CBOE FIX Protocol Support

Version 9.0.2

Volume 3B: Application Layer: Fundamentals and Field (Tag) Dictionary

Programmer's guide to the CBOE FIX 4.2 Service Application Layer, including product identification, product information, market status, product status, email notification, and the Field (Tag) Dictionary

CONFIDENTIAL
CBOE Proprietary Information

15 July 2011

Document #[FIX-03B]

Production Release

Front Matter

Disclaimer

Change Notices

The following change notices are provided to assist users of the CBOE FIX Services in determining the impact of changes to their applications.

Date	Version	Description of Change
15 Jul 2011	9.0.2	Added the new UDF, CancelOpenQty[9310]
29 Apr 2011	9.0.1	Added new field, ExchMkt[30]
14 Jan 2011	9.0	Added the new UDFs below.
		LightOrderIndicator[9317]
		TradingSessionID[20104]
		OrderTypeLogonIndicator[9318]
		PendingFillQty[20102]
		PendingCxlQty[20103]
17 Aug 2010	7.0	Added new C2 value, C2OX, to SecurityExchange[207]
13 Apr 2010	7.0	Added the new UDF, TradingGroupInfo[20010] to the Security Definition message.
08 Jan 2010	7.0	Added the new UDF, ShortSaleIndicator[20101]
14 Aug 2009	6.1	No changes
22 May 2009	6.0	Added the new UDF, DirectedFirm[5941]
		Modified tags 55, 167, 200, 205, 201 and 202 for the option symbology initiative.
25 Nov 2008	5.3	Added the new UDFs BrokerRoutingID [tag 6818] and ConcurrentOrder/QuoteIndicator [tag 9192]
03 Oct 2008	5.2	Added the new UDF AuxAuctionInfo [tag 9221]
23 July 2008	5.1	No changes
29 Feb 2008	5.0	Updated the definition of RatioQty [tag 319]
		Added definitions for the new tags representing Paired Strategy Order Entry Using The One Step Method
18 Jan 2008	4.2.4	Included two TradeLiquidityIndicator[9730] types to the tag dictionary
		Added two new sub-section under the section: Scenarios for Business

Date	Version	Description of Change
		Message Reject
02 Nov 2007	4.2.3	Removed the CBSX reference from UDF, TradeLiquidityIndicator[9730]
		Modified ISO values for TimeInForce[59]
01 June 2007	4.2.2	Added new UDF, TradeLiquidityIndicator [9730]
23 Feb 2007	4.2.1	No changes
27 Dec 2006	4.2	Updated Rule80A (aka OrderCapacity) [tag47] in the Tag Dictionary
15 Dec 2006	4.2	Modified values for TimeInForce[tag 59] for ISO in the Tag Dictionary
20 Sept 2006	4.1	Modified values for OrdType[tag 40] for Stock contingency types in the Tag Dictionary
08 Sept 2006	4.1	Updated OrdType[tag 40] to include the Stock contingency types in the Tag Dictionary
25 May 2006	4.0	No changes.
06 Jan 2006	3.2b	Updated the values for tag 326
		Changed the description of tag 382
		Added UDF SuppressOrderStatus[9191] to the tag dictionary
		Added Table 28 Security Definition message used to send Security Status messages in the compact mass message format
12 Aug 2005	3.2	No updates
29 July 2005	3.2	Added UDF tags 9380 and 9381
08 Apr 2005	3.1a	Documentation Errata Release
30 Nov 2004	3.1	Internalization and automated auction
20 July 2004	3.0a	Documentation Update
18 June 2004	3.0	API Enhancements
28 April 2004	2.52	Version Changes
06 February 2004	2.63	Support added for the CBOE Futures Exchange (CFE) and Stock Trading On CBOEdirect (STOC)
08 July 2003	2.51	Documentation Errata Release
23 Apr 2003	2.5	Support for Hybrid Trading
24 Jan 2003	2.1	Additional Options Linkage information added to document.
07 Oct 2002	2.01	Errata release for CBOEdirect® 2.0 that includes specifications for order routing through the Options Linkage Authority.
22 Apr 2002	2.0	Production Release
27 Feb 2002	2.0b	Re-titled document to “Application Layer: Fundamentals”
		Moved up material from market making volume that applies to all users.
01 Feb 2002	2.0	CBOEdirect 2.0 Beta 1 release
		-Incorporated Quote Risk Monitoring from fix-03c into this document; Support for strategy and futures products is provided; general cleanup and

Date	Version	Description of Change
		corrections.

Support and Questions Regarding This Document

Questions regarding this document can be directed to The Chicago Board Options Exchange at 312.786.7300 or via e-mail: api@cboe.com.

The latest version of this document can be found at the CBOE web site http://systems.cboe.com.

Table of Contents

Tables & Figures

List of Tables

About This Document

Purpose

This document is intended to provide information and guidance on how to connect to the CBOE FIX 4.2 service to access CBOE markets.

Intended Audience

Management requiring a deeper technical understanding of CBOE's support for FIX 4.2 in making decisions on how best to participate in CBOE markets and developers of applications that will use the FIX 4.2 service to communicate with CBOE markets.

Related Documents

Document Number	Document Description
FIX-RELNOTES	CBOE FIX Release Notes Version 2.0
FIX-ROADMAP	CBOE FIX Document Road Map
FIX-01	CBOE FIX Volume 1: Overview & Concepts
FIX-03A	CBOE FIX Volume 3A: FIX 4.2 Programmer's Guide: FIX Session Layer
FIX-03B	CBOE FIX Volume 3B: FIX 4.2 Programmer' Guide: Application Layer: Fundamentals and Field (Tag) Dictionary
FIX-03C	CBOE FIX Volume 3C: FIX 4.2 Programmer' Guide: Order Routing
FIX-03D	CBOE FIX Volume 3D: FIX 4.2 Programmer' Guide: Market making
FIX-06	CBOE FIX Volume 6: FIX 4.2 Certification Guide
FIX-07	CBOE FIX Volume 7: FIX 4.2 CBOE Market Data FIX Engine (CFIX)
NET-01	CBOE Network Connectivity Guide
	Financial Information Exchange Protocol (FIX) Version 4.2 (http://www.fixprotocol.org)
	Financial Information Exchange Protocol (FIX) Version 4.3 (http://www.fixprotocol.org)

Usage and Conventions

The FIX 4.2 Specification contains definitions for all standard FIX messages and tags. With the exception of the Tag Dictionary, the standard definitions for FIX messages have been omitted from this document. This was done for brevity and to not obscure the text describing CBOE's particular implementation of a message or a tag. In the Tag Dictionary, the standard definition is provided next to CBOE's usage.

Fields that follow the FIX standard and are not subject to any CBOE specific constraints are denoted with the phrase "**Per standard.**"

Fragments of FIX messages are shown in the courier new font. The ^ is used to represent the FIX field separator (ASCII 01).

```
55=IBM^48=1237^167=OPT^200=200010^201=0^202=105.00^207=W^
```

FIX Tags are shown are presented by name in italics followed by the tag number in brackets [].

SecurityType[167]

Symbol[55]

Firms should put the repeating group tags in exactly the same order as they appear in the FIX Specification. This will become a requirement for FIX 4.3.

FIX Application Layer

Common functionality that is useful to both order routing and market making applications is presented in this volume. All FIX users can benefit from access to market status that is available by using the Trading Session Status Request to subscribe for trading session status events and the Security Status Request to subscribe for product status events.

The CBOE FIX 4.2 Service provides all users with the ability to retrieve product class and product information using the Security Definition Request message. The Security Definition Request is also the message that is used to define strategies within CBOEdirect.

All users are required to be able to receive and respond to Email (text) messages generated by the CBOE help desk and trading operations staff.

As with most applications, the CBOE FIX 4.2 Service only uses a subset of the optional fields available within the FIX specification. CBOE does not persist, nor does it return fields that are not explicitly documented as being supported.

Any additional, optional FIX fields supplied on any firm request messages will be ignored by the CBOE FIX 4.2 service. These unsupported fields will not be carried through the system. This means that the unsupported fields will not be sent into CBOEdirect, nor will any unsupported fields be reported back to the firm on Execution Reports.

Request Processing Flow

Valid FIX request messages are sent to CBOEdirect. The FIX 4.2 service receives an acknowledgement when CBOEdirect has accepted the request.

A FIX Request is not officially received by CBOE until the requesting firm receives an Execution Report indicating that a New Order or Pending Cancel/Replace request has been received. This applies to all trading sessions, electronic or open outcry.

Guaranteed Delivery of Messages To Firm

Certain messages that impact the ability for a firm to track their current position in the market are implemented using guaranteed message delivery. In the event that your connection is lost to the CBOE FIX 4.2 Service, the following message types will be queued and delivered to you once the connection has been reestablished. All other message types transmitted after a connection loss will not be sent to the user.

Table 1 FIX Messages for which delivery is guaranteed

FIX Message	Events for which message is generated
Execution Report	New Order, Order Cancel, Order Fill, Trade Bust, Trade Bust with Reinstate, Nothing dones, Quote fills, Quote Busts
Security Status	Changes in state for products for which the user has subscribed
Email	Email from CBOE help desk to FIX user

Possible Resend in the event of failure in delivery

In the event that there is a failure before the delivery of a message is confirmed – CBOEdirect will resend the guaranteed message. The *PossResend[97]* tag will be set to "Y" for all possible resend messages. Please note that a *PossResend[97]*

=”Y” indicates that you may be receiving a duplicate message under a different FIX sequence number. This differs from the *PossDupFlag[43]* which indicates a resend of a message with the same sequence number. The *PossResend[97]* ="Y” is used at the Application Layer to indicate resending a message. The *PossDupFlag[43]* is controlled at the FIX Session Level and is used to indicate a response to a Resend Request Message.

