-form text message.
7695	*MassCancelID*	Copied from the incoming Purge Request.
	Standard Message Trailer	

6.6 Trade Capture Report Ack

The Trade Capture Report Ack is sent by Cboe to acknowledge the receipt of a Trade Capture Report. It is a technical-level ack, the Trade/Cancel/Amend is not considered to have fully succeeded until a Trade Capture Report is sent with with *TradeReportType* (856) of 2 (Accept).

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = AR
15	*Currency*	Copied from the incoming TradeCaptureReport, if present.
22	*IDSource*	Copied from the incoming TradeCaptureReport, if present.
31	*LastPx*	Copied from the incoming TradeCaptureReport, if present. If *GrossTradeAmt* was used instead of *LastPx*, the value here will be indicative. Price is adjusted for allowed precision.
32	*LastShares*	Copied from the incoming TradeCaptureReport.
48	*SecurityID*	Copied from the incoming TradeCaptureReport, if present.
55	*Symbol*	Copied from the incoming TradeCaptureReport, if present.

58	*Text*	If present, indicates reason for the message. Format is one letter reason code followed by colon and space followed by free form text message. Reason codes are: `A` = Admin `D` = Duplicate *TradeReportID* `H` = Halted `U` = User Requested `Y` = Symbol Not Supported `X` = Expired `Z` = Unforseen Reason `m` = Market Access Risk Limit Exceeded `o` = Max Open Trade Count Exceeded `v` = MiFID II Double Cap related
60	*TransactTime*	Copied from the `TradeCaptureReport` if present, or defaulted by Cboe.
63	*SettlType*	Omitted if relevant value is `0` (Regular). Otherwise, will be `6` (Future) with the relevant date included in *SettlDate* (64).
64	*SettlDate*	Copied from incoming `TradeCaptureReport`, if present.
75	*TradeDate*	Copied from the `TradeCaptureReport` if present, or defaulted by Cboe.
150	*ExecType*	Copied from the incoming `TradeCaptureReport`.
207	*SecurityExchange*	Copied from the incoming `TradeCaptureReport`, if present.
381	*GrossTradeAmt*	Copied from the incoming `TradeCaptureReport`, if present and valid.
487	*TradeReportTransType*	Copied from the incoming `TradeCaptureReport`.
571	*TradeReportID*	Copied from the incoming `TradeCaptureReport`.
572	*TradeReportRefID*	Unique identifier for the trade report as provided by Cboe
574	*MatchType*	Copied from the incoming `TradeCaptureReport`.
730	*SettlPrice*	Copied from the incoming `TradeCaptureReport`, if present.
820	*TradeLinkID*	Copied from the incoming `TradeCaptureReport`, if present.
828	*TrdType*	Copied from the incoming `TradeCaptureReport`.
829	*TrdSubType*	Copied from the incoming `TradeCaptureReport`, if present.
855	*SecondaryTrdType*	Copied from the incoming `TradeCaptureReport`, if present.
856	*TradeReportType*	Copied from the incoming `TradeCaptureReport`.
939	*TrdRptStatus*	Will be `0` (Accepted) or `1` (Rejected)
1003	*TradeID*	Copied from the incoming `TradeCaptureReport`, if present.
1115	*OrderCategory*	Copied from the incoming `TradeCaptureReport`, if present.
1116	*NoRootPartyIDs*	Copied from the incoming `TradeCaptureReport`, if present.

Repeating Group *RootParties* must occur the number of times specified in *NoRootPartyIDs* (1116)

1117	*RootPartyID*	Applicable when *TradeHandlingInst* (1123) = `0`, the third party responsible for submitting the pre-matched trade. Will be 'LISX' for the LISX service.
1118	*RootPartyIDSource*	Applicable when *TradeHandlingInst* (1123) = `0`. Possible values: `D` = Proprietary / Custom Code `G` = MIC (**Drop only**)
1119	*RootPartyRole*	Applicable when *TradeHandlingInst* (1123) = `0`. Specifies the role of the third party to the trade. Possible values: `6` = IntroducingFirm `64` = MultilateralTradingFacility (**Drop only**)

1123	*TradeHandlingInstr*	Copied from the incoming `TradeCaptureReport`.

1390	*TradePublishIndicator*	Copied from the incoming `TradeCaptureReport`.
1430	*VenueType*	Copied from the incoming `TradeCaptureReport`.
1838	*NoTradePriceConditions*	Copied from the incoming `TradeCaptureReport`.
Repeating Group *TradePriceConditionGrp* must occur the number of times specified in *NoTradePriceConditions* (1838)		
…	…	Entire block copied from incoming TradeCaptureReport, although order may be adjusted
2373	*IntraFirmTradeIndicator*	This field is valid from 1 Jan 2024, and indicates if a transaction is between entities within the same group carried out for intra group risk management purposes to transfer risk accumulated as a result of previously executed transactions or where necessary to comply with regulatory requirements. It is set to `Y` if it is an intra-group transaction.
2405	*ExecMethod*	Copied from the incoming `TradeCaptureReport`, if present.
2667	*AlgorithmicTradeIndicator*	Copied from the incoming `TradeCaptureReport`, if present.
2896	*TertiaryTrdType*	Copied from the incoming `TradeCaptureReport`, if present.
9128	*Tolerance*	Copied from the incoming `TradeCaptureReport`, if present.
552	*NoSides*	Copied from the incoming `TradeCaptureReport`. **Drop only.** When *TradeHandlingInst* (1123) = 0 and the configured third party requires counterparty anonymisation, will be limited to the number of sides your Drop profile permits you access.

Repeating Group *TrdCapRptSideGrp* must occur the number of times specified in *NoSides* (552)		
...	...	Entire block copied from incoming TradeCaptureReport, although order may be adjusted
775	*BookingType*	**Drop only**. Possible values: 0 = Regular (default) 1 = CFD (Contract for Difference)
453	*NoPartyIDs*	The number of parties involved in the trade. Conventionally, will be copied from the incoming `TradeCaptureReport` with a value of 1. On **Drop only**, may be 2, 3 or 4.
Repeating Group *Parties* must occur the number of times specified in *NoPartyIDs* (453)		
...	...	Generally, entire block copied from incoming TradeCaptureReport. Potential exceptions detailed below
448	*PartyID*	The end-client responsible for the trade. Will be an identifier (4 or 5 (**Drop only**)) uppercase letters, except when *PartyRole* (452) = 14, where it may be up to 16 characters (**Drop only**).
452	*PartyRole*	Specifies the role of the party to the trade. At this time, only the following values are valid: 1 = ExecutingFirm 3 = Client ID (**Drop only**) 6 = IntroducingFirm (**Drop only**) 7 = EnteringFirm 14 = GiveupClearingFirm (**Drop only**) 17 = ContraFirm
802	*NoPartySubIDs*	**Drop only**. Only relevant when *PartyRole* (452) = 3. The number of sub-parties involved. If set, must always be 1
Repeating Group *PtysSubGrp* must occur the number of times specified in *NoPartySubIDs* (802)		
523	*PartySubID*	**Drop only**. Additional information about the end-client responsible for the trade. Only valid when *PartyRole* (452) = 3.
803	*PartySubIDType*	**Drop only.** Only valid when *PartyRole* (452) = 3. Acceptable values: 4001 = BatsClientIdForIntroducingBroker
9688	*OrigCompID*	**Drop only.** *TargetCompID* (56) of original FIX TradeCaptureReport from Cboe to the Participant. Drop port must be configured to send this optional field.
9689	*OrigSubID*	**Drop only.** *TargetSubID* (57) of original FIX TradeCaptureReport from Cboe to the Participant. Drop port must be configured to send this optional field.
	Standard Message Trailer	

6.7 Trade Capture Report

The Trade Capture Report is sent from Cboe to the participant in order to confirm that a Trade Capture Report has been fully processed. It is a business-level confirmation as distinct from the technology level acknowledgement sent as a Trade Capture Report Ack.

The majority of the fields in this message are copied from the incoming `TradeCaptureReport`. When using the *One-Party Report for Matching* model, it is important to note that the majority of these fields will be sourced from the party reporting the sell leg. Also on this model, the *Account*1 field for the contra leg will not be populated.

Tag	Name	Description
	Standard Message Header	*MsgType* (35) = `AE`
15	*Currency*	Copied from the incoming `TradeCaptureReport`, if present.
22	*IDSource*	Copied from the incoming `TradeCaptureReport`, if present.
30	*LastMkt*	Populated with the segment MIC of this fill.
31	*LastPx*	Traded Price.
32	*LastShares*	Copied from the incoming `TradeCaptureReport`.
48	*SecurityID*	Copied from the incoming `TradeCaptureReport`, if present.
55	*Symbol*	Copied from the incoming `TradeCaptureReport`, if present.
58	*Text*	If present, indicates reason for the message. Format is one letter reason code followed by colon and space followed by free form text message. Reason codes are: `A` = Admin `D` = Duplicate *TradeReportID* `H` = Halted `U` = User Requested `Y` = Symbol Not Supported `X` = Expired `Z` = Unforseen Reason `m` = Market Access Risk Limit Exceeded `o` = Max Open Trade Count Exceeded `v` = MiFID II Double Cap related
60	*TransactTime*	Copied from the `TradeCaptureReport` if present, or defaulted by Cboe.
63	*SettlType*	Omitted if relevant value is `0` (Regular). Otherwise, will be `6` (Future) with the relevant date included in *SettlDate* (64).
64	*SettlDate*	Copied from incoming `TradeCaptureReport`, if present.
75	*TradeDate*	Copied from the `TradeCaptureReport` if present, or defaulted by Cboe.
150	*ExecType*	Copied from the incoming `TradeCaptureReport`.
207	*SecurityExchange*	Copied from the incoming `TradeCaptureReport`, if present.
375	*ContraBroker*	`BATS`: Trade Reported on BXE book `CHIX`: Trade Reported on CXE book `CEUX`: Trade Reported on DXE book
487	*TradeReportTransType*	Copied from the incoming `TradeCaptureReport`.
571	*TradeReportID*	Unique identifier for the trade report confirm as provided by Cboe.
572	*TradeReportRefID*	Contains the *TradeReportID* (571) of the original trade capture report to which this message relates
573	*MatchStatus*	Will be `0` (Matched) for confirm, and `1` (Unmatched) for a decline
574	*MatchType*	Copied from the incoming `TradeCaptureReport`.
730	*SettlPrice*	Copied from the incoming `TradeCaptureReport`, if present.
820	*TradeLinkID*	Copied from the incoming `TradeCaptureReport`, if present.
828	*TrdType*	Copied from the incoming `TradeCaptureReport`.