Custom Defined Tags (User or CBOE Defined Fields)

FIX firms should never reject messages sent from the CBOE to the firm. If the firm receives messages from CBOE containing tags that the firm does not recognize (CBOE User Defined Tags or other), the firm should ignore the tag. The firm should not try to validate the unknown tag and reject CBOE's message. CBOE reserves the right to force logout any user that rejects a message sent from the CBOE to the firm.

Application Queues

The following custom fields have been added to outbound messages from CBOE where application level queues can occur. The information is provided to assist both CBOE and the firm in identifying performance issues. Most often, backlogs occur because the remote firm FIX engine is unable to read off the messages in a timely manner due to contention or bandwidth issues. For messages reporting on orders and quotes, CBOE will deregister the user’s subscription. For Market Data Messages, the latest information will be provided.

Under exceptional circumstances (a firm queuing heavily on order and quote statuses or an internal system problem) the CBOEdirect system will take the following actions

- For backlogs in receiving Order Status, Quote Status, and Execution Reports, the CBOE will automatically deregister the user’s subscription to Order Status and Execution Reports. The user would have to re-register for Order Status and Quote Status.
- For backlogs in receipt of market data messages, the firm will be unsubscribed to the market data by the CBOEdirect system. The user will be notified via a Market Data Reject message that the subscription removal has occurred.
- For other subscriptions (product statuses, trading session status, etc.) the user will be unsubscribed from that service and a business message reject reason will be sent with a reason text.

Table 2 From Firm To CBOE

Tag#	User Defined Fieldname	Description
6700	ApplicationQueueAction RequestResolution	Optional customer header user defined field that indicates the action that should be taken to resolve the Application queue depth (backlog). 0- No action taken 1- Queue flushed 2- Overlay last 3- End session subscription This tag will be applicable ONLY on Market Data subscription request messages. It will be rejected on all the other messages.

NO_ACTION

CBOE currently does not support this designation. It will return the same resulting actions as DISCONNECT_CONSUMER.

DISCONNECT_CONSUMER:

The way this works is that if the firm's queue to its subscription exceeds a particular number (configured by CBOE), the subscription will automatically be unregistered and the firm will received a "CallbackDeregistration" message from CBOE telling you that CBOE has removed the subscription because the queue was too large (see document FIX-07). This value is currently configured to a queue size of 5,000, but is subject to change.

FLUSH_QUEUE

This option allows the end user to continue as normal despite building up large queues. Once the queue reaches a certain size (currently configured to 1,000), we will automatically flush (throw out) the queue for the firm, and then allow it to build again.

OVERLAY_LAST

For a given subscription, CBOE only keeps and delivers the most recent message for a given product. For example, if 10 updates for IBM JAN 130 PUT arrive in the queue before CBOE has finished sending the last update, then the next transmission of IBM JAN 130 PUT to the firm will only include the most recent update. This allows the firm's subscription to always receive the most current message that CBOE has despite the firm not being able to keep up. CBOE will throw out any stale message when this happens. If a firm wishes to see all messages, even stale ones that are sitting in queue, the firm should not use OVERLAY_LAST.

Table 3 From CBOE To Firm

Tag#	User Defined Fieldname	Description
6699	ApplicationQueueDepth	A User defined field that provides the number of application level events that are queued for processing behind this current message. For instance, when ApplicationQueueDepth > 0 on a Execution Report - corresponding application response message sent from CBOE to the firm, this indicates that there are still ApplicationQueueDepth # of reports that have yet to be generated and transmitted to the user. This information is provided to help counter parties manage throughput and backlog issues. This tag is available for Order Status, Quote Status, Order Execution Reports, Quote Execution Reports, and all related Market Data messages.
6701	ApplicationQueueActionTaken	Optional application message field that indicates the action taken to resolve the Application queue depth (backlog). 0- No action taken 1- Queue flushed 2- Overlaid last 3- Subscription Ended This tag will be applicable ONLY on Market Data response messages.

Application Level Error Handling

Business Message Reject

The Business Message Reject was introduced in FIX 4.2 to provide an Application Level message to reject FIX application messages that do not have a specific response message. The Business Message Reject message is used to reject an otherwise valid FIX application messages for business reasons or invalid data when there is no other means of rejecting the field.

The Business Message Reject message is not used to reject malformed or invalid requests – this is done by the Reject Message.

Table 4 CBOE FIX 4.2 Standardized Error Handling

Session-level problem meeting the criteria of the session-level Reject message	Use the session-level Reject message (MsgType=3)
In response to • New Order – Single • New Order- List • Order Status Request	Use the Execution Report message
In response to: • Order Cancel Request • Order Cancel/Replace Request	Use the Order Cancel Reject message
In response to: • Quote Request • Quote • Mass Quote • Quote Cancel • Quote Status Request	Use the Quote Acknowledgment message
In response to: • Market Data Request	Use the Market Data Request Reject message
In response to: • Security Definition Request	Use the Security Definition message
In response to: • Security Status Request	Use the Security Status message
In response to: • Trading Session Status Request	Use the Trading Session Status message

Note the only exception to this rule is in the event a business message is received, fulfills session-level rules. However, the message cannot be communicated to the business-level processing system. In this situation a Business Message Reject with BusinessRejectReason = "Application not available at this time" can be issued if the system is unable to send the specific "reject" message listed above due to this condition.

Messages which can be referenced via the Business Message Reject message are:

(the "ID" field BusinessRejectRefID refers to noted in [ ])

- Email [EmailThreadID]
- Order Cancel Reject [ClOrdID]
- Market Data-Snapshot/Full Refresh [MDReqID]
- Market Data-Incremental Refresh [MDReqID]
- Market Data Request Reject [MDReqID]
- Quote Acknowledgment [QuoteID]
- Security Definition [SecurityResponseID]

- Security Status [SecurityStatusReqID]
- Trading Session Status [TradSesReqID]
- Order Single [ClOrdID]
- Order Cancel Replace [ClOrdID]
- Order Cancel Request [ClOrdID]
- Mass Quote [QuoteID]
- Quote [QuoteID

Table 5 Business Message Reject

Tag	Field Name	FIX Req'd	Comments
	Standard Header	Y	*MsgType[35]* = j (lowercase)
45	RefSeqNum	N	MsgSeqNum of rejected message
372	RefMsgType	Y	The MsgType of the FIX message being referenced.
379	BusinessRejectRefID	N	The value of the business-level "ID" field on the message being referenced. Required unless the corresponding ID field (see list above) was not specified.
380	BusinessRejectReason	Y	Code to identify reason for a Business Message Reject message.
58	Text	N	Where possible, message to explain reason for rejection
	Standard Trailer	Y	

Scenarios for Business Message Reject:

BusinessRejectReason[380]	
Enumeration	**BusinessRejectReason Description**
0	Other
1	Unknown ID
2	Unknown Security
3	Unsupported Message Type (receive a valid, but unsupported MsgType)
4	Application not available
5	Conditionally Required Field Missing

Business Reject Messages Requiring Special Attention

An addition to CBOE inbound order and quote error handling is a quality of service feature (remote transaction timeout (RTT) exceptions), wherein the processes will set a maximum service time for an inbound order or quote to process internally within the CBOEDirect system. The intention of this addition is to release control of the inbound processing back to the end user with an indication an error or slow down has occurred without waiting for the situation to clear. While the request will abort with an exception it does not mean the original request will not eventually complete. This is in essence a "maybe" condition and it will require the user to take action to with regard to the inbound quote or order stream at the time of the error.

Regardless of the values of RefMsgType [372], BusinessRejectRefId [379], BusinessRejectReason [380]. If CBOE sends out a Business Reject Message with the following Text [58] value

"Ambiguous Message Dispatch Failure (SE). Contact CBOE helpdesk for confirmation of action.",

then ***it is very important that this message be considered carefully***. The Electronic Trading System to which CBOE FIX 4.2 belongs is a distributed system. That particular text is only produced when there is an internal communications failure within the CBOE system. That means that the message referred to by RefMsgType [372], and BusinessRejectRefID [379] did not complete its normal processing task or did not complete within a reasonable time frame. If it is a message that affects the Market Position of the SenderCompID [49], then the SenderCompID [49] needs to:

1. Take immediate action to compensate for the ambiguous message failure (e.g., if some form of a Quote is involved, cancel or log out to cancel any resting Quotes).
2. If an order origination request failed a cancel request should be sent to ensure the order is not in the market before taking an alternate action regarding the order.

Contact the CBOE helpdesk to help determine where and what the internal communications involved failed or succeeded.

Other Messages Requiring Special Attention

Besides the Business Reject Message [35=j], it is possible that other messages could also be sent with the following Text [58] value

"Ambiguous Message Dispatch Failure (SE). Contact CBOE helpdesk for confirmation of action."

It is very important that these messages be considered carefully. The same factors used to determine the correct response to the Business Reject Messages above should be used to respond to these messages. The Messages involved will be one of the following:

- Quote Acknowledgement Message [35=b] in response to either a Mass Quote Message [35=i] or a Quote Message [35=S]
- Execution Report [35=8] reporting a reject of one or more parts of a Cross Order or Auction Pair.

Again, the fact that these messages report an action (reject) regarding a Quote or Order Action does not override the important word **`"Ambiguous"`** in the Text.

Email

The email message is used to exchange messages between the exchange and the firm.

A firm must be able to send and receive email (text) messages to CBOEdirect to be a certified user of CBOEdirect.

Table 6 Email

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
	Standard Header	Y	Y	*MsgType[35]* = C
164	EmailThreadID	Y	Y	Unique identifier for the email message thread
94	EmailType	Y	Y	
42	OrigTime	N	N	
147	Subject	Y	Y	Specifies the Subject text
33	LinesOfText	Y	Y	Specifies the number of repeating lines of text specified
58	Text	Y	Y	Repeating field, number of instances defined in LinesOfText which contains the body of the email message.
	Standard Trailer	Y	Y	

Identifying products using the FIX instrument block

CBOE uses a subset of the FIX standard security identification tags to identify securities in CBOE markets.

CBOE permits identification of the following types of products via the FIX Standard Security identification.

- Option Classes
- Option Series
- Underlying Products, such as equities or indexes.
- Strategies (multileg products, such as option strategies, calendar spreads, etc.)
- Single stock futures (OneChicago electronic market)
- CBOE Futures Exchange (CFE) futures
- CBOE Equity-based products (QQQ, SPDR, OEF, etc. – e.g. these are the underlying QQQs, not the options overlying the QQQs)

All products (option series, futures, underlyings, strategies) are treated as products (securities) within the CBOEdirect system. Each product is given a unique numeric identifier referred to as the CBOEdirect ProductKey. This ProductKey is used as the FIX *SecurityID[48]* to identify a product.

Security Types used by CBOEdirect markets

The following security types are used in CBOEdirect. CBOE has defined two additional non-standard *SecurityType[167]* enumerations for use with CBOEdirect to identify underlying indexes and multileg instruments.

Table 7 CBOE Defined Security Types for Indexes and Strategies

Constant	Description	Standard Status
OPT	Equity or index options.	
CS	Type used to identify all stock products within CBOEdirect The CBOEdirect system does not differentiate the type of stock (equity). For this reason, all stocks that serve as underlyings with CBOEdirect will be designated as *Fix SecurityType[167]* = "CS"	
FUT	Futures, including single stock (securities) futures	
INDX	Underlying Index for an Index Option product, such as OEX, SPX, NDX, VIX	Nonstandard
MLEG	Multi-leg security, such as an option strategy or a combination of underlying and derivative (buy write for example)[1].	Nonstandard
USTB	The CBOEdirect system does not differentiate the type of debt For this reason, all debt products that serve as underlyings with CBOEdirect will be designated as *Fix SecurityType[167]* = "USTB" US Treasury underlying the Interest Rate products	

[1] Adopted as part of FIX 4.3

Unused Optional Security Identification Fields

The following fields from the FIX security identification tags are not used with CBOEdirect.

Table 8 Unused and unsupported FIX security tags

Tag	Field
65	SymbolSfx
106	Issuer
206	OptAttribute
223	CouponRate
231	ContractMultiplier
348	EncodedIssuerLen
349	EncodedIssuer
350	EncodedSecurityDescLen
351	EncodedSecurityDesc

Product Identification on Inbound Requests to CBOEdirect

The FIX instrument block usage is dependent upon the type of product (*SecurityType[167]*). CBOEdirect permits the specification of a product using industry standard symbology and by the CBOEdirect Product Key (*SecurityID[48]*).

Option Product Class identification

All CBOEdirect products are members of a product class.

Some inbound requests, such as security definition request and market data requests permit you to specify the option class – not the individual series.

Option classes are specified using only the Ticker Symbol. Because the symbol for the option class can be the same as the underlying security – you are required to explicitly specify the *SecurityType[167]* of "OPT".

Table 9 Identifying Option Classes

Required

Tag	Field	Usage
55	Symbol	Standard Option Symbol (Examples: MSQ, DLQ, IBM)
167	SecurityType	"**OPT**"
336	TradingSessionID	Although not part of the FIX Instrument block – the TradingSessionID is required by CBOEdirect. Examples: W_MAIN (CBOE regular trading hours) ONE_MAIN (OneChicago main trading session) CFE_MAIN (CBOE Futures Exchange) COF_MAIN (CFE Options on Futures) W_STOCK (Stock Trading On CBOEdirect – formerly SEMS products)

Optional

Tag	Field	Usage
		There are no optional tags at this time

Not used in this context:
MaturityMonthYear[200], PutOrCall[201], StrikePrice[202]. MaturityDay[205], SecurityID[48], IDSource[22]

Example identifying option classes.

Represents Microsoft Equity Option Class

```
55=MSQ^167=OPT^336=W_MAIN
```

Example identifying an index option class.

Represents OEX Option Class trading in the regular trading hours (open outcry) session

```
55=OEX^167=OPT^336=W_MAIN
```

Option series identification using industry standard tags

There are two ways to identify a specific option series. The first method uses industry standard ticker symbol and option characteristics, such as "IBM 200010 105.00 Put". The second method uses the CBOEdirect ProductKey that identifies each product.

Note that both identification techniques can be used within the same session – either approach is acceptable at any time.

Table 10 Identifying option series using industry standard identifiers

Required

Tag	Field	Usage
55	Symbol	Standard Options Symbol (Trading Symbol) (Examples: OEX, OEY, OEW; RUT, RUW, RUZ; MSQ, MQF)
167	SecurityType	"**OPT**"
200	MaturityMonthYear	Expiration Year and Month (Proper format: 200204)
205	MaturityDay	Required for "by Name" product look up. Specifies the actual expiration day. For example, specifying "1" will mean the option expires on the first day of the month.
201	PutOrCall	0 = Put, 1 = Call
202	StrikePrice	0-99999999.9999 (Decimal places may vary but may not be greater than four.
336	TradingSessionID	Although not part of the FIX Instrument block – the TradingSessionID is required by CBOEdirect (Examples: W_MAIN, ONE_MAIN).

Optional

Tag	Field	Usage
		There are no optional tags at this time

Not used in this context: *SecurityID[48], IDSource[22]*

Example specifying an option series using option symbol and contract terms

Represents the Russell 2000 index April 2002 520.00 Call. Note that you must use the specific symbol for the option series.

```
55=RUW^167=OPT^200=200204^201=1^202=520.00^336=W_MAIN
```

Option series identification using the CBOE Product Key

Alternative identification using the CBOEdirect product key as the *SecurityID[48]*. CBOEdirect has a unique numeric identifier for every product traded at the exchange. A product can be an option series, a futures contract, an underlying.

Table 11 Identifying an option series using the CBOEdirect Product Key

Required		
Tag	**Field**	**Usage**
55	Symbol	Standard Options Symbol
48	SecurityID	CBOE supplied and maintain unique product key – format is an integer.
336	TradingSessionID	Although not part of the FIX Instrument block – the TradingSessionID is required by CBOEdirect. Possible value: W_MAIN
Optional		
Tag	**Field**	**Usage**
167	SecurityType	"**OPT**" – This is not required, as the SecurityID will identify the product uniquely.
22	IDSource	If used must be set to "8" (Exchange)
Not used in this context: *MaturityMonthYear[200], PutOrCall[201], StrikePrice[202],. MaturityDay[205]*		

Example specifying an option series using the CBOEdirect ProductKey

Assuming the CBOE Product Key for the October 2000 105.0 Call is 1237 – the following FIX message will identify the product.

Represents the IBM October 2000 105.00 Call.

```
55=IBM^48=1237^336=W_MAIN
```

Security futures product identification using industry standard tags

Table 12 Futures product identification using industry standard identifiers

Required

Tag	Field	Usage
55	Symbol	Security Futures Contract Symbol. (Examples: MSFT1C, IBM1C)
167	SecurityType	"FUT"
200	MaturityMonthYear	Expiration Year and Month. Format: YYYYMM (e.g., 200209)
336	TradingSessionID	Although not part of the FIX Instrument block – the TradingSessionID is required by CBOEdirect The value for OneChicago security futures markets: **ONE_MAIN**.

Optional

Tag	Field	Usage
205	MaturityDay	Required for "by Name" option product look up. Specifies the actual expiration day. For example, specifying "1" will mean the option expires on the first day of the month.

Not used in this context: *PutOrCall[201], StrikePrice[202], SecurityID[48], IDSource[22]*

Example specifying an futures contract using the futures symbol and contract terms

Represents the IBM October 2002 futures contract

```
55=IBM1C^167=FUT^200=200210
```

Security futures product identification using the CBOE Product Key

Alternative identification using the CBOEdirect product key as the *SecurityID[48]*. CBOEdirect has a unique numeric identifier for every product. A product can be an option series, a futures contract, an underlying.

Table 13 Futures product identification using the CBOEdirect Product Key

Required		
Tag	**Field**	**Usage**
55	Symbol	Security Futures Contract Symbol. (Examples: MSFT1C, IBM1C)
48	SecurityID	CBOE supplied and maintained unique product key – format is an integer.
336	TradingSessionID	Although not part of the FIX Instrument block – the TradingSessionID is required by CBOEdirect.
Optional		
Tag	**Field**	**Usage**
167	SecurityType	"**FUT**"
22	IDSource	If used must be set to "8" (Exchange)
Not used in this context: *MaturityMonthYear[200], PutOrCall[201], StrikePrice[202]*. *MaturityDay[205]*		

Example specifying an futures contract using the CBOEdirect ProductKey

Assuming for this example only the CBOEdirect Product Key for the IBM October 2002 Security Futures Contract is 3213[2] – the following FIX message will identify the product.

Represents the IBM October 2002 Security Futures Contract

```
55=IBM1C^48=3213
```

[2] The value 3213 is given only for example - the actual product key for this particular futures contract will be different.

Strategy Identification

To specify an already defined strategy you must specify the CBOEdirect product key as the *SecurityID[48]* and set the SecurityType to "MLEG".

If you do not have the CBOEdirect product key available, you must use a Security Definition Request to define the strategy. If the strategy already exists, you will receive the product key for the existing product. If a strategy does not exist, one will be defined for you and the product key will be returned.

Table 14 Strategy identification uses the CBOEdirect ProductKey only

Required		
Tag	**Field**	**Usage**
55	Symbol	Standard Product Class Symbol (Examples: IBM, MSQ (for options), MSFT1C (for futures) etc.)
48	SecurityID	CBOE supplied and maintain unique product key – format is an integer.
336	TradingSessionID	Although not part of the FIX Instrument block – the TradingSessionID is required by CBOEdirect.
Optional		
Tag	**Field**	**Usage**
167	SecurityType	"MLEG" – This is required
22	IDSource	Optional, if used must be set to "8" (Exchange)
Not used in this context: *MaturityMonthYear[200], PutOrCall[201], StrikePrice[202]*. *MaturityDay[205]*		

Example identifying a strategy product

Assuming for example purposes the CBOE Product Key for a previously defined IBM option strategy is 1237:

```
55=IBM^48=1237^167=MLEG
```

Assuming for example purposes the CBOE Product Key for a previously defined IBM futures strategy is 3276:

```
55=IBM1C^48=3276^167=MLEG
```

Underlying or Equity-Based product identification using industry standard tags

Underlyings are identified using the symbol and specifying the *SecurityType[167]*.