829	*TrdSubType*	Copied from the incoming `TradeCaptureReport`, if present.
855	*SecondaryTrdType*	Copied from the incoming `TradeCaptureReport`, if present.
856	*TradeReportType*	Will be 2 (Accept) for a confirm, 3 (Decline) for a decline and `0` (Submit) for an unsolicited change.
1003	*TradeID*	ID representing the trade, as seen on outbound market data. Allocated by Cboe. Required to amend, cancel or release a report. To derive TVTIC (Trading Venue Transaction Identification Code) from an *ExecID* as needed in Cboe Transaction Reporting, please refer to the Cboe Participant Manual.
1115	*OrderCategory*	Copied from the incoming `TradeCaptureReport`, if present.
1116	*NoRootPartyIDs*	Copied from the incoming `TradeCaptureReport`, if present.
Repeating Group *RootParties* must occur the number of times specified in *NoRootPartyIDs* (1116)		
...	...	Entire block copied from incoming TradeCaptureReport, although order may be adjusted
1123	*TradeHandlingInst*	Copied from the incoming `TradeCaptureReport`.
1390	*TradePublishIndicator*	Will be 2 (Deferred Publication) if deferment is requested and the trade is eligible for such. Otherwise, copied from the incoming `TradeCaptureReport`.
1430	*VenueType*	Copied from the incoming `TradeCaptureReport`.
1838	*NoTradePriceConditions*	Indicates the number of *TradePriceCondition* (1839) fields, if present.
Repeating Group *TradePriceConditionGrp* must occur the number of times specified in *NoTradePriceConditions* (1838)		
1839	*TradePriceCondition*	Mostly copied from the incoming TradeCaptureReport, with exceptions. (see below) Indicate values in MMT v3/v4.1 levels 3.6 and 3.8 For MMT Level 3.6 'Special Divided Indicator', supported values are: `13` = Special Dividend (SDIV) For MMT Level 3.8 'Contribution to Price Formation or the Price Discovery Process', supported values are: `15` = Non-Price Forming Trade (NPFT) If the value 16 is used in the incoming TradeCaptureReport, then it will not be in this confirm message. Instead, *TrdRegPublicationReasons* (8013) = 2 will be set if appropriate.
2373	*IntraFirmTradeIndicator*	This field is valid from 1 Jan 2024, and indicates if a transaction is between entities within the same group carried out for intra group risk management purposes to transfer risk accumulated as a result of previously executed transactions or where necessary to comply with regulatory requirements. It is set to `Y` if it is an intra-group transaction.
2405	*ExecMethod*	Copied from the incoming `TradeCaptureReport`, if present.
2667	*AlgorithmicTradeIndicator*	Copied from the incoming `TradeCaptureReport`, if present.
2896	*TertiaryTrdType*	Copied from the incoming `TradeCaptureReport`, if present.

8013	*TrdRegPublication Reasons*	If present, indicates the pre-trade waiver or deferral derived by Cboe for this trade. Valid values are: 0 = Negotiated Trade in Liquid Instrument (NLIQ) 1 = Negotiated Trade in Illiquid Instrument (OILQ) 2 = Negotiated Trade Subject to Conditions Other Than the Current Market Price (PRIC) 6 = Deferral for Large in Scale (LRGS) 9 = Large In Scale (Pre-Trade Transparency Waiver) (Will also be used to indicate NTLS, effectie 29 Apr 2024, for BXE and CXE) 17 = Negotiated Trade Subject to a Pre-Trade Transparency Waiver (replaces NLIQ, OLIQ, and PRIC, effective 29 Apr 2024, for BXE and CXE only)(NETW) Multiple values in this field will be separated by spaces.
7570	*RptTime*	Indicates the time at which a deferred trade report will be automatically published. Where *RptTime* falls outside of the systems operating time, the report will be published during operating hours on the next trading day. When no deferral is requested, or when the trade does not qualify for a deferral, any time returned will match *TransactTime* (60). Microsecond level precision.
9688	*OrigCompID*	**Drop only.** *TargetCompID* (56) of original FIX TradeCaptureReport from Cboe to the Participant. Drop port must be configured to send this optional field.
9689	*OrigSubID*	**Drop only.** *TargetSubID* (57) of original FIX TradeCaptureReport from Cboe to the Participant. Drop port must be configured to send this optional field.
552	*NoSides*	Copied from the incoming `TradeCaptureReport`. **Drop only.** When *TradeHandlingInst* (1123) = 0 and the configured third party requires counterparty anonymisation, will be limited to the number of sides your Drop profile permits you access.

Repeating Group *TrdCapRptSideGrp* must occur the number of times specified in *NoSides* (552)

...	...	Entire block copied from incoming TradeCaptureReport, although order may be adjusted
1427	*SideExecID*	Available on an opt-in basis. Side unique version of the day-unique *TradeID* (1003).
7772	*CentralCounterparty*	The CCP handling this trade leg for a confirm, and not present for a decline: EMCF = European Multilateral Clearing Facility LCHL = LCH.Clearnet XCLR = SIX x-clear NONE = Clearing Suppressed
9882	*FeeCode*	Specific fee code associated with the trade. See the Fee Schedule for the respective market for possible values.
775	*BookingType*	**Drop only**. Possible values: 0 = Regular (default) 1 = CFD (Contract for Difference)
453	*NoPartyIDs*	The number of parties involved in the trade. Conventionally, will be copied from the incoming `TradeCaptureReport` with a value of `1`. On **Drop only**, may be 2, 3 or 4.

Repeating Group *Parties* must occur the number of times specified in *NoPartyIDs* (453)

...	...	Generally, entire block copied from incoming TradeCaptureReport. Potential exceptions detailed below
448	*PartyID*	The end-client responsible for the trade. Will be an identifier (4 or 5 (**Drop only**)) uppercase letters, except when *PartyRole* (452) = 14, where it may be up to 16 characters (**Drop only**).
452	*PartyRole*	Specifies the role of the party to the trade. At this time, only the following values are valid: 1 = ExecutingFirm 3 = Client ID (**Drop only**) 6 = IntroducingFirm (**Drop only**) 7 = EnteringFirm 14 = GiveupClearingFirm (**Drop only**) 17 = ContraFirm
802	*NoPartySubIDs*	**Drop only**. Only relevant when *PartyRole* (452) = 3. The number of sub-parties involved. If set, must always be 1

Repeating Group *PtysSubGrp* must occur the number of times specified in *NoPartySubIDs* (802)

523	*PartySubID*	**Drop only**. Additional information about the end-client responsible for the trade. Only valid when *PartyRole* (452) = 3.
803	*PartySubIDType*	**Drop only.** Only valid when *PartyRole* (452) = 3. Acceptable values: 4001 = BatsClientIdForIntroducingBroker

	Standard Message Trailer	

7 Example Messages and Message Flow

7.1 New Trade Capture Reports

Below illustrates an example of key elements of the message flow for a new trade capture report.

Participant to Cboe:

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `CLIENTID111` - Day-unique ID chosen by client
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `0` - Submit

Cboe to Participant (Technical Ack) - if accepted:

- *MsgType* (35) = `AR`
- *TrdRptStatus* (939) = `0` - Accepted
- *TradeReportID* (571) = `CLIENTID111` - Day-unique ID chosen by client copied from the incoming trade capture report.
- *TradeReportRefID* (572) = `WXYZ1234` - Unique identifier for the trade capture report provided by Cboe.
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `0` - Submit

Cboe to Participant (Business Ack)

- *MsgType* (35) = `AE`
- *TradeReportID* (571) = `WXYZ1234` - Unique identifier for the trade report confirm as provided by Cboe.
- *TradeReportRefID* (572) = `CLIENTID111` - Contains the TradeReportID of the original trade capture report to which this message relates.
- *TradeID* (1003) = `ABCD1234` - Represents the trade as seen on outbound market data allocated by Cboe. Requires to amend, cancel or release a report
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `2` - Accept

TradeID (1003) = `is` a Cboe allocated ID as per the MiFID II definition of a Transaction Identification Code. This is the ID seen on outbound market data (Trade Message, Extended Trade Message and Trade Message Unknown).

7.2 Trade Capture Report Cancellations

Below illustrates an example of key elements of the message flow for cancellation of previously confirmed trade captures.

Participant to Cboe:

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeID* (1003) = `ABCD1234` - The TradeID for the confirmed trade report
- *TradeReportTransType* (487) = `1` - Cancel

Cboe to Participant (Technical Ack) - if rejected:

- *MsgType* (35) = `AR`
- *TrdRptStatus* (939) = `1` - Rejected
- *Text* (58) = `Reason for reject`

Cboe to Participant (Technical Ack) - if accepted:

- *MsgType* (35) = `AR`
- *TrdRptStatus* (939) = `0` - Accepted

Cboe to Participant - if cancellation is declined:

- *MsgType* (35) = `AE`
- *TradeReportTransType* (487) = `1` - Cancel
- *TradeReportType* (856) = `3` - Decline
- *Text* (58) = `Reason for decline`

Cboe to Participant - if cancellation is confirmed:

- *MsgType* (35) = `AE`
- *TradeReportTransType* (487) = `1` - Cancel
- *TradeReportType* (856) = `2` - Accept

7.3 Trade Capture Report Amendments

Below illustrates an example of key elements of the message flow for amendment of previously confirmed trade captures.

Participant to Cboe:

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeID* (1003) = `ABCD1234` - The TradeID for the confirmed trade report
- *TradeReportTransType* (487) = `2` - Replace

Cboe to Participant (Technical Ack) - if rejected:

- *MsgType* (35) = `AR`
- *TrdRptStatus* (939) = `1` - Rejected
- *Text* (58) = `Reason for reject`

Cboe to Participant (Technical Ack) - if accepted:

- *MsgType* (35) = `AR`
- *TrdRptStatus* (939) = `0` - Accepted

Cboe to Participant - if amendment is declined:

- *MsgType* (35) = `AE`
- *TradeReportTransType* (487) = `2` - Replace
- *TradeReportType* (856) = `3` - Decline
- *Text* (58) = `Reason for decline`

Cboe to Participant - if amendment is confirmed:

- *MsgType* (35) = `AE`
- *TradeReportTransType* (487) = `2` - Replace
- *TradeReportType* (856) = `2` - Accept

7.4 Deferred Publication Trade Reports

Below illustrates an example of key elements of the message flow for deferred publication of trades captures.