Table 15 Underlying product identification using industry standard tags		
Required		
Tag	**Field**	**Usage**
55	Symbol	Primary Underlying Ticker (IBM, MSFT, QQQ, SPDR)
167	SecurityType	“CS”, “INDX”, “USTB”
336	TradingSessionID	Although not part of the FIX Instrument block – the TradingSessionID is required by CBOEdirect.
Optional		
Tag	**Field**	**Usage**
		There are no optional tags at this time
Not used in this context: *MaturityMonthYear[200], PutOrCall[201], StrikePrice[202]. MaturityDay[205], SecurityID[48], IDSource[22]*		

Example identifying a stock or equity-based underlying product

The following example shows how to represent IBM stock product.

```
55=IBM^167=CS
```

Example identifying an index underlying product

The following example shows how to represents the OEX index (the index itself not the option on the index).

```
55=OEX^167=INDX
```

Underlying or Equity-Based product identification using CBOE Product Key

Alternative identification using the CBOEdirect product key as the *SecurityID[48]*. CBOEdirect has a unique numeric identifier for every product traded at the exchange. A product can be an option series, a futures contract, an underlying.

Table 16 Underlying product identification using a CBOE Product Key

Required		
Tag	**Field**	**Usage**
55	Symbol	Primary Underlying Ticker
48	SecurityID	CBOE supplied and maintain unique product key – format is an integer.
336	TradingSessionID	Although not part of the FIX Instrument block – the TradingSessionID is required by CBOEdirect.
167	SecurityType	"CS", "INDX", "USTB"
Optional		
Tag	**Field**	**Usage**
22	IDSource	If used must be set to "8" (Exchange)
Not used in this context: *MaturityMonthYear[200], PutOrCall[201], StrikePrice[202],. MaturityDay[205]*		

Example specifying an underlying product using the CBOEdirect ProductKey

Assuming the CBOE Product Key for IBM stock is 90360– the following FIX message will identify the product.

Represents IBM stock

```
55=IBM^48=90360
```

Product Identification on Responses Outbound from CBOE

CBOE fully populates the Security identification block on outbound messages, including the *SecurityID[48]* which contains the CBOE Product key.

Option series - outbound security identification

Option series are identified using the following FIX fields on outbound messages.

Table 17 Option Series Identification – Outbound from CBOE

Tag	Field	Usage
55	Symbol	Standard Option Symbol
167	SecurityType	"OPT"
200	MaturityMonthYear	Will contain the expiration Year and Month
201	PutOrCall	Will contain the Put or Call code
202	StrikePrice	Will contain the strike price in decimal format
207	SecurityExchange	W = CBOE C2OX = CBOE 2
205	MaturityDay	Will contain the actual expiration date as specified in CBOEdirect
48	SecurityID	The CBOEdirect productkey for the option series
22	IDSource	"8"
107	SecurityDesc	Contains the CFICode for the option.

Futures product - outbound security identification

Futures products are identified using the following FIX fields on outbound messages.

Table 18 Futures Product Identification – Outbound from CBOE

Tag	Field	Usage
55	Symbol	Standard Futures Symbol
167	SecurityType	"FUT"
200	MaturityMonthYear	Will contain the expiration Year and Month
207	SecurityExchange	50 = OneChicago
205	MaturityDay	Will contain the actual expiration date as specified in CBOEdirect
48	SecurityID	The CBOEdirect product key for the futures product
22	IDSource	"8"
107	SecurityDesc	Contains the CFICode for the futures product.

Strategy Product - outbound security identification

Option series are identified using the following FIX fields on outbound messages.

Table 19 Strategy Product Identification – Outbound from CBOE

Tag	Field	Usage
55	Symbol	Standard Option Symbol
167	SecurityType	"MLEG"
207	SecurityExchange	W = CBOE; 50 = OneChicago, C2OX = CBOE 2
48	SecurityID	The CBOEdirect productkey for the option series
22	IDSource	"8"
107	SecurityDesc	Will contain the strategy name if the strategy is of a CBOEdirect standard type: COMBO DIAGONAL PSEUDO_STRADDLE RATIO STRADDLE TIME UNKNOWN VERTICAL BUY_WRITE

Underlying or Equity-Based product - outbound security identification

Underlying products are identified using the following FIX fields on outbound messages.

Table 20 Underlying Product Identification – Outbound from CBOE

Tag	Field	Usage
55	Symbol	Primary Underlying Ticker
167	SecurityType	"CS", "INDX", "USTB"
207	SecurityExchange	W = CBOE; 50 = OneChicago, C2OX = CBOE 2
48	SecurityID	The CBOEdirect Product Key for the option series
22	IDSource	"8"
107	SecurityDesc	Contains the CFICode for the underlying product

The ISO CFICode an introduction and overview

The CFICode is an international standard that is used to describe a security in a very granular manner.

The CFICode was adopted in FIX 4.3 as part of FIX 4.3. CBOE is using this field in our implementation of FIX 4.2 to provide access to the security type of the underlying.

Indicates the type of security using ISO 10962 standard, Classification of Financial Instruments (CFI code) values.

Identify Trading Sessions

The FIX *TradingSessionID[336]* is used to identify the market destination for an order.

CBOEdirect requires that you specify a trading session on FIX application messages. CBOE has defined the following trading sessions available for trading at CBOE. The following trading sessions have been defined.

For Options Linkage orders, the only Trading Session a market maker must enter for tag 336 is "W_MAIN". The NBBO exchange destination is established in ExDestination [100].

Table 21 Trading Sessions available from FIX 4.2

TradingSessionID	Description	Market Time
W_MAIN	The regular open outcry markets for equity, index, and interest rate options. Use W_MAIN for Linkage.	8:30AM – 3:00PM Central
ONE_MAIN	OneChicago security futures regular trading hours session	8:30AM - 3:00PM Central (3:15 PM closing for futures on Exchange Traded Funds)
CFE_MAIN	CBOE Futures Exchange (CFE) electronic trading	8:30AM – 3:15PM Central
COF_MAIN	CFE Options on Futures	8:30AM – 3:02PM Central
W_STOCK	Stock Trading On CBOEdirect (formerly traded on SEMS)	8:30AM - 3:00PM Central (3:15 PM closing for futures on Exchange Traded Funds).
C2_MAIN	CBOE fully electronic options exchange	8:30AM – 3:00PM Central

Orders and quotes can only be active in one trading session.

You can only submit an order to one trading session. An order cannot exist in both trading sessions.

You can submit orders to any trading sessions that are accepting orders.

Note that product download using Security Definition Requests will be available starting at 4:30AM Central.

The status of CBOE trading sessions is provided using Trading Session Status Request / Trading Session Status messages.

You must specify the trading session on requests, such as New Order Single. Default values will no longer be supported.

Trading Session Status

Trading Session Status Request Message

The Trading Session Status Request is used to subscribe for trading session status. CBOEdirect reports the opening and closing of trading sessions only at this time. You are not able to query (*SubscriptionRequestType[263]*=0 (Snapshot)) for trading session status - you can only subscribe for status. If backlogs in receiving trading session status occur, the CBOE will automatically unsubscribe the user from the service and send a business message reject reason with a reason text.

A firm must request Trading Session Status messages from CBOEdirect to be a certified user of CBOEdirect.

Table 22 Trading Session Status Request

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
	Standard Header	Y	Y	MsgType = g (lowercase)
335	TradSesReqID	Y	Y	A unique string value that identifies your trading session status request. Your TradSesReqID will be returned on all subsequent Trading Session Status Report messages that are sent as a result of this subscription.
~~*336*~~	~~*TradingSessionID*~~	~~*N*~~	~~*N*~~	NOT SUPPORTED AT THIS TIME. CBOEdirect returns status for all trading sessions. Any value specified for TradingSessionID on input is ignored by the CBOEdirect system.
263	SubscriptionRequestType	Y	Y	1 - Snapshot + Update (Subscribe) 2 - Unsubscribe NOTE: Snapshots (queries) for trading session status are not supported by CBOEdirect.
	Standard Trailer	Y	Y	

Trading Session Status Message

The Trading Session Status is sent from CBOEdirect to FIX users that previously subscribed for Trading Session Status using the Trading Session Status Request (MsgType=g).

A firm must be able to receive and identify Trading Session Status messages from CBOEdirect to be a certified user of CBOEdirect.

Table 23 Trading Session Status

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
	Standard Header	Y	Y	MsgType = h (lowercase)
335	TradSesReqID	N	N	The TradSesReqID that was provided on the Trading Session Status Request.
336	TradingSessionID	Y	Y	Identifier for Trading Session. Possible values include: W_MAIN ONE_MAIN CFE_MAIN COF_MAIN W_STOCK C2_MAIN
325	UnsolicitedIndicator	N	N	'Y' if message is sent unsolicited as a result of a previous subscription request.
340	TradSesStatus	Y	Y	State of the trading session: 2 - OPEN 3 - CLOSED
341	TradSesStartTime	N	N	Scheduled opening time of the opening of the trading session
345	TradSesEndTime	N	N	Scheduled closing time of the trading session
	Standard Trailer	Y	Y	

Security Definition Interface

The Security Definition Interface can be used to retrieve a list of products (such as list of option and futures classes, option series, stock futures, etc.) traded on the CBOE or OneChicago exchange.

The Security Definition message can also be used to define an options strategy, that once defined, can then be traded by specifying the CBOEdirect product key assigned to the strategy product in the SecurityID[48] field on the New Order - Single (MsgType=D), the Quote (MsgType=S), and Mass Quote(MsgType=i) messages.

Security Definition Request

The Security Definition Request message is used for the following:

1. Request information on a specific product within the CBOEdirect system.
2. Define a strategy product a made up of one or more products.
3. Request a list of the Security Types that can be traded on the CBOEdirect system.
4. Request a list of product classes that are available within a CBOEdirect trading session (OEX, DJX, RUT, SPX, etc.)
5. Request a list of products for a specified product class (example a list of option series belonging to a specific product class, such as IBM, DJX, OEX, etc.).
6. Request a list of Securities that can be traded with the second party. This request can optionally be qualified with Symbol, TradingSessionID, SecurityExchange, and Security Type.
7. Subscribe to security status by class.

The Security Definition Request and Security Definition messages in FIX 4.2 are distinct in that they support multiple types of requests and responses. The message is extremely overloaded.[3] Your implementation using the Security Definition Request is likely best implemented as a factory that can produce each type of request and process each type of process.

Implicit Subscription For Security Status (Product States)

Security Status Requests can be performed only by *class* and not by *product* or *session*. However, CBOE publishes security status by individual product only.

A firm must be able to receive and identify Security Status messages from CBOEdirect to be a certified user of CBOEdirect.

Whenever your firm requests a list of securities using Security Definition Request (35=c) for a particular class, it now has an option to subscribe for security status changes/updates with a user defined tag SubscriptionRequestType[9463]. This tag can take in two values: 0 (SNAPSHOT only) and 1 (SNAPSHOT + UPDATES).

By default, if your firm doesn't specify this tag, it would automatically be subscribed to security status changes/updates. Your firm can disable the subscription by specifying 9463=0 in the request.

Implicit Subscription For New Strategy Product Updates

When a firm sends a Security Definition Request to the CBOE for a class of type MLEG (strategy) or subscribes to Security Status for a class of type MLEG (strategy), the firm would automatically be subscribed to the New Strategy Product updates for that class. Therefore, when another party in the marketplace creates a new MLEG product, your firm would receive a Security Definition containing the newly created product.

[3] Please note that the Security Definition Request / Security Definition message functionality was divided up into multiple messages in the FIX 4.3 specification to eliminate the overloading and the resultant confusion and difficulty required to implement this message.

The Underlying* Repeating Group - further overloading

The Security Definition Request and the Security Definition messages both contain an Underlying* repeating group (fields such as UnderlyingSymbol, UnderlyingSecurityID, UnderlyingSecurityType, etc.). The meaning of Underlying* does not mean the underlying product. It was a poor choice of terminology in the specification. The best way to consider the Underlying* block is to forget that it is "Underlying" and view it as a repeating group of products, which can be:

If the *SecurityType[167]*=MLEG, then the Underlying* repeating group contains the product definition for each leg that make up the multileg (strategy) product.

If the *SecurityType[167]* does not equal MLEG, then the Underlying* repeating group will contain the list of products that belong to a product class (for example, it could be the option products that belong to the product class "IBM").

In the case of performing a retrieval of the security types within the CBOEdirect system, then the Underlying* repeating group contains a list of SecurityTypes specified in the UnderlyingSecurityType[310] field.

As you can see these message are quite overloaded!

Types of Requests

The SecurityRequestType[321] is used to specify the type of request. The following table specifies the usage of each request type. As you can tell from the table - this message performs quite a few functions.

Security Request Type [312]	SecurityType [167]	Description of Request	Description of Response
0	MLEG *only*	Define a strategy product to CBOEdirect by explicitly listing each leg of the strategy.	Security Definition containing the strategy definition of the strategy product defined within CBOEdirect. If the strategy product does not already exist, an attempt will be made to create it. If the creation is successful, the return will be exactly the same as if it already existed. If the strategy product already exists, the pre-existing strategy product will be returned. If the strategy did not exist and could not be defined the field is set to SecurityResponseType[323] = 5 (Reject)
1	MLEG	Define a strategy product to CBOEdirect by specifying a strategy type in the SecurityDesc[107] field.	Security Definition containing the strategy definition of the strategy product defined within CBOEdirect. If the strategy did not exist or could not be defined the field is set to SecurityResponseType[323] = 5 (Reject).
1	*not an MLEG product*	Retrieve information for a specific product within the CBOEdirect trading system (can be option, future, or an underlying).	Returns a Security Definition message containing the product information for a specific product within the CBOEdirect system. If the product does not exist, the field is set to SecurityResponseType[323] = 6 (No match)

Security Request Type [312]	SecurityType [167]	Description of Request	Description of Response
2	*all*	Return a list of security types available within the CBOEdirect system.	Returns a Security Definition message with the Underlying repeating group containing a list of SecurityTypes[167] traded within the system.
SecurityRequestType[312]=3 (List of Securities) is by far the most context sensitive of all the request types. Depending on how you populate the main instrument block (Symbol, SecurityID, SecurityType, MaturityMonthYear, StrikePrice, PutOrCall) this particular request can return different types of results. You will need to pay special attention on how to format your Security Definition Request so you obtain the desired results.			
3	*not an MLEG product class*	Returns a list of Product Classes if the Symbol[55] is omitted from request.	Will return one to N Security Definition messages with a maximum of 100 product classes per message. The Underlying* repeating group will contain up to 100 product classes.
3	*not an MLEG product class*	Return a list of products that match the product class selection criteria if the Symbol[55] contains a valid Product Class Symbol (such as options: "IBM", "DJX", "AOL", "OEX", "MSQ"; futures: "MSFT1C")	Will return one to N Security Definition messages with a maximum of 100 products per message. Each Security Definition message will contain the specification for the product class. The Underlying* repeating group will contain up to 100 of the products that belong to the product class. If the product does not exist, the field is set to SecurityResponseType[323] = 6 (No match)
3	*not an MLEG product class*	Returns the definition for a specific product, if the product is fully specified on the request.	Will return one Security Definition message for the product. The Underlying* repeating group will not be populated. If the product does not exist, the field is set to SecurityResponseType[323] = 6 (No match)
3	MLEG	Return a list of strategy products that match the product class selection criteria.	Will return one to N Security Definition messages. Each Security Definition message will contain the specification for one Strategy Product. The Underlying* repeating group will contain the legs that make up the Strategy product. If the product does not exist, the field is set to SecurityResponseType[323] = 6 (No match)

Table 24 Security Definition Request

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
	Standard Header	Y	Y	*MsgType[35]* = c (lowercase)
320	SecurityReqID	Y	Y	This is the user specified request identifier to identify the request. The content is an alphanumeric (including punctuation characters). This identifier will be returned on all Security Definition and Security Status messages pertaining to this request.
321	SecurityRequestType	Y	Y	0 - Request identity of security by name 1 - Request identity of security 2 - List security types 3 - List securities
55	Symbol	N	Y	For SecurityRequestType[312] equal 0 or 1 and SecurityType[167] = "MLEG" should contain the product class symbol (examples: options: OEX, DJX, MSQ; futures: MSFT1C, IBM1C). For SecurityRequestType[312] equal 1 and SecurityType[167] **NOT** equal "MLEG" should contain the product ticker symbol (examples: options: OEZ, OEX, OEY, MSW, MSQ, DJV, DJX) For SecurityRequestType[312] equal 2 this field should be blank. For SecurityRequestType[312] equal 3 this field should contain the Product Class (examples: options: OEX, DJX, MSQ; futures: MSFT1C, IBM1C)
48	SecurityID	N	N	The CBOEdirect Product Key
22	IDSource	N	N	Ignored on messages received from firms, if specified it should be set to "8" meaning that the SecurityID is an exchange proprietary identifier.
167	SecurityType	N	Y	"OPT", "CS", "FUT", "INDX", "MLEG", "USTB"— Required for “by Name” option product look up or if SecurityID[tag 48] is not used.
200	MaturityMonthYear	N	Y	Expiration Year and Month in YYYYMM format (Example: "200204"). Required for “by Name” option product look up or if SecurityID[tag 48] is not used.
205	MaturityDay	N	Y	Required for “by Name” option product look up or if SecurityID[tag 48] is not used. Specifies the actual expiration day.
201	PutOrCall	N	Y	Used for options only. 0 = Put, 1 = Call. Required for “by Name” option product look up or if SecurityID[tag 48] is not used.
202	StrikePrice	N	Y	Used for options only. Valid values: 0 - 99999999.9999 Required for “by Name” option product look up or if SecurityID[tag 48] is not used.
107	SecurityDesc	N	N	For a standard defined strategy product (such as a COMBO or STRADDLE) will contain the Strategy Type. Not used for standard products.

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
336	TradingSessionID	N	N	W_MAIN ONE_MAIN CFE_MAIN COF_MAIN W_STOCK C2_MAIN
9463	SubscriptionRequestType	N	N	This tag would be specified if the firm wants to subscribe to security status (product state callbacks). If not specified, the firm would be subscribed to security status by default. 0 = SNAPSHOT only 1 = SNAPSHOT + UPDATES 500=subscribe for auction participation 501=unsubscribe from auction participation
146	NoRelatedSym	N	N	
Begin Repeating Group for NoRelatedSym				
311	→UnderlyingSymbol	N	N	The UnderlyingSymbol must be specified as the first field in the repeating group. Required if NoRelatedSym is used. When returning a list of SecurityTypes - the value is set to "NA".
309	→UnderlyingSecurityID	N	N	SecurityID of the product
305	→UnderlyingIDSource	N	N	This field is ignored on input
310	→UnderlyingSecurityType	N	N	SecurityType of the product
313	→UnderlyingMaturityMonthYear	N	N	Maturity month and year for a product, if the product is an option or future.
314	→UnderlyingMaturityDay	N	N	Recommended that this field be omitted on the request.
315	→UnderlyingPutOrCall	N	N	Used for options only. 0 = Put, 1 = Call
316	→UnderlyingStrikePrice	N	N	Used for options only. Valid values: 0 - 99999999.9999
308	→UnderlyingSecurityExchange	N	N	Ignored on input
307	→UnderlyingSecurityDesc	N	N	Ignored on input
319	→RatioQty	N	N	Quantity of particular leg in a strategy product. This is the multiplier, which is multiplied by the order quantity to determine the quantity of a particular product that will be bought or sold as part of the strategy product. Only used for strategy products (SecurityType[167]="MLEG") Note: The RatioQty values of all the legs for a strategy product are reduced to their lowest common denominator before being used as a multiplier.
54	→Side	N	N	Indicates if this leg of the security is to be Bought or Sold as part of a strategy product. Only used for strategy products (SecurityType[167]="MLEG")
End Repeating Group				