Participant to Cboe:

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradePublishIndicator* (1390) = `2` - Deferred Publication
- *TransactTime* (60) = `Time of Trade` - As long as this is within the acceptable deferment period, the trade will not be regarded as late

Cboe to Participant (Technical Ack) - if rejected:

- *MsgType* (35) = `AR`
- *TrdRptStatus* (939) = `1` - Rejected
- *Text* (58) = `Reason for reject`

Cboe to Participant (Technical Ack) - if accepted:

- *MsgType* (35) = `AR`
- *TrdRptStatus* (939) = `0` - Accepted

Cboe to Participant - if report is declined:

- *MsgType* (35) = `AE`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `3` - Decline
- *Text* (58) = `Reason for decline`

Cboe to Participant - if report is confirmed and deferment permitted:

- *MsgType* (35) = `AE`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `2` - Accept
- *TradePublishIndicator* (1390) = `2` - Deferred Publication

Cboe to Participant - if report is confirmed and deferment is not permitted:

- *MsgType* (35) = `AE`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `2` - Accept
- *TradePublishIndicator* (1390) = `1` - Publish trade Immediately
- *Text* (58) = `A: Trade accepted, but ineligible for deferment` - Exact text may vary

Then, once the Participant would like the deferred trade released to the market (note - this may be immediately if the Participant has held onto the trade for the deferment period).

Participant to Cboe:

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeID* (1003) = `ABCD1234` - The TradeID for the confirmed trade report
- *TradeReportTransType* (487) = `3` - Release

Cboe to Participant (Technical Ack) - if rejected:

- *MsgType* (35) = `AR`
- *TrdRptStatus* (939) = `1` - Rejected

- *Text* (58) = `Reason for reject`

Cboe to Participant (Technical Ack) - if accepted:

- *MsgType* (35) = `AR`
- *TrdRptStatus* (939) = `0` - Accepted

Cboe to Participant - if release is declined:

- *MsgType* (35) = `AE`
- *TradeReportTransType* (487) = `3` - Release
- *TradeReportType* (856) = `3` - Decline
- *Text* (58) = `Reason for decline`

Cboe to Participant - if release is confirmed:

- *MsgType* (35) = `AE`
- *TradeReportTransType* (487) = `3` - Release
- *TradeReportType* (856) = `2` - Accept

7.5 ETR Matched Trade Report

Below illustrates an example of an ETR submitted for matching with a counterparty. In this case, you might be the ABCD party.

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `1234`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `0` - Submit
- *TradeHandlingInstr* (1123) = `2` - ETR Matching
- *TradePublishIndicator* (1390) = `1` - Publish Trade Immediately
- *NoSides* (552) = `2`
- *Side* (54) = `1` - Buy
- *NoPartyIDs* (453) = `1`
- *PartyID* (448) = `ABCD`
- *PartyRole* (452) = `7` - EnteringFirm
- *Side* (54) = `2` - Sell
- *NoPartyIDs* (453) = `1`
- *PartyID* (448) = `XYZZ`
- *PartyRole* (452) = `17` - ContraFirm

Then, the counterparty (XYZZ in this case), would enter a similar report, but with party information reversed (except for PartyRoles):

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `1234`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `0` - Submit
- *TradeHandlingInstr* (1123) = `2` - ETR Matching
- *TradePublishIndicator* (1390) = `1` - Publish Trade Immediately
- *NoSides* (552) = `2`
- *Side* (54) = `2` - Sell
- *NoPartyIDs* (453) = `1`
- *PartyID* (448) = `XYZZ`
- *PartyRole* (452) = `7` - EnteringFirm
- *Side* (54) = `1` - Buy
- *NoPartyIDs* (453) = `1`
- *PartyID* (448) = `ABCD`
- *PartyRole* (452) = `17` - ContraFirm

7.6 Withdrawing an ETR Matching Instruction

Below illustrates an example of an ETR submitted for matching with a counterparty.

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `YOUR1234`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `0` - Submit
- *TradeHandlingInstr* (1123) = `2` - ETR Matching

When we acknowledge your matching instruction, you will receive a message similar to:

- *MsgType* (35) = `AR` - Trade Capture Report Ack
- *TradeReportID* (571) = `YOUR1234`
- *TradeReportRefID* (572) = `BATS1234`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `0` - Submit
- *TradeHandlingInstr* (1123) = `2` - ETR Matching

Should you need to withdraw the matching instruction, you should send in a trade report similar to:

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `YOUR3456`
- *TradeReportRefID* (572) = `BATS1234`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `6` - TradeReportCancel
- *TradeHandlingInstr* (1123) = `2` - ETR Matching

If successful, you will then receive an acknowledgement of your withdrawal request:

- *MsgType* (35) = `AR` - Trade Capture Report Ack
- *TradeReportID* (571) = `YOUR3456`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `6` - TradeReportCancel
- *TradeHandlingInstr* (1123) = `2` - ETR Matching

Followed by a decline of your original report:

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `BATS1234`
- *TradeReportRefID* (572) = `YOUR1234`
- *TradeReportTransType* (487) = `0` - New
- *TradeReportType* (856) = `3` - Decline
- *TradeHandlingInstr* (1123) = `2` - ETR Matching
- *Text* (58) = `U: A Message` - 'U' indicates User

7.7 Withdrawing an ETR Matched Cancellation Instruction

Below illustrates an example of an ETR cancellation instruction submitted for matching with a counterparty.

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `YOUR1234`
- *TradeID* (1003) = `0001234567`
- *TradeReportTransType* (487) = `1` - Cancel
- *TradeReportType* (856) = `0` - Submit
- *TradeHandlingInstr* (1123) = `2` - ETR Matching

When we acknowledge your cancellation instruction, you will receive a message similar to:

- *MsgType* (35) = `AR` - Trade Capture Report Ack
- *TradeReportID* (571) = `YOUR1234`
- *TradeReportRefID* (572) = `BATS1234`
- *TradeID* (1003) = `0001234567`
- *TradeReportTransType* (487) = `1` - Cancel
- *TradeReportType* (856) = `0` - Submit
- *TradeHandlingInstr* (1123) = `2` - ETR Matching

Should you need to withdraw the cancellation instruction, you should send in a trade report similar to:

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `YOUR3456`
- *TradeReportRefID* (572) = `BATS1234`
- *TradeID* (1003) = `0001234567`
- *TradeReportTransType* (487) = `1` - Cancel
- *TradeReportType* (856) = `6` - TradeReportCancel
- *TradeHandlingInstr* (1123) = `2` - ETR Matching

If successful, you will then receive an acknowledgement of your withdrawal request:

- *MsgType* (35) = `AR` - Trade Capture Report Ack
- *TradeReportID* (571) = `YOUR3456`
- *TradeID* (1003) = `0001234567`
- *TradeReportTransType* (487) = `1` - Cancel
- *TradeReportType* (856) = `6` - TradeReportCancel
- *TradeHandlingInstr* (1123) = `2` - ETR Matching

Followed by a decline of your original instruction:

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `BATS1234`
- *TradeReportRefID* (572) = `YOUR1234`
- *TradeID* (1003) = `0001234567`
- *TradeReportTransType* (487) = `1` - Cancel
- *TradeReportType* (856) = `3` - Decline
- *TradeHandlingInstr* (1123) = `2` - ETR Matching
- *Text* (58) = `U: A Message` - 'U' indicates User

7.8 ETR Matched Trade Report with Consideration and Tolerance

Below illustrates an example of an ETR submitted for matching with a counterparty, where a consideration is specified instead of a price and a tolerance is allowed.

- *MsgType* (35) = `AE` - Trade Capture Report
- *TradeReportID* (571) = `1234`
- *TradeReportTransType* (487) = `0`
- *TradeReportType* (856) = `0`
- *VenueType* (1430) = `0` - Off Book
- *MatchType* (574) = `3` - trade reporting (on-exchange)
- *TradePriceCondition* (1839) = `16` - Negotiated Trade Subject to Conditions Other than the current Market Price (PRIC)
- *TradeHandlingInstr* (1123) = `2` - ETR Matching
- *Currency* (15) = `GBX`
- *LastShares* (32) = `100000`
- *GrossTradeAmt* (381) = `10000000000` - GBX 10B or GBP 100M
- *Tolerance* (9128) = `1000` - GBX 1000 or GBP 10
- *NoSides* (552) = `2`
- *Side* (54) = `1` - Buy
- *NoPartyIDs* (453) = `1`
- *PartyID* (448) = `ABCD`
- *PartyRole* (452) = `7` - EnteringFirm
- *Side* (54) = `2` - Sell
- *NoPartyIDs* (453) = `1`
- *PartyID* (448) = `XYZZ`
- *PartyRole* (452) = `17` - ContraFirm

7.9 ETR Matched Trade Report with Price and Tolerance

Below illustrates an example of an ETR submitted for matching with a counterparty, where a price is specified rather than consideration and a tolerance is allowed.

- *TradeHandlingInstr* (1123) = `2` - ETR Matching
- *Currency* (15) = `GBX`
- *LastShares* (32) = `100000`
- *LastPrice* (31) = `100000` - yielding consideration of GBP 100M
- *Tolerance* (9128) = `1000` - GBX 1000 or GBP 10

7.10 Third Party Confirmed Trades

Below illustrates an example of key elements of the message flow when confirmed trades are reported via a third party (whether a broker or a vendor). Generally, participants will only see these variants of messages if utilising ODROP as the original incoming message will be sent by the third party.

Information on sides of the trade that the customer does not have authorisation to view may be anonymised should the third party require such.