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
	Standard Trailer	Y	Y	

Security Definition

The Security Definition message is used for the following:

1. Accept the security defined in a Security Definition message.
2. Accept the security defined in a Security Definition message with changes to the definition and/or identity of the security.
3. Reject the security requested in a Security Definition message
4. Return a list of Security Types
5. Return a list of Securities
6. Return trading group information by class

Table 25 Security Definition

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
	Standard Header	Y	Y	*MsgType* = d (lowercase)
320	SecurityReqID	Y	Y	This will contain the SecurityReqID[320] that was provided on the Security Definition Request corresponding to this Security Definition message.
322	SecurityResponseID	Y	Y	Identifier for the Security Definition message which is generated by the CBOEdirect FIX adapter. This contains a unique identifier for each Security Definition message returned in response to a Security Definition Request. This is provided because it is possible to receive multiple Security Definition messages for a single Security Definition Request. For example, requesting a list of products (option series) for an option class will result in 100 products being returned per Security Definition message.
323	SecurityResponseType	N	N	1 - Accepted the security definition request for a specific security or strategy as defined in the request. 2 - Accepted the security definition request for a specific product or strategy with revisions, such as returning a strategy product with an opposite definition in terms of buy and sell sides per leg. 3 - Response contains a list of security types traded on the CBOEdirect system. 4 - Response contains a list of products. 5 - Request was rejected. 6 - The product specified in the request was not found (No match)
55	Symbol	N	N	
48	SecurityID	N	N	
22	IDSource	N	N	"8"
167	SecurityType	N	N	OPT, CS, FUT, INDX, MLEG, USTB
200	MaturityMonthYear	N	N	Expiration Year and Month (Proper format: 200204)

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
205	MaturityDay	N	N	Will contain the Maturity expiration day of the month for derivative products.
201	PutOrCall	N	N	Used for options only. 0 = Put, 1 = Call
202	StrikePrice	N	N	Used for options only. Valid values: 0 - 99999999.9999
207	SecurityExchange	N	N	W = CBOE; 50 = OneChicago, C2OX = CBOE 2
107	SecurityDesc	N	N	Used for the CFICode for a future, option, index, underlying, or equity product. This is the strategy type if a standard strategy product.
336	TradingSessionID	N	N	W_MAIN ONE_MAIN CFE_MAIN COF_MAIN W_STOCK C2_MAIN
58	Text	N	N	Will contain one of the following to help identify the content of the Security Definition: "CLASS_LIST" "PRODUCT_LIST" "SECTYPE_LIST" "STRATEGY_LIST" "STRATEGY_DEFN"
393	TotNumSecurities	N	N	Total Number of Securities being returned across all Security Definition messages sent as a response to the Security Definition Request identified by *SecurityReqID[320]*
146	NoRelatedSym	N	N	Number of securities returned as part of the definition.
Begin Repeating Group for NoRelatedSym				
311	→UnderlyingSymbol	N	N	The Symbol must be specified as the first field in the repeating group. Required if *NoRelatedSym* >0
309	→UnderlyingSecurityID	N	N	CBOEdirect product key for the product
305	→UnderlyingIDSource	N	N	"8"
310	→UnderlyingSecurityType	N	N	OPT, CS, FUT, INDX, MLEG, USTB
313	→UnderlyingMaturityMonthYear	N	N	Expiration Year and Month (Proper format: 200204)
314	→UnderlyingMaturityDay	N	N	Will contain the Maturity expiration day of the month for derivative products.
315	→UnderlyingPutOrCall	N	N	Used for options only. 0 = Put, 1 = Call
316	→UnderlyingStrikePrice	N	N	Used for options only. Valid values: 0 - 99999999.9999
308	→UnderlyingSecurityExchange	N	N	
307	→UnderlyingSecurityDesc	N	N	Will contain the CFICode if the Security Definition contains a list of classes (*Text[58]* ="CLASS_LIST")
319	→RatioQty	N	N	Quantity of particular leg in a strategy product. Only specified if SecurityType[167]="MLEG" Note: The RatioQty values of all the legs for a strategy product are reduced to their lowest common denominator before being used as a multiplier.

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
436	→UnderlyingContractMultiplier	N	N	When the CBOE publishes a Security Definition for all classes in a particular trading session, this tag will contain the Contract Size (shares per contract, or SPC) for each class. This value will usually = 100 unless corporate actions (stock splits, etc.) have caused it to change.
54	→Side	N	N	Indicates if this leg of a strategy product is to be Bought or Sold. Only specified if SecurityType[167]="MLEG"
20010	→TradingGroupInfo	N	N	Provides trading group information by class. The value will equal the trade server environment. Example: 20010=test1EquityTradeServer1
End Repeating Group				
The following custom fields are used by CBOE (note: they are not part of the repeating group)				
9365	PremPriceTickBreakPoint	N	N	The price at which the Premium Price Tick Size changes from the PremPriceTickBelow size to the PremPriceTickAbove size
9366	PremPriceTickAbove	N	N	The Premium Price Tick size when the premium price is above the PremPriceTickBreakPoint
9367	PremPriceTickBelow	N	N	The Premium Price Tick size when the premium price is below the PremPriceTickBreakPoint
	Standard Trailer	Y	Y	

Security Definition Message Fragmentation

The Security Definition message is designed to be able to transmit the response to a request across multiple messages. CBOEdirect will break up requests providing only 100 securities per message.

The *TotNumSecurities[393]* field will contain the total number of securities that will be provided in the response across all messages. The *NoRelatedSym[146]* field will contain the number of securities provided within a message.

Security Definition Usage

Obtain a list of security types available at the exchange

Request for list of security types

```
8=FIX.4.2^35=c^49=LLO^56=TESTFIX201^320=REQUEST_ID^321=2^
```

Response for list of security types

```
8=FIX.4.2^9=0614^35=d^49=TESTFIX201^56=ABC^34=61^52=20010504-
15:27:33^320=REQUEST_ID^322=988990054943^323=3^22=8^
393=11^146=11^
311=NA^305=8^310=MLEG^307=STRATEGY ^
311=NA^305=8^310=WAR^307=WARRANT ^
311=NA^305=8^310=VIX^307=VOLATILITY_INDEX ^
311=NA^305=8^310=UIT^307=UNIT_INVESTMENT_TRUST^
311=NA^305=8^310=OPT^307=OPTION
311=NA^305=8^310=LNKNT^307=LINKED_NOTE
311=NA^305=8^310=INDX^307=INDEX
311=NA^305=8^310=FUT^307=FUTURE^
311=NA^305=8^310=CS^307=EQUITY^
311=NA^305=8^310=USTB^307=DEBT^
311=NA^305=8^310=CMDTY^307=COMMODITY
```

Obtain a list of security types in a trading session

Request for list of security types available in W_MAIN Trading Session

```
8=FIX.4.2^35=c^49=LLO^56=TESTFIX201^320=REQUEST_ID^321=2^336=W_MAIN^
```

Response for list of security types available in W_MAIN Trading Session

```
8=FIX.4.2^9=0204^35=d^49=TESTFIX201^56=ABC^34=62^52=20010504-
15:27:50^320=REQUEST_ID^322=988990072371^323=3^22=8^336=W_MAIN^393=1^146=1^311=N
A^305=8^310=OPT^307=OPTION^
```

NOTE: *REQUEST_ID* you create in can be any string that you want to identify the security definition request. Examples could be: *REQ1*, *ID1*, *MyRequest*, a time stamp, etc.

Request for list of security types available in Underlying Trading Session

```
8=FIX.4.2^35=c^49=LLO^56=TESTFIX201^320=REQUEST_ID^321=2^336=Underlying^
```

Response for list of security types available in Underlying Trading Session

```
8=FIX.4.2^9=0202^35=d^49=TESTFIX201^56=ABC^34=64^52=20010504-
15:28:09^320=REQUEST_ID^322=988990090636^323=3^22=8^336=Underlying^
393=2^146=2^
311=NA^305=8^310=INDX^307=INDEX^
311=NA^305=8^310=CS^307=EQUITY
```

NOTE: *REQUEST_ID* can be any string that you want to identify the security definition request.

Obtain a list of classes available at the exchange

Request for list of classes traded on exchange for a particular security type

```
8=FIX.4.2^35=c^49=LLO^56=TESTFIX201^320=LIST_CLASSES^321=3^167=OPT^
```

Response for list of classes traded on exchange

```
8=FIX.4.2^9=3575^35=d^49=TESTFIX201^56=ABC^34=71^52=20010504-
15:30:38^320=LIST_CLASSES^322=0^323=4^22=8^393=1544^146=100^311=ATSN^309=2753509
^305=8^310=OPT^311=MEDI^309=2759632^305=8^310=OPT^311=ICOS^309=2758096^305=8^310
=OPT^311=AMZ1^309=2753105^305=8^310=OPT^311=MRVC^309=2759950^305=8^310=OPT^311=E
DS^309=2756003^305=8^310=OPT^311=TVLY^309=2763873^305=8^310=OPT^311=HPZ^309=2757
759^305=8^310=OPT^311=BFRE^309=2753922^305=8^310=OPT^311=AWE^309=2753635^305=8^3
10=OPT^311=HPT^309=2757752^305=8^310=OPT^311=PGEX^309=2761177^305=8^310=OPT^311=
OIZ^309=2760789^305=8^310=OPT^311=OIX^309=2760782^305=8^310=OPT^311=AVT^309=2753
621^305=8^310=OPT^311=RMBS^309=2754151^305=8^310=OPT^311=AVP^309=2753612^305=8^3
10=OPT^311=ACRU^309=2752649^305=8^310=OPT^311=OIL^309=2760769^305=8^310=OPT^311=
RUT^309=2762103^305=8^310=OPT^311=AVI^309=2753587^305=8^310=OPT^311=HOW^309=2757
731^305=8^310=OPT^311=AVE^309=2753572^305=8^310=OPT^311=ECM^309=2755987^305=8^31
0=OPT^311=HOU^309=2757724^305=8^310=OPT^311=HOT^309=2757717^305=8^310=OPT^311=AV
A^309=2753556^305=8^310=OPT^311=HON^309=2757707^305=8^310=OPT^311=LSCC^309=27593
09^305=8^310=OPT^311=SIFY^309=2762502^305=8^310=OPT^311=CSTR^309=2755388^305=8^3
10=OPT^311=HNZ^309=2757690^305=8^310=OPT^311=DVSA^309=2755938^305=8^310=OPT^311=
CINR^309=2754823^305=8^310=OPT^311=DEST^309=2755718^305=8^310=OPT^311=MOGN^309=2
759848^305=8^310=OPT^311=JMED^309=2758774^305=8^310=OPT^311=IMRS^309=2758321^305
=8^310=OPT^311=ATU^309=2753525^305=8^310=OPT^311=I-
USH01^309=2757938^305=8^310=OPT^311=TOPP^309=2763620^305=8^310=OPT^311=I-
USH00^309=2757930^305=8^310=OPT^311=TLCV^309=2763495^305=8^310=OPT^311=RSH^309=2
762062^305=8^310=OPT^311=RSG^309=2762055^305=8^310=OPT^311=TLCP^309=2763487^305=
8^310=OPT^311=SHPGY^309=2762473^305=8^310=OPT^311=TLCM^309=2763479^305=8^310=OPT
^311=CBST^309=2754499^305=8^310=OPT^311=FHCC^309=2756597^305=8^310=OPT^311=YLJ^3
09=2765122^305=8^310=OPT^311=HMA^309=2757676^305=8^310=OPT^311=ASO^309=2753440^3
05=8^310=OPT^311=CBSI^309=2754491^305=8^310=OPT^311=ASF^309=2753392^305=8^310=OP
T^311=ASD^309=2753385^305=8^310=OPT^311=HLT^309=2757640^305=8^310=OPT^311=HLR^30
9=2757633^305=8^310=OPT^311=ASA^309=2753368^305=8^310=OPT^311=LVLT^309=2759407^3
05=8^310=OPT^311=CSRE^309=2755381^305=8^310=OPT^311=IMPX^309=2758314^305=8^310=O
PT^311=MERCS^309=2759686^305=8^310=OPT^311=OEX^309=2760727^305=8^310=OPT^311=ARM
^309=2753352^305=8^310=OPT^311=OEM^309=2760720^305=8^310=OPT^311=IMPH^309=275830
6^305=8^310=OPT^311=SLOT^309=2762608^305=8^310=OPT^311=OEI^309=2760713^305=8^310
=OPT^311=AMTD^309=2753093^305=8^310=OPT^311=ATMS^309=2753501^305=8^310=OPT^311=A
TML^309=2753310^305=8^310=OPT^311=AMSY^309=2753076^305=8^310=OPT^311=ERTS^309=27
56315^305=8^310=OPT^311=ICIX^309=2758079^305=8^310=OPT^311=RPZ^309=2762023^305=8
^310=OPT^311=ODP^309=2760703^305=8^310=OPT^311=KEYN^309=2758880^305=8^310=OPT^31
1=ZILA^309=2765222^305=8^310=OPT^311=ERTH^309=2756307^305=8^310=OPT^311=TLAB^309
=2763469^305=8^310=OPT^311=BMET^309=2754102^305=8^310=OPT^311=ICII^309=2758072^3
05=8^310=OPT^311=RPM^309=2762016^305=8^310=OPT^311=ZBRA^309=2765171^305=8^310=OP
T^311=IMNX^309=2758291^305=8^310=OPT^311=DBCC^309=2755647^305=8^310=OPT^311=APW^
309=2753303^305=8^310=OPT^311=IMNR^309=2758283^305=8^310=OPT^311=FDRY^309=275654
6^305=8^310=OPT^311=NABI^309=2760143^305=8^310=OPT^311=ROX^309=2762009^305=8^310
=OPT^311=ACLS^309=2752641^305=8^310=OPT^311=ROS^309=2762002^305=8^310=OPT^311=SI
AL^309=2757985^305=8^310=OPT^311=APH^309=2753264^305=8^310=OPT^311=HIT^309=27576
16^305=8^310=OPT^311=APC^309=2753240^305=8^310=OPT^311=ROK^309=2761995^305=8^310
=OPT^311=TLXNE^309=2763517^305=8^310=OPT^
```

NOTE: There were multiple Security Definition messages sent in response to the request – only one of these messages is being shown in this example to save space.

Obtain a list of product classes available in a Trading Session for a particular security type

```
8=FIX.4.2^35=c^49=M^56=CBOEFIX001^320=SecReq13^321=3^167=OPT^336=W_MAIN^
```

Obtain a list of products for a product class of a particular security type in a Trading Session

```
8=FIX.4.2^35=c^49=M^56=CBOEFIX001^320=SecReq15^321=3^55=IBM^167=OPT^336=W_MAIN^
```

Due to the extremely large number of classes and products available for trading in the W_MAIN session on the CBOE trading floor, CBOE recommends performing product downloads only for those products that the trader wishes to trade and not all products available to trade in all classes in all sessions.

Query for a specific product

```
8=FIX.4.2^35=c^49=SMJN01^56=PFIX01^320=SecReq16^321=3^55=IBM^167=OPT^200=200101^
201=1^ 202=75.0^
```

Request a list of existing strategy products

```
8=FIX.4.2^9=0097^35=c^34=3^49=ZZCBOE4^56=DFIX203^52=20020923-
15:53:49^320=LIST_1^321=3^55=NDX^336=W_MAIN^167=MLEG^10=093^
```

List of Strategy Products Response

There are three strategy products returned in this example.

```
8=FIX.4.2^9=0286^35=d^34=3^49=DFIX203^56=ZZCBOE4^52=20020923-
15:53:49^320=LIST_1^322=LIST_1^323=1^55=NDX^48=8323079^22=8^167=MLEG^107
=2^336=W_MAIN^393=3^146=2^311=NDT^309=7280129^310=OPT^313=200209^314=21^
315=1^316=2375^319=1^54=1^311=NDT^309=7280130^310=OPT^313=200209^314=21^
315=0^316=2375^319=1^54=1^10=055^

8=FIX.4.2^9=0286^35=d^34=4^49=DFIX203^56=ZZCBOE4^52=20020923-
15:53:49^320=LIST_1^322=LIST_1^323=1^55=NDX^48=8585253^22=8^167=MLEG^107
=8^336=W_MAIN^393=3^146=2^311=NDT^309=7280169^310=OPT^313=200212^314=21^
315=1^316=2400^319=1^54=1^311=NDT^309=7280170^310=OPT^313=200212^314=21^
315=0^316=2400^319=1^54=2^10=041^

8=FIX.4.2^9=0286^35=d^34=5^49=DFIX203^56=ZZCBOE4^52=20020923-
15:53:49^320=LIST_1^322=LIST_1^323=1^55=NDX^48=8323146^22=8^167=MLEG^107
=2^336=W_MAIN^393=3^146=2^311=NDV^309=7280295^310=OPT^313=200212^314=21^
315=1^316=1500^319=1^54=1^311=NDV^309=7280296^310=OPT^313=200212^314=21^
315=0^316=1500^319=1^54=1^10=039^
```

Subscribing to security status in security definition request

The two Security Definition Request examples below subscribe the user to product status updates. Either of the two following requests would return the same result, which is a list of products for class MSFT in session ONE_MAIN as well as subscribe the user to the product state updates for each of these products.

```
8=FIX.4.2^A35=c^A49=AATG0^A56=DFIX201^A57=TEST^A52=20020725-
15:14:34^A320=5195^A321=3^A55=MSFT^A167=FUT^A336=ONE_MAIN^A
```

or

```
8=FIX.4.2^A35=c^A49=AATG0^A56=DFIX201^A57=TEST^A52=20020725-
15:14:34^A320=5195^A321=3^A55=MSFT^A167=FUT^A336=ONE_MAIN^A9463=1^A
```

The Security Definition Request example below would only return the list of products for MSFT in ONE_MAIN and the firm would not be subscribed to security status changes/updates. The firm would have to separately subscribe to security status changes using Security Status request (35=e).

```
8=FIX.4.2^A35=c^A49=AATG0^A56=DFIX201^A57=TEST^A52=20020725-
15:14:34^A320=5195^A321=3^A55=MSFT^A167=FUT^A336=ONE_MAIN^A9463=0^A
```

Receive trading group information by Class

By default, TradingGroupInfo [20010] in the Security Definition Messages provides the trading group information for that class.

```
8=FIX.4.2^9=3972^35=d^34=127^49=DFIX101^56=BEN1^52=20100115-
4:40:03^320=XPZ0001-
20100114^322=61^323=4^22=8^336=W_STOCK^393=6165^146=65^311=SUPX^309=2371
09291^305=8^310=CS^436=1^308=NASD^20010=test1EquityTradeServer1^311=SIVB
O^309=583798158^305=8^310=CS^436=1^308=NASD^20010=test1EquityTradeServer
1^311=HLCS^309=537797736^305=8^310=CS^436=1^308=NASD"20010=test1EquityTr
adeServer1"^311=SUPR^309=69221411^305=8^310=CS^436=1^308=NASD^20010=test
1EquityTradeServer1^
```

Defining Strategy Products

CBOEdirect can be used to define (create) strategy products (multileg instruments).