Participant to Cboe:

- *MsgType* (35) = `AE`
- *TradeHandlingInstruction* (1123) = `0` - Confirmed
- *NoRootPartyIDs* (1116) = `1`
- *RootPartyID* (1117) = `3PBV` - A code that identifies the third party
- *RootPartyIDSource* (1118) = `D` - Proprietary / Custom Code
- *RootPartyRole* (1119) = `6` - IntroducingFirm
- *NoSides* (552) = `2`
- *Side* (54) = `1` - Buy
- *PartyID* (448) = `PTY1` - A code that identifies the 1^{st} party to the trade
- *Side* (54) = `2` - Sell
- *PartyID* (448) = `PTY2` - A code that identifies the 2^{nd} party to the trade

Cboe to Participant - if confirmed:

- *MsgType* (35) = `AE`
- *TradeHandlingInstruction* (1123) = `0` - Confirmed
- *TradeReportType* (856) = `2` - Accept
- *TradeID* (1003) = `ABCD1234` - The TradeID for the confirmed trade report
- *NoRootPartyIDs* (1116) = `1`
- *RootPartyID* (1117) = `3PBV` - A code that identifies the third party
- *RootPartyIDSource* (1118) = `D` - Proprietary / Custom Code
- *RootPartyRole* (1119) = `6` - IntroducingFirm
- *NoSides* (552) = `2`
- *Side* (54) = `1` - Buy
- *PartyID* (448) = `PTY1` - A code that identifies the 1^{st} party to the trade
- *Side* (54) = `2` - Sell
- *PartyID* (448) = `PTY2` - A code that identifies the 2^{nd} party to the trade

7.11 Cboe BIDS Trade Confirmations

Trade Reports associated with the Cboe BIDS service follow a very similar message format to a standard Trade Report, with the only differences being an alternative set of Party Roles in the *PartyRole* (452), in addition to a new *BookingType* (775) tag and *FeeCode* (9882) tag. These are described in the following sub-sections.

7.11.1 Party Roles

Trade Reports can have a number of parties associated with each leg. Each is described in the Trade Capture Report via the repeating group Parties, and the *PartyRole* (452) field within each Party group indicates the role of that party on this leg of the trade. The possible party roles are summarised below.

PartyRole (452)	Description
1 (Executing Broker)	Cboe Firm Id for the Designated Broker
3 (Client Id)	BIDS Client Id, identifying the buy-side client.
6 (Introducing Firm)	BIDS Firm Id for the Introducing Firm
14 (Giveup Firm)	BIDS Firm Id for the Give-Up Firm

Executing Broker (452=1)

This firm Id specifies the Cboe participant acting as a Designated Broker for a Cboe BIDS client, or a sell-side firm trading on their own behalf. This is a 4-character alphanumeric field, assigned by Cboe.

Client Id (452=3)

The 5-character BIDS client identifier representing the buy-side client. This Party block may also include a PartySubID, containing an alternate identifier up to 16 alpha-numeric characters.

The Client Id and corresponding sub-ID will only only be present in Cboe CXE ETRs where it is appropriate to include it on the Designated Broker's drop feed. (This depends on the relationship between the Introducing Firm and the Designated Broker, and is managed within Cboe BIDS.)

Introducing Firm (452=6)

A 4-character Introducing Firm is allocated by the Cboe BIDS service, and will always be present, though may represent the same firm as the Designated Broker.

Giveup Firm (452=14)

A value of up to 16-characters (but typically 4) which is passed through from the buy-side OMS to Cboe BIDS, which instructs the Designated Broker to 'give up' the trade to the specified broker. This Party block will only be present when applicable. When present, the value of the *BookingType* (775) field takes significance, indicating whether the give-up is to be in the form a CFD.

7.11.2 Booking Type and Fee Code

The *BookingType* (775) tag can have the following possible values. If the tag isn't present then a default value of Regular can be assumed.

0 = Regular
1 = CFD (Contract for Difference)

The *FeeCode* (9882) tag describes the category of fee applicable to the Cboe BIDS trade. Valid values are:

L1 = Designated Broker, clearing for a buyside firm.
L2 = Executing Broker, trading in their own name.

7.11.3 Examples

Below illustrates examples of key elements of the trade confirmations from trades resulting from use of the Cboe BIDS service. Participants will only see these variants of messages if utilising ODROP. Information on sides of the trade that the customer does not have authorisation to view will be anonymised.

Designated Broker - own flow:

Suppose broker Acme Corporation was trading on their own behalf. They may have a Cboe Firm Id of ‘ACME’, and a BIDS Firm Id of ‘ACMC’. We would expect to see their leg of any trade with:

NoPartyIDs (453) = 2
PartyRole (452) = 1 - **ExecutingFirm** *PartyID* (448) = `ACME`
PartyRole (452) = 6 - **IntroducingFirm** *PartyID* (448) = `ACMC`

Buy Side firm using Designated Broker:

Suppose buy side firm FMRCO was using broker Acme Corporation, with the above identifiers, we would expect to see their leg of any trade with:

NoPartyIDs (453) = 3
PartyRole (452) = 1 - **ExecutingFirm** *PartyID* (448) = `ACME`
PartyRole (452) = 6 - **IntroducingFirm** *PartyID* (448) = `ACMC`
PartyRole (452) = 3 - **ClientID** *PartyID* (448) = `FMRCO`.

Optionally, if FMRCO is known to Acme Corporation as FMR-LISX, we would also see *PartySubID* (523) = `FMR-LISX` along with other supporting tags as shown in later examples.

Buy Side firm using Introducing Broker:

Suppose buy side firm FMRCO was using Introducing Broker BROK, who was using broker Acme Corporation as their Designated Broker, we would expect to see their leg of any trade with:

NoPartyIDs (453) = 3
PartyRole (452) = 1 - **ExecutingFirm** *PartyID* (448) = `ACME`
PartyRole (452) = 6 - **IntroducingFirm** *PartyID* (448) = `BROK`
PartyRole (452) = 3 - **ClientID** *PartyID* (448) = `FMRCO`.

Optionally, if FMRCO is known to their introducing broker as FMR-LISX, we would also see *PartySubID* (523) = `FMR-LISX` along with other supporting tags as shown in later examples.

7.11.4 Standard Trades

Cboe to Participant - with anonymised side:

- *MsgType* (35) = AE
- *TradeHandlingInstruction* (1123) = 0 - Confirmed
- *TradeReportType* (856) = 2 - Accept
- *TradeID* (1003) = ABCD1234 - The TradeID for the confirmed trade report
- *NoRootPartyIDs* (1116) = 1
- *RootPartyID* (1117) = LISX
- *RootPartyIDSource* (1118) = G - MIC
- *RootPartyRole* (1119) = 64 - MultilateralTradingFacility
- *NoSides* (552) = 1
- *Side* (54) = 1 - Buy
- *FeeCode* (9882) = L1 - Identifies the fee category
- *NoPartyIDs* (54) = 3
- *PartyID* (448) = PTY1 - A four character code that identifies the 1^{st} Designated Broker
- *PartyRole* (452) = 1 - ExecutingFirm
- *PartyID* (448) = PTY2 - A four character code that identifies the 1^{st} Introducing Broker
- *PartyRole* (452) = 6 - IntroducingFirm
- *PartyID* (448) = PTY3 - A four or five character code that identifies the 1^{st} Client
- *PartyRole* (452) = 3 - ClientID
- *NoPartySubIDs* (802) = 1
- *PartySubID* (523) = PTY2-PTY3 - Up to 16 characters identifying the Client to the 1^{st} Introducing Broker
- *PartySubIDType* (803) = 4001 - BatsClientIdForIntroducingBroker

7.11.5 CFD Booking with Give Up

Cboe to Participant - with anonymised side:

- *MsgType* (35) = `AE`
- *TradeHandlingInstruction* (1123) = `0` - Confirmed
- *TradeReportType* (856) = `2` - Accept
- *TradeID* (1003) = `ABCD1234` - The TradeID for the confirmed trade report
- *NoRootPartyIDs* (1116) = `1`
- *RootPartyID* (1117) = `LISX`
- *RootPartyIDSource* (1118) = `G` - MIC
- *RootPartyRole* (1119) = `64` - MultilateralTradingFacility
- *NoSides* (552) = `1`
- *Side* (54) = `1` - Buy
- *FeeCode* (9882) = `L1` - Identifies the fee category
- *BookingType* (775) = `1` - CFD
- *NoPartyIDs* (54) = `4`
- *PartyID* (448) = `PTY1` - A four character code that identifies the 1^{st} Designated Broker
- *PartyRole* (452) = `1` - ExecutingFirm
- *PartyID* (448) = `PTY2` - A four character code that identifies the 1^{st} Introducing Broker
- *PartyRole* (452) = `6` - IntroducingFirm
- *PartyID* (448) = `PTY3` - A four or five character code that identifies the 1^{st} Client
- *PartyRole* (452) = `3` - ClientID
- *NoPartySubIDs* (802) = `1`
- *PartySubID* (523) = `PTY2-PTY3` - Up to 16 characters identifying the Client to the 1^{st} Introducing Broker
- *PartySubIDType* (803) = `4001` - BatsClientIdForIntroducingBroker
- *PartyID* (448) = `PTY4` - Up to 16 characters that identifies the 1^{st} Give Up Broker
- *PartyRole* (452) = `14` - GiveupClearingFirm

8 Common Session Level Issues

Cboe uses FIX 4.2 as specified by the FPL Document Version 4.2 (with Errata 20010501) with business level extensions as described in this document. The session level of the FPL specification is followed as closely as possible.

The version with errata cleared up many session level ambiguities present in the earlier version 4.2 (March 1, 2000). The following sections emphasize a few common problem areas in implementations of the FIX session protocol.

Typographical conventions:

- Anchor locations in the FPL document are shown in blue.
- Text in **bold** was emphasized in the original FPL specification.
- Emphasis added by Cboe is shown in purple.
- Notes added by Cboe are shown in green.

8.1 Ordered Message Processing

From Financial Information Exchange Protocol/FIX Message Format and Delivery/Ordered Message Processing:

> The FIX protocol assumes complete ordered delivery of messages between parties. Implementers should consider this when designing message gap fill processes. Two options exist for dealing with gaps, either request all messages subsequent to the last message received or ask for the specific message missed while maintaining an ordered list of all newer messages. For example, if the receiver misses the second of five messages, the application could ignore messages 3 through 5 and generate a resend request for messages 2 through 5, **or**, preferably 2 through 0 (where 0 represents infinity). Another option would involve saving messages 3 through 5 and resending only message 2. In both cases, messages 3 through 5 should not be processing before message 2.

8.2 Logon

From Financial Information Exchange Protocol/Session Protocol/Logon:

> After the initiator has been authenticated, the acceptor will respond immediately with a confirming *Logon* message.

8.3 Message Recovery

From Financial Information Exchange Protocol/Session Protocol/Message Recovery:

> When the incoming sequence number does not match the expected number, corrective processing is required. Note that the SeqReset-**Reset** message ([Cboe: this refers only to *GapFillFlag* (123) = `N`] used only to recover from a disaster scenario vs. normal resent request processing) is an exception to this rule as it should be processed without regards to its *MsgSeqNum* (34). **If the incoming message has a sequence number less than expected and the PossDupFlag (43) is not set, it indicates a serious error. It is strongly recommended that the session be terminated and manual intervention be initiated**. If the incoming sequence number is greater than expected, it indicates that messages were missed and retransmission of the messages is requested via the *Resend Request* (see earlier section, *Ordered Message Processing*).
>
> ...

If there are consecutive administrative messages to be resent, it is suggested that only one *SeqReset-GapFill* message be sent in their place. The sequence number of the *SeqReset-GapFill* message is the next expected outbound sequence number. The *NewSeqNo* (36) field of the GapFill message contains the sequence number of the highest administrative message in the group plus 1. For example, during a Resend operation there are 7 sequential administrative messages waiting to be resent. They start with sequence number 9 and end with sequence number 15. Instead of transmitting 7 GapFill messages (which is perfectly legal, but not network friendly), a *SeqReset-GapFill* message may be sent. **The sequence number of the Gap Fill message is set to 9 because the remote side is expecting that as the next sequence number.** The *NewSeqNo* (36) field of the Gap Fill message contains the number 16, because that will be the sequence number of the next message to be transmitted.

Sequence number checking is a vital part of FIX session management. However, a discrepancy in the sequence number stream is handled differently for certain classes of FIX messages. The table below lists the actions to be taken when the incoming sequence number is greater than the expected incoming sequence number.

NOTE: In all cases except the Sequence Reset – Reset message, the FIX session should be terminated if the incoming sequence number is less than expected and the PossDupFlag (43) is not set. A Logout message with some descriptive text should be sent to the other side before closing the session.

Response by Message Type

Message Type	Action to Be Taken on Sequence # Mismatch
Logon	Must always be the first message transmitted. Authenticate and accept the connection. After sending a *Logon* confirmation back, send a *ResendRequest* if a message gap was detected in the *Logon* sequence number.

...

8.4 Resend Request

From Financial Information Exchange Protocol/Administrative Messages/Resend Request:

Note: the sending application may wish to consider the message type when resending messages; e.g., if a new order is in the resend series and a significant time period has elapsed since its original inception, the sender may not wish to retransmit the order given the potential for changed market conditions. (The Sequence Reset-Gap Fill message is used to skip message that a sender does not wish to resend.)

8.5 Sequence Reset – Gap Fill

From Financial Information Exchange Protocol/Administrative Messages/Sequence Reset (Gap Fill):

The sequence reset message is used by the sending application to reset the incoming sequence number on the opposing side. This message has two modes: "Sequence Reset – Gap Fill when *GapFillFlag* (123) is 'Y' and "Sequence Reset – Reset" when *GapFillFlag* (123) is 'N' or not present. The "Sequence Reset – Reset" mode should **only** be used to recover from a disaster situation which cannot be otherwise recovered via "Gap Fill" mode. The sequence reset message can be used in the following situations:

- During normal resend processing, the sending application may choose not to send a message (e.g., an aged order). The Sequence Reset – Gap Fill is used to mark the place of that message.

- During normal resend processing, a number of administrative messages are not resent, the Sequence Reset – Gap Fill message is used to fill the sequence gap created.

...

The sending application will initiate the sequence reset. **The message in all situations specifies the NewSeqNo (36) to reset as the value of the next sequence number immediately following the messages and/or sequence numbers being skipped.**

...

If the *GapFillFlag* (123) field is present (and equal to 'Y'), the *MsgSeqNum* (34) should conform to standard message sequencing rules (i.e., the *MsgSeqNum* (34) of the SeqReset-GapFill message should represent the beginning *MsgSeqNum* (34) in the gap fill range because the remote side is expecting that next message).

The sequence reset can only increase the sequence number. If a sequence reset is received attempting to decrease the next expected sequence number, the message should be rejected and treated as a serious error. It is possible to have multiple resend requests issued in a row (i.e., 5 to 10 followed by 5 to 11). If sequence number 8, 10, and 11 represent application messages while 5–7 and 9 represent administrative messages, the series of messages as a result of the resend request may appear as SeqReset-GapFill with *NewSeqNo* (36) of 8, message 8, SeqReset-GapFill with *NewSeqNo* (36) of 10, and message 10. This could then be followed by SeqReset-GapFill with *NewSeqNo* (36) of 8, message 8, SeqReset-GapFill with *NewSeqNo* (36) of 10, message 10, and message 11. One must be careful to ignore the duplicate SeqReset-GapFill which is attempting to lower the next expected sequence number. This can be detected by checking to see if its *MsgSeqNum* (34) is less than expected. If so, the SeqReset-GapFill is a duplicate and should be discarded.

9 FIX Drop

Cboe offers two types of FIX Drop ports (Standard FIX Drop and Order by Order FIX Drop). Both port types do not accept orders. Their purpose is to provide real time information about order flow. They may be configured to send order flow based on various combinations of information relating to specific Participants, trading firm identifiers, and/or sessions. With proper authorisation (e.g., clearing or sponsored relationships), a single FIX Drop session can be used to obtain information about multiple Participants.

9.1 Standard FIX Drop

Standard FIX drop ports only send execution information on fills (i.e., execution reports where *ExecType* (150) = `1` (Partially Filled) or `2` (Filled)). It can also be configured to send execution reports for trades resulting from Trade Capture Reports (where *ExecType* (150) = `F` (Trade)) and Trade Cancel/Correction messages (where *MsgType* (35) = `UCC`) resulting from cancels/amends.

9.2 Order By Order FIX Drop

All order message types are supported including, but not limited to:

- *ExecType* (150) = `0`: Acknowledgments
- *ExecType* (150) = `1` or `2`: Partially Filled, Filled
- *ExecType* (150) = `4`: Canceled
- *ExecType* (150) = `5`: Replaced
- *ExecType* (150) = `8`: Rejected
- *MsgType* (35) = `9`: Order Cancel Reject
- *MsgType* (35) = `UCC`: Trade Cancel/Correction (optionally, if configured at the port level)
- *MsgType* (35) = `AE`: Trade Capture Report (optionally, if configured at the port level)
- *MsgType* (35) = `AR`: Trade Capture Report Ack (optionally, if configured at the port level)

BOE rejects are not available on Order by Order FIX DROP.

If rejects or cancels are due to incomplete or incorrect clearing information, they may be unavailable on Order by Order FIX Drop ports.

Users of Order by Order FIX Drop must always be prepared to receive new/unknown FIX tag and FIX tag values for BOE/FIX ports being monitored. Cboe reserves the right to add new FIX tags and to update values distributed on Order by Order FIX Drop with no notice.

9.2.1 Cboe BIDS

When configured, Order by Order FIX Drop ports will also include Trade Confirmations from the Cboe BIDS system. These Trade Confirmation messages will contain additional fields. See section § 7.11, p. 61 for a description.

9.3 Port Options

Both types of FIX Drop ports can be optionally configured with the following features:

1. Choice of various Cboe supported symbology types.
2. Sending of Trade Cancel/Correction (*MsgType* (35) = `UCC`) messages. Please note that enabling these messages will be dependent on enabling of trade cancels/corrections on the corresponding FIX order entry ports.
3. Enable unique wash execution identifiers.

9.4 Cboe BIDS ODROP

See section § 7.11, p. 61 for a description on the Trade Confirmations which will be sent on ODROP for Cboe BIDS.

10 FIX Differences Between US and Europe

This section describes, in detail, the differences between the FIX implementations of the Cboe US Equities exchanges and Cboe Europe. The FIX session level implementation and supported messages are mostly identical between the US and Europe.

Trade Capture Reports

All messaging related to the use of Trade Capture Reports is only available in Europe.

Tick Sizes

In the United States, there is currently a single tick band for all symbols. Prices less than $1.00 have a tick size of $0.0001. Prices greater than $1.00 have a tick size of $0.01. In Europe, tick sizes vary by market, price, and symbol. Reference data files are available daily which enumerate the tick sizes.

Routing Instructions

The values which are common across the US and Europe are *RoutingInst* = `B`, `P`, `Q`.

Display Indicator

Values for the *DisplayIndicator* (9479) are different. Europe only defines *DisplayIndicator* (9479) = `X` (visible) and `I` (invisible). US Equities offers additional values for price sliding.

Bypass Hidden

The *BypassHidden* (9687) is not supported in Europe.

Trade Liquidity Indicator

Values for the *TradeLiquidityIndicator* (9730) are different. Values which are common across the US and Europe are *TradeLiquidityIndicator* (9730) = `A` (Added) and `R` (Removed). US Equities offers values for routing which Europe does not. Europe offers values for fills done in the Cboe Dark Pool which the US does not.

Discretionary Orders

For regulatory reasons, discretionary orders, specified in the US with *DiscretionAmount* (9622), are not supported in Europe.

Execution Instruction

US Equities support *ExecInst* (18) values for intermarket sweep orders (`18=f`) and Dark Scan (`18=z`) which are not supported in Europe.

Working and Display Price Fields

The *WorkingPrice* (9690) and *InitialDisplayPrice* (9691) fields are not supported in Europe. These fields are relevant for price sliding which is only supported in the US.

Symbology

Europe allows specifying ISIN, RIC, or Uniform symbologies which require the use of *SecurityExchange* (207), *Currency* (15), *SecurityID* (48), *IDSource* (22), and/or *Symbol* (55). US Equities uses *Symbol* (55) and *SymbolSfx* (65) only.

Market Hours

Pre- and post-market trading are supported in the US, but not in Europe.

Central Counterparty

The optional reply field *CentralCounterparty* (7772) is only supported in Europe.

Contra Broker

The *ContraBroker* (375) tag will contain completely different values in Europe than in the US.

11 Port Attributes

The table below lists port attributes that are configurable on the port or firm level. Changes to these attributes can be made by contacting the Cboe Trade Desk. Note that not all port attributes are applicable to all Cboe venues and services.

Attribute	Default	Description
All Midpoints Dark	No	Enable or disable forcing all incoming dark orders to be midpoint pegged.
Allow Dark/Lit Sweep Order Type^	No	Allow or disallow Dark/Lit sweep orders (9303 = u).
Allow Dark/Periodic Sweep Order Type^	No	Allow or disallow Dark/Periodic sweep orders (9303 = BH).
Allow Lit and Dark Sweep Orders^	No	Allow or disallow interbook sweep orders, designed to be a simple solution for multi-book access on the UK venue only between BXE and CXE.
Allow Test Symbols Only (Port setting)	No	Allow or disallow orders in non-test symbols at a port level.
Allow Test Symbols Only (Risk setting)	No	Allow or disallow orders in non-test symbols at a trading idenitifier level.
Allow Trade Capture	No	Allow or disallow Trade Capture Report submission.
Allowed Clearing MPID(s)^	All MPIDs	Trading Identifier(s) (a.k.a bank codes) allowed for order entry.
Allowed MPIDS(s) for Trade Reports^	All MPIDs	Trading Identifier(s) (a.k.a bank codes) allowed for Trade Report submission.
Bill To	Firm	Drives how your billing reports are constructed for each billing period.
Broker Preferencing	No	Allow or disallow eligible orders submitted to the PAB to be executed in priority against each other at the Execution Allocation phase.
Cancel on Disconnect	Option #1	Cboe offers two options for cancelling orders as a result of a session disconnect: 1) Cancel all open orders (continuous book and on-open, on-close and periodic auction orders). 2) Do not cancel any open orders.
Cancel on Reject	No	Enable or disable cancelling an order upon a cancel or modify reject for that order.
Cancel Open Orders on DROP Port Disconnect	Yes	Only applicable for sessions where Reject Orders on DROP Port Disconnect has been enabled. If all DROP ports associated with an order entry port become disconnected, cancel all open orders on the order entry port.
Capacity Strategy	Default (house)	When set, the capacity of individual orders received on the port will default to either Client or House as applicable.
CCP Style ExecID	No	Enable or disable converting the format of the ExecID (17) and TradeID (1003) returned by Cboe to match that sent to the CCPs.
Concatenate CompId and SubId	No	Requires all FIX traffic to contain concatenated (combined) Comp and SubIDs.
Default Dark on Expiry	No	Specifies a default value for Dark on Expiry behaviour if RoutingInst (9303) is not set.
Default Participant Trade Prevention	None	Specifies default values for PreventMemberMatch (7928) to prevent Participant Match.
Default Routing Instruction	B (Lit book)	Specifies a default value for RoutingInst (9303).

Duplicative Order Protection Action	Option #1	Action taken when Duplicative Order Protection criteria is met: 1. Not enabled 2. Reject new offending orders 3. Disable port for ClearingFirm. (Must call Cboe Trade Desk to re-enable.)
Duplicative Order Protection Order Count Threshold	None	Number of consecutive orders with the same Trading Identifier, Side, Price, OrdQty, and Symbol that must be seen to initiate Duplicative Order Protection Action.
Duplicative Order Protection Order Time Threshold	None	Defines the time window threshold for Duplicative Order Protection in seconds.
Enable Enhanced Settlement^	No	Allow or disallow non-standard currency and settlement location instructions to be placed on Trade Capture Reports.
Enable ETR Matching^	No	Allow or disallow ETR Matching, a type of Trade Capture Report.
Execution Collar (%)	20%	This controls the maximum percentage away from the Cboe reference price (see Cboe Participant Manual) that an execution is allowed to occur.
FIX Version	4.2	Defines the FIX Version the port will run (4.2 or 4.4). 4.4 is possible for an order handler port if dedicated for trade reporting only.
Forced Routing Instruction^	None	Used to override the routing instruction provided on incoming New Order Messages.
Indicate Self Matched Executions	No	Enable or disable self match values for TradeLiquidityIndicator (9730) on eligible trades.
Lock Auction Orders	No	Disallow order cancellation during periodic auction: No = Allow cancellations Yes = Do not allow cancellations
Max Open Trades	0 (Unlimited)	The maximum number of trades you can have open via one port at any one time.
Maximum Order Size	100,000	Maximum number of shares allowed per order.
Maximum Order Value	10,000,000 EUR	Maximum order EUR value per order.
Minimum Quantity Mechanism	Not set (defaults to MAQ Then MES)	Controls how an order's MinQty is handled: 'M' (MAQ Then MES) - MinQty orders can generate multiple fills with volumes below the minimum, so long as the total is at least as large as the MinQty. 'P' (Pure MES) - ALL fills on MinQty orders will be greater or equal to the MinQty. 'F' (MAQ On IOC Only) - Like PureMES for persistent orders but simulates Fill Or Kill (FoK) by making non-persistent orders like MAQThenMES. For all flavours, once resting ALL fills (including as a result of pegging) will be larger than the MinQty.
Periodic All or None	No	Determines whether Periodic All or None behaviour is enabled, if not explicitly set per order.
Periodic All or None Lit Sweep	No	Determines whether Periodic All or None for Lit Sweep behaviour is enabled, if not explicitly set per order.

Port Order Rate Threshold	5000 msgs/sec	The maximum allowed message rate on the session. When the first non-administrative message is received, a one second window begins. During the second no more than 4,999 additional non-administrative messages will be allowed within that window. If the rate is exceeded all new orders in the time window are rejected, modifies are treated as cancels, and cancels are processed.
Reject Orders on DROP Port Disconnect	No	Allows Member/Sponsoring Firms to associate a DROP port(s) to an order entry port(s). Once the association has been established, if no DROP ports associated with an order entry port are connected, reject orders on the order entry port until at least one of the DROP port session connections have been established.
Reject Orders on DROP Port Timeout (sec)	30	Only applicable for sessions where 'Reject Orders on DROP Port Disconnect' has been enabled. When the last associated DROP port for the order entry session has disconnected, the reject/cancel actions will be taken on the order entry session if an associated DROP port has not reestablished its connection in the defined time.
Restate on Reload^	No	Generates FIX Restatement on reserve order reload with new PITCH Order Id in Tag 198, and populates the new PITCH Order ID in Tag 198 of a Cancel/Replace acknowledgement that increases the size of a reserve order.
Send CCP	No	Return CentralCounterparty (7772) on Execution Reports.
Send Fee Code	No	Return FeeCode (9882) on Execution Reports.
Send Fix tag 1 (account)	No	Return Account (1) value on Drop Copy.
Send Fix tag 40 (order type)	No	Return OrdType (40) value on FIX Ack.
Send Fix tag 47 (capacity)	No	Return OrderCapacity (47) value on FIX Ack.
Send Fix tag 9303 (routing instruction)	No	Return RoutingInst (9303) on Execution Reports.
Send LastMkt	Yes	Mandates that the sub-MIC of the market you receive an execution from or the market you are submitting TCRs to, is sent on all order executions and trade capture report ack.
Send Peg Restatements^	Option #1	Send order restatements for Peg order moves: 1. No Peg restatements 2. Market Maker Peg orders only 3. All Peg orders except Market Maker Peg orders 4. All Peg orders
Send SubLiquidity	No	Enable or disable sending the 2nd character for TradeLiquidityIndicator (9730). Configurable for Drop Copy only.
Send Trade Breaks^	No	Enable or disable the Trade Break Message (35=UCC).
Suppress fee code	No	Enable or disable the suppression of fee code on execution reports.
Symbol Order Rate Threshold	5000 msgs/sec	Functions the same as the Port Order Rate Threshold but is calculated at the symbol level. It is capped by the Port Order Rate Threshold.
Test Symbol Order Rate Threshold	1	Maximum number of orders allowed to submit on a test symbol per second.
Trading Group	Blank	Free text field to describe the port.
Unique Wash Execution Ids	No	Appends a '.B' for a Buy or '.S' for a Sell to ExecID (17) on all trades.

Venue	Static Value	Ports within Cboe UK venue will have UK (United Kingdom) and Cboe NL will have NL (Netherlands).

†Port attribute can be overridden on an order-by-order basis.

^Requires certification.

12 Support

Please email questions or comments regarding this specification to `tradedeskeurope@cboe.com`.

Revision History

9 July 2008	Initial draft version.
16 July 2008	Europe URL and email address.
2 September 2008	Updated tick sizes. Removed references to *WorkingPrice* (9690). Removed wording that prices will be slid to a less aggressive amount if they do not fit the tick size for a symbol. Orders will be rejected if they do not fit the tick size. Added *DisplayIndicator* (9479) to "Cboe Specific Fields" and to allowed fields for "New Order – Single". Added section entitled "Hidden Orders".
8 September 2008	Added information about the *Text* (58) that will be received in the event of an order which, if executed, would happen outside the allowed price collar. Added clarification that *OrdRejReason* (103) will not always be sent on order rejects.
1 October 2008	Adjusted price collar for allowed executions to be up to 20% away from the market.
11 November 2008	Added ability to allocate trades to house or client account regardless of order capacity by using the *Account* (1) field. Updated front page text to note FSA authorisation.
16 December 2008	Added information on how notional value is determined for different order types.
3 February 2009	Added details for new Trade Cancel/Correct (UCC) message type.
26 March 2009	Discretionary orders are no longer permitted.
27 March 2009	Added new order type Post Only At Limit (*RoutingInst* (9303) = `Q`) and new tag *MaxRemovePct* (9618).
9 July 2009	Added Cboe Dark Pool routing. New values for *RoutingInst* (9303) and *TradeLiquidityIndicator* (9730).
15 July 2009	Added ability to post only to Cboe Dark Pool.
16 July 2009	Added Cboe Market On Close. New values for *ExecInst* (18), *OrdType* (40), *TimeInForce* (59), and *ExecType* (150).
23 July 2009	Added Participant Trade Prevention functionality. New tag of *PreventParticipantMatch* (7928). Added new section "FIX Differences Between US And Europe". Documented *OrdStatus* (39) = `A`. Only occurs on a Cancel/Replace reject if the FIX Handler is awaiting an acknowledgement on a new order from the Matching Engine.
30 July 2009	*RoutingInst* (9303) = `BA` or `PA` will now route any order which is not eligible for the Cboe Dark Pool to the integrated book.
24 August 2009	Updates to participant match prevention.
9 September 2009	Removed reference to *WorkingPrice* (9690) which isn't used on Cboe. Fixed wording about which symbology tags are required on cancel order messages.
7 October 2009	Central Counterparty updates.
19 October 2009	Clarified in the Execution Report that *TradeLiquidityIndicator* (9730) can also take on values `AD` and `RD` for executions which occur in the Cboe Dark Pool.
23 October 2009	Noted that *ExecInst* (18) will be returned on Execution Reports if set on the original order.

19 November 2009	Version 2.0. Added new “FIX Drop” section. Numerous formatting changes. *ExecType* (150) = 3 (Done for Day) was listed as a possible value Cboe would send, but it is not.
24 January 2010	Version 2.1. Removed *RoutingInst* (9303) = PA or PD. Added dark and onward routing. New values for *RoutingInst* (9303), *ExecInst* (18), *TradeLiquidityIndicator* (9730), *Text* (58) (reject message if routing is unavailable).
24 January 2010	Version 2.2. New values for *ContraBroker* (375).
18 February 2010	Version 2.3. Added Common Session Level Issues (§ 8, p. 65). Removed section about Market BBO and added Execution Collars (§ 1.9, p. 7) and Peg Order Pricing (1.11).
9 March 2010	Version 2.4. *RoutingInst* (9303) = RL or RC will re-route (RECYCLE) a booked order if another market locks or crosses the limit.
16 March 2010	Version 2.5. By default Cboe Trading At Last orders are now visible unless explicitly hidden using *DisplayIndicator* (9479) = I. *OrdType* (40) values 1 (Market) and 2 (Limit) are now accepted during the TAL phase.
15 April 2010	Version 2.6 Removed order quantity limit of 999,999. The new limit is 99,999,999 shares, but may be lowered if requested by a Participant or Sponsor.
30 April 2010	Version 2.7 Added Cboe Dark Self Cross (§ **??**, p. **??**). Added *CrossFlag* (77401). Added new value for *RoutingInst* (9303) (BX = Cboe Dark Self Cross). New values for *TradeLiquidityIndicator* (9730) (AM, RM). New values for *ContraBroker* (375).
7 May 2010	Version 2.8 Added *PreventParticipantMatch* (7928) = d.
20 May 2010	Clarified that Post Only orders will only reject when removing visible liquidity.
1 June 2010	Version 2.9 Added Cboe Plus (§ **??**, p. **??**) and Account Field (§ 1.15, p. 8) sections. Update (§ **??**, p. **??**) to clarify differences to Cboe Plus. New values for *RoutingInst* (9309) and clarification of *Account* (1)
26 August 2010	Version 2.10 Fixed minor typographical errors.
27 August 2010	Version 2.11 Parallel-D routing strategy. New values for *RoutingInst* (9303).
12 October 2010	Version 2.12 Restatement execution reports may be optionally received on reserve reload, allowing Participants to know the new OrderID that will be shown on the Cboe market data feeds. New initial character on *Text* (58) field of r indicating the restatement was sent due to reserve reload. See new section § 1.10, p. 7.
22 October 2010	Version 2.13 Midpoint peg orders may now have a limit price at one-half the tick size (§ 1.4, p. 6).
25 November 2010	Version 2.14 Parallel-2D routing strategy. New values for *RoutingInst* (9303). Added *Side* (54) to Trade Cancel/Correct (UCC) message type.

14 January 2011	Version 2.15 Noted that midpoint peg orders at half-tick sizes may only specify limit prices out to a maximum of four decimal places.
2 February 2011	Version 2.16 *OrdStatus* (39) = `C` was documented, but never sent in practice. Removed this value from the specification.
8 February 2011	Version 2.17 Noted that midpoint peg orders are not tick size validated.
22 March 2011	Version 2.18 Corrected various instances where MBBO was incorrectly referenced instead of PBBO.
5 April 2011	Version 2.19 Removed `1` and `2` from values which are communicated in *ExecTransType* (20) for execution reports. Removed *ExecRefID* (19) from execution reports as it is never sent. *AvgPx* (6) was missing from the execution report documentation.
14 April 2011	Version 2.20 Updated *RoutingInst* (9303) values of `R`, `RL`, and `RC` to note that Parallel-D is used, not CYCLE.
27 April 2011	Version 2.21 Added new value `AI` for *TradeLiquidityIndicator* (9730).
26 May 2011	Version 2.22 *Symbol* (55) may now be sent as the RIC or Ticker of the stock identified by *SecurityID* (48). Added new values `ECCP` and `NONE` for *FIXCentralCounterparty* (7772).
31 May 2011	Version 2.23 Added new value `RT` for *TradeLiquidityIndicator* (9730).
29 June 2011	Version 2.24 Noted that, if opted into at the firm or port level, *TransactTime* (60) will be sent with microsecond resolution.
30 June 2011	Version 2.25 Noted that *ClearingFirm* (439) is optional.
7 July 2011	Version 2.26 Noted that, if opted into at the firm or port level, *SendingTime* (52) and *OrigSendingTime* (122) will be sent with microsecond resolution.
25 Jul 2011	Version 2.27 Added optional 4^{th} character to *RoutingInst* (9303) used to select the resting book for routed orders. The Cboe Plus *RoutingInst* values of `PP` and `PL` have been deprecated in favour of `Y` and `L` which provide backwards compatibility with the US specification.
10 Oct 2011	Version 2.28 Removed all references to MOC/TAL.
27 Oct 2011	Version 2.29 Added reason code of 'm' to *Text* (58)
7 November 2011	Version 2.30 Clarified wording for *MinQty* (110) to note that it is usable with Cboe Dark Pool orders. Added Cboe Select.
13 December 2011	Version 2.31 Added reason code of 'T' to *Text* (58)
4 January 2012	Version 2.32 Added reason code of 'o' to *Text* (58). Noted that Cboe reserves the right to add new FIX tags and to update values distributed on Order by Order FIX Drop with no notice.

19 January 2012	Version 2.33 Updated to reflect the new BATS Chi-X Europe name. Clarified CHIX as a valid value for both *ContraBroker* (375) and *TargetCompID* (56) on CXE ports.
5 March 2012	Version 2.34 Pegged orders are now temporarily suspended rather than cancelled when the exchange is not in continous trading.
23 April 2012	Version 2.35 Clarity around session level reject messages (*MsgType* (35) = 3).
21 May 2012	Version 2.36 Renamed Cboe Select to Cboe DRTOnly and clarified use of the 2^{nd} and 3^{rd} characters of tag *RoutingInst* (9303).
28 May 2012	Version 2.37 Updated *PreventMemberMatch* (7928) to include a 3^{rd} character for Trading Group ID.
26 October 2012	Version 2.38 Added Lit Sweep and Dark Sweep interbook order types.
9 November 2012	Version 2.39 Clarified contents of tag *ContraBroker* (375).
21 November 2012	Version 2.40 Added Lit Sweep (Sequential) interbook order type.
7 December 2012	Version 2.41 Added SubLiquidityIndictor value K
19 December 2012	Version 2.42 Added TradeCaptureReport and associated message types. Removed Cboe Dark Self Cross order type.
15 February 2013	Version 2.43 Enhancements to Trade Capture Reports: *TrdType* (828) = 30 added, used to specify a 'special price' trade; *TrdType* (828) = 0 will now be used to specify regular (not 'special') priced trades, rather than 54; *TransactTime* (60) and *TradeDate* (75) are no longer ignored.
25 March 2013	Version 2.44 Enhancements to Trade Capture Reports: *PartyRole* (452) = 7 added, used to specify the party reporting the trade; *PartyRole* (452) = 17 added, used to specify the party the trade is alleged against; *PartyId* (448) Relaxed requirement for both parties to be identical
17 May 2013	Version 2.45 Clarification of optional availability of Trade Capture Reports over FIX DROP ports
23 May 2013	Version 2.46 Rewording of FIX Differences Between US and Europe § 10, p. 70, following Recognised Investment Exchange status, to remove ambigious usage of the terms "the MTF" and "the Exchange".
31 May 2013	Version 2.47 Removal of references to the Cycle routing strategy which has been retired (Parallel-D and Parallel-2D replace it).
17 June 2013	Version 2.48 Enhancements to Trade Capture Reports: Support for cancels and amends. Added message flow section
28 June 2013	Version 2.49 Removal of SEDOL support. Clarified usage of *MinQty* (110) for Cboe Dark Pool. Added Large In Scale reason code.

18 July 2013	Version 2.50 Added *CorrectedSize* (6655) for Trade Cancel/Correct messages. Made *OrderCapacity* (528) optional for Trade Capture Reports. Added various fields to the TradeCaptureReport that are available in the TRF to help normalise the two specifications. Adjusted behaviour of *TradeReportTransType* (487) and *TradeReportType* (856) in trade capture confirmations/declines. Added example message flow section for deferred publication trade reports.
20 August 2013	Version 2.51 Clarified behaviour of *TradeReportID* (571) for Participant submitted trade reports. Added *RptTime* (7570). Remove behaviour no longer valid now that we're past the ETR Cancel/Amend effective date
3 October 2013	Version 2.52 Remove 'Effective' notes now we're past their go live date
26 October 2013	Version 2.53 Clarification of optionally configured Trade Cancel/Correct messages over FIX DROP ports.
22 January 2014	Version 2.54 Removed *MTFAccessFee* (9621) from trade capture confirmations. Additional supported values for *TrdType* (828).
21 March 2014	Version 2.55 Addition of *ExecMethod* (2405) for trade captures.
28 March 2014	Version 2.56 Removed 'effective from' labels.
3 April 2014	Version 2.57 Allow the use of *TrdSubType* (829) and *SecondaryTrdType* (855) for trade reports. Clarification that *AccessFee* (9621) is indicative
20 May 2014	Version 2.58 Bug fix location of *SendingTime* (52) (should be session header). Clarified name of proposed MMT 3.7 field.
16 June 2014	Version 2.59 Increased the number of decimal places supported for Trade Capture Reports.
25 June 2014	Version 2.60 Added changes to Trade Capture messages effective with the Cboe Q3 2014 release, being those that follow. *VenueType* (1430), *MatchType* (574) and *PublicationMode* (1390) become mandatory. Deprecated use of *VenueType* (1430) = X in favour of *VenueType* (1430) = 0. Deprecated use of *PreviouslyReported* (570) in favour of values in *PublicationMode* (1390). Added support for *PublicationMode* (1390) = 0 (Do Not Publish). Deprecated use of *TrdType* (828) = 59 in favour of *TrdType* (828) = 62. Deprecated use of *TrdType* (828) = 60 in favour of *TrdType* (828) = 63. Deprecated use of *SecondaryTrdType* (855) = 58 in favour of *SecondaryTrdType* (855) = 64. Moved *TradingSessionSubID* (625) into the repeating group. Added support for *OrderCategory* (1115).
25 July 2014	Version 2.61 Removed 'effective from 25th of July' labels. Addition of Dark Lit orders.
25 July 2014	Version 2.62 Removed deprecated content.
8 October 2014	Version 2.63 Refer to the Participant Manual for execution collar (aka order price collar) for the actual percentages used.
10 October 2014	Version 2.64 Clarified ability to reuse *ClOrdId* with MODIFY ORDERs when daily limit trading risk controls are enabled.

29 October 2014	Version 2.65 Clarified usage of *OrderCategory*. Added AtTheOpen and AtTheClose to *TimeInForce* for auctions.
10 November 2014	Version 2.66 Added new reject reason code.
27 November 2014	Version 2.67 Clarified usage of *PartyRole* (452) = 7. Added new value C for *TradeLiquidityIndicator* (9730).
19 January 2015	Version 2.68 Clarified usage of *OrderCategory* (1115). Added missing reason code 'p'. Clarified usage of FIX 4.4 sessions. Added additional fields to Trade Capture Report Ack and Cboe to Participant Trade Capture Report. Clarified side normalisation in Cboe to Participant Trade Capture related messages. Added support for *GrossTradeAmt* (381). Changed *TradeHandlingInstr* (1123) in Cboe to Participant Trade Capture Report to echo back supplied value.
2 March 2015	Version 2.70 Cboe Clear NV clearing trade reports will now be truncated at the same level of precision as elsewhere. Add *TradeReportRefID* (572) to Trade Capture Report.
14 April 2015	Version 2.71 Clarification of *MinQty* (110) behaviour following "Minimum Execution Size" (MES) changes.
27 April 2015	Version 2.72 Remove 'Effective' notes now we're past their go live date
19 May 2015	Version 2.73 Added additional example messages for ETR Matching.
12 June 2015	Version 2.74 Added *PreventParticipantMatch* (7928) = P = Port Owner
13 June 2015	Version 2.75 Added support within Trade Capture Report messages for *SettlType* (63), *SettlDate* (64) and *SettlPrice* (730), as well as for DROP fed Execution Report and Trade Cancel Or Correct messages. Added support for periodic auctions. Added support for *LastMkt* (30).
14 July 2015	Version 2.76 Clarify that settlement prices cannot be amended.
1 December 2015	Version 2.77 Clarified that *TransactTime* (60) is only optional when *TradeReportTransType* (487) = 0.
8 January 2016	Version 2.78 Removed support for the Partial Post Only at Limit order type. Added support for a MiFID II Double Cap related reject reason code.
19 February 2016	Version 2.79 Updated for new branding.
30 March 2016	Version 2.80 Updated description for *TradeDate* (75).
29 April 2016	Version 2.81 Removed 'Effective' content related to the Q2 2016 release.

2 June 2016	Version 2.82 Formatting changes regarding repeating groups. Added detail about TRADE CAPTURE REPORT ACK messages being supported on ODROP. Added optional SideExecID.
27 July 2016	Version 2.83 Removed support for midpoint price, *ExecInst* (18) = `M`, on Trade Capture Reports. Clarified PreventParticipantMatch values.
18 August 2016	Version 2.84 Added support for new MiFID II Record Keeping fields on new orders. These fields are all considered optional at this stage, prior to the MiFID II regulations coming into effect.
26 August 2016	Version 2.85 Incorporated detail on upcoming Cboe BIDS fields in Drop.
8 September 2016	Version 2.86 Minor corrections to the FIX tag number associated with PartyRole, in 'Cboe BIDS Trade Confirmations'.
22 September 2016	Version 2.87 Redefinition of *PartyIDSource* (447), and addition of *Algo* (20001), MiFID II related fields on "New Order – Single".
28 September 2016	Version 2.88 Correction to expected values for *LiqProvOnly* (9215) and *Algo* (20001).
8 November 2016	Version 2.89 Add Order Record Keeping Fields to Execution Report Message from Cboe to Participant.
31 January 2017	Version 2.90 Change in spec to support MMT v3 and latest Order Record Keeping changes.
8 February 2017	Version 2.91 Review feedback for support of MMT v3 and Order Record Keeping.
2 March 2017	Version 2.92 Correction to FIX values for *TrdRegPublicationReasons* (8013) for TCRs to include LargeInScale.
20 March 2017	Version 2.93 Correction to FIX values for *TrdRegPublicationReasons* (8013) for Execution Report to include Negotiated waiver types. Correction to valid values range for Short Code. Clarified TrdType values in TradeCaptureReport. Confirmed values for *TrdRegPublicationReasons*
25 April 2017	Version 2.94 Update FIX values for *PartyRoleQualifier* (2376) to allow None when the corresponding PartyId is one of the reserved values.
3 May 2017	Version 2.95 Correction for *FeeCode* (9882) on Trade Capture Report Message from Cboe to Participant. Clarified valid values for *OrderOrigination* (1724).
24 May 2017	Version 2.96 Corrected description for *TradePriceCondition* (1839)=16 to drive MMT v3 'PRIC' on market data.
2 June 2017	Version 2.97 Corrected description for *TradePriceCondition* (1839). Driving MMT v3 'RPRI' on market data is not valid on-exchange.
20 July 2017	Version 2.98 Corrected description for *TrdType* (828). Trade reports on dark trades are not valid on-exchange. MMT v3.04 support for Q4 2017 release.

20 September 2017	Version 2.99 Removed support for ILQD and SIZE deferrals which are only applicable to RTS 2 instruments from *TrdRegPublicationReasons* (8013). *TrdRegPublicationReasons* (8013) was added to inbound Trade Capture Reports when the Q4 2017 release was announced; no longer necessary due to removing support for SIZE.
24 November 2017	Version 3.0 Updated definition of reserved value '3' in *PartyID* (448) from 'CLIENT' to 'NORE'
27 November 2017	Version 3.1 Deprecated use of *TradePublishIndicator* (1390)=0 (Do Not Publish).
26 March 2018	Version 3.2 Update text for *MinQty* (110).
02 May 2018	Version 3.3 Added support for *SettlCurrency* (120) and *RootPartyRole* (1119) value 10 (Settlement Location) Clarified hidden orders notional value calculation for midpoint peg orders.
06 August 2018	Version 3.4 Clarified definitions of *LastPx* (31) and *LastShares* (32). These should not be zero.
13 August 2018	Version 3.5 Added pending periodic auction (k) reject reason.
28 September 2018	Version 3.6 Cboe will enforce port level day-uniqueness for *TradeReportID* (571).
15 November 2018	Version 3.7 Update *TradeReportID* (571) description.
14 January 2019	Version 3.8 Added Purge Port messages. Removed references to external routing.
28 February 2019	Version 3.9 Clarified behaviour of orders and TCRs with an incorrect Cboe RIC for the venue being submitted to.
31 May 2019	Version 3.10 Add DXE environment. Add Cboe Closing Cross.
11 July 2019	Version 3.11 Corrected typo for *CustomGroupID* (7699).
11 September 2019	Version 3.12 Updated DXE RIC suffix to be consistent with value published on Cboe Europe Brexit FAQ page.
14 October 2019	Version 3.13 Clarified usage of field TradeHandlingInstr in conjunction with SettlType.
17 April 2020	Version 3.14 Added A-LPS BBO liquidity (AS and RS) to *TradeLiquidityIndicator* (9730) .
28 April 2020	Version 3.15 Added dark periodic sweep (BH) to *RoutingInst* (9303) .
30 September 2020	Version 3.16 Updated FIX Differences Between US and Europe.
30 December 2020	Version 3.17 Added clarifications on construction of TVTIC.

04 February 2020	Version 3.18 Further clarification on how *TradeID* should be used for TVTIC.
14 January 2022	Version 3.19 Added Port Attributes section.
21 March 2022	Version 3.20 Added that *TimeInForce*=IOC is treated as an Accept-Or-Cancel for Periodic Auction orders. Added optional 3rd char to *RoutingInst* for Periodic Auction price improvement.
9 May 2022	Version 3.21 Updated description for *Tolerance* (9128)
17 May 2022	Version 3.22 Now includes a greatly expanded list of the available port attributes.
01 June 2022	Version 3.23 Clarify that rejects recevied over a BOE Port will not be available over Order by Order Drop.
12 September 2022	Version 3.24 Updated description for *ExecInst* (18) and *PegDifference* (211) for periodic auction orders. Added description for *ExtExecInst* (9416) for periodic auction orders. Added optional 4th char to *RoutingInst* for Periodic Auction Dark-on-Expiry.
15 February 2023	Version 3.25 Removed deprecated *TradePriceCondition* values.
17 May 2023	Version 3.26 Added Default port attributes for Dark on Expiry, Periodic All or None, and Periodic All or None for Lit Sweep.
05 June 2023	Version 3.27 Updated description for *ExecInst* (18).
14 June 2023	Version 3.28 New routing instruction for periodic lit sweeps.
30 June 2023	Version 3.29 Updated with support for ESMA RTS 1 and RTS 2 changes.
19 October 2023	Version 3.30 Updated with support for MMT v4.1 resulting from FCA's PS 23/4.
16 November 2023	Version 3.31 Updated with RTS 2 non-equities support on *TrdType* (828).
15 January 2024	Version 3.32 Deprecating support for Agency Cross Trade Indicator on BXE and CXE on *TrdSubType* (829).
09 February 2024	Version 3.33 Add ExecutionReport Trade Reg Publication Reasons *TrdRegPublicationReasons* (8013). Fix valid values for ExecutionReport publication reasons.