Define Strategy Product Request

The example below demonstrates how to request the security definition of a strategy product by specifying its individual legs. If the strategy product requested does not exist, it will be created.

```
8=FIX.4.2^9=0210^35=c^34=2^49=ZZCBOE4^56=DFIX203^52=20020923-
15:53:39^320=AQ_AR^321=1^55=NDX^336=W_MAIN^167=MLEG^146=2^311=LCY^310=OP
T^313=200401^315=1^316=17.50^54=1^319=1^311=LCY^310=OPT^313=200401^315=0
^316=17.50^54=1^319=1^10=132^
```

Define Strategy Product Response

```
8=FIX.4.2^9=0226^35=d^34=2^49=DFIX203^56=ZZCBOE4^52=20020923-
15:53:46^320=AQ_AR^322=1032796427037^323=1^55=NDX^48=9109511^22=8^167=ML
EG^336=W_MAIN^393=1^146=2^311=LCY^310=OPT^313=200401^315=1^316=17.5^311=
LCY^310=OPT^313=200401^315=0^316=17.5^10=216^
```

Subscribing for Security Status

Security Status Request

The Security Status Request message permits you to request the status of a group of securities in a class. One or more Security Status messages are returned as a result of a Security Status Request message. Keep in mind that it will be most efficient to perform the implicit subscription for security status in the Security Definition Request message. If your firm subscribes to security status updates while requesting securities, then CBOE will not allow your firm to perform a secondary Security Status Request on the same product class. The request id will be the same as that provided on the original Security Definition Request. When CBOE publishes security status, it will always be individual product status for each option or each future, not the whole class.

Security Status Requests can be performed only by *class* and not by *product* or *session*.

Implicit Subscription For New Strategy Product Updates

When a firm subscribes to Security Status for a class of type MLEG (strategy), the firm would automatically be subscribed to the New Strategy Product updates for that class. Therefore, when another party in the marketplace creates a new MLEG product, your firm would receive a Security Definition containing the newly created product.

Subscription Types

The Security Status Request message contains a *SubscriptionRequestType[263]* field. This tells the counter party what type of request is being made:

0 – indicates that the requestor only wants a snapshot or the current status.

1 – indicates that the requestor wants a snapshot (the current status) plus updates as the status changes. This is similar to subscribing for information and can be implemented in applications as a subscription mechanism. Keep in mind that obtaining a list of products for a product class in a specific trading session will result in the automatic subscription for security status for each product belonging to the class.

2 – is used to cancel a previous subscription request (unsubscribe). This is an optional operation since at time of logout, a user will automatically be unsubscribed to all previous subscriptions. The CBOEdirect FIX 4.2 service keeps track of subscription requests by *SecurityStatusReqID[324]* and product. You must specify the *SecurityStatusReqID[324]* and product information for the security status subscription you wish to cancel.

Table 26 Security Status Request

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
	Standard Header	Y	Y	MsgType = e (lowercase)
324	SecurityStatusReqID	Y	Y	Must be unique, or the ID of previous Security Status Request to disable if SubscriptionRequestType = Disable previous Snapshot + Updates Request (2).
55	Symbol	Y	Y	Product symbol for which security status is being obtained.
22	IDSource	N	N	Ignored by CBOEdirect FIX 4.2 Service on messages received from firms.
167	SecurityType	N	N	OPT, CS, FUT, INDX, MLEG, USTB
263	SubscriptionRequestType	Y	Y	SubcriptionRequestType indicates to the other party what type of response is expected. A snapshot request (“0”) only asks for current information. A subscribe request (“1”) asks for updates as the status changes. Unsubscribe (“2”) will cancel any future update messages from the counter party.

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
336	TradingSessionID	N	N	W_MAIN ONE_MAIN CFE_MAIN COF_MAIN W_STOCK C2_MAIN
	Standard Trailer	Y	Y	

Security Status

The Security Status message provides for the ability to report changes in status to a security. The Security Status message contains fields to indicate trading status, corporate actions, and financial status of the company. The Security Status message is used by one trading entity (for instance an exchange) to report changes in the state of a security. Security Status messages containing product state information will be issued each time a product state change occurs.

At the open, CBOE sends thousands of product status messages at a high rate, which has proven to be an impact on the available CPU of CBOE and customer FIX servers, as well as customer bandwidth. To streamline this data, CBOE has implemented an ***alternate*** method of disseminating product status using existing standard FIX messaging. This method will handle Security Status messages in the compact mass message format. The Security Status messages encapsulated in the Security Definition message will be used to support this format. To access this alternate method, users will need to enable the Compact Product Status (COMPPRODSUB) in TargetSubID[57] at login. CBOE will continue support of the current product status model as the default and will use the new method only upon request.

It is expected that the Security Status message that is sent as a response should indicate what type of request is being provided. If the message is being generated as a result of a RequestType =1, then the response should have a RequestType=1 to permit the requestor to determine why the message was sent. If backlogs in receiving security status occur, the CBOE will automatically unsubscribe the user from the service and send a business message reject reason with a reason text.

A firm must be able to receive and identify Security Status messages from CBOEdirect to be a certified user of CBOEdirect.

Table 27 Security Status

Tag	Field Name	Comments
	Standard Header	*MsgType[35]* = f (lowercase)
324	SecurityStatusReqID	The request id
55	Symbol	Product symbol for which security status is being provided.
48	SecurityID	CBOEdirect product key.
22	IDSource	Field will be set to “8” (Exchange proprietary symbol)
167	SecurityType	OPT, CS, FUT, INDX, MLEG, USTB
200	MaturityMonthYear	For Options or Futures to specify the month and year of maturity. Format: YYYYMM (e.g., 200204)
205	MaturityDay	Specifies the actual expiration day. For example, specifying “1” will mean the option expires on the first day of the month. For Options or Futures and can be used in conjunction with MaturityMonthYear to specify a particular maturity date.
201	PutOrCall	0 = Put, 1 = Call
202	StrikePrice	Used for options only
207	SecurityExchange	W = CBOE; 50 = OneChicago, C2OX = CBOE 2
107	SecurityDesc	Strategy Products: Strategy product names as described in the CBOEdirect W_MAIN / ONE Strategy Order Test Plan. Future, Option, Index, or Underlying Products: Contains the CFICode for the underlying product
336	TradingSessionID	W_MAIN, or ONE_MAIN
325	UnsolicitedIndicator	Set to ‘Y’ if message is sent as a result of a subscription request not a snapshot request

Tag	Field Name	Comments
326	SecurityTradingStatus	Identifies the trading status applicable to the product. 2 = Halted 17 = Open 18 = Closed 21 = Pre-Open 22 = Opening Rotation 23 = Fast 24 = ON_HOLD 25 = OFF_HOLD 26 = Temporarily_Not_Available
60	TransactTime	Time security status changed
	Standard Trailer	

Security Status Examples

Subscription for Security Status

```
8=FIX.4.2^35=e^49=LLO^56=TESTFIX201^324=STAT_SUBSCRIBE^55=MSFT^167=OPT^263=1^336
=W_MAIN^
```

Response to Security Status Subscription

```
8=FIX.4.2^9=0112^35=f^49=TESTFIX201^56=ABC^34=110^52=20010504-
15:33:51^324=STAT_SUBSCRIBE^55=MSFT^167=OPT^336=W_MAIN^325=N^326=20^
```

Unsubscribe for Security Status

```
8=FIX.4.2^35=e^49=LLO^56=TESTFIX201^324=STAT_SUBSCRIBE^55=MSFT^167=OPT^263=2^336
=W_MAIN^
```

Table 28 Security Definition message used to send Security Status messages in the compact mass message format

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
	Standard Header	Y	Y	*MsgType* = d (lowercase)
320	SecurityReqID	Y	Y	This will contain the SecurityReqID[320] that was provided on the Security Definition Request corresponding to this Security Definition message.
322	SecurityResponseID	Y	Y	Identifier for the Security Definition message which is generated by the CBOEdirect FIX adapter. This contains a unique identifier for each Security Definition message returned in response to a Security Definition Request. This is provided because it is possible to receive multiple Security Definition messages for a single Security Definition Request. For example, requesting a list of products (option series) for an option class will result in 100 products being returned per Security Definition message.

Tag	Field Name	FIX Req'd	CBOE Req'd	Comments
323	SecurityResponseType	N	N	1 - Accepted the security definition request for a specific security or strategy as defined in the request. 2 - Accepted the security definition request for a specific product or strategy with revisions, such as returning a strategy product with an opposite definition in terms of buy and sell sides per leg. 3 - Response contains a list of security types traded on the CBOEdirect system. 4 - Response contains a list of products. 5 - Request was rejected. 6 - The product specified in the request was not found (No match) 7 - Response to a 'snapshot' product status request 8 - Response to a 'snapshot + update' product status request
393	TotNumSecurities	N	N	Total Number of Securities being returned across all Security Definition messages sent as a response to the Security Definition Request identified by *SecurityReqID[320]*
146	NoRelatedSym	N	N	Number of securities returned as part of the definition.
Begin Repeating Group for NoRelatedSym				
311	→UnderlyingSymbol	N	N	The Symbol must be specified as the first field in the repeating group. Required if *NoRelatedSym* >0
309	→UnderlyingSecurityID	N	N	CBOEdirect product key for the product
305	→UnderlyingIDSource	N	N	"8"
310	→UnderlyingSecurityType	N	N	OPT, CS, FUT, INDX, MLEG, USTB
313	→UnderlyingMaturityMonthYear	N	N	Expiration Year and Month (Proper format: 200204)
314	→UnderlyingMaturityDay	N	N	Will contain the Maturity expiration day of the month for derivative products.
315	→UnderlyingPutOrCall	N	N	Used for options only. 0 = Put, 1 = Call
316	→UnderlyingStrikePrice	N	N	Used for options only. Valid values: 0 - 99999999.9999
308	→UnderlyingSecurityExchange	N	N	W = CBOE; 50 = OneChicago, C2OX = CBOE 2
307	→UnderlyingSecurityDesc	N	N	Identifies the trading status applicable to the product. 2 = Halted 17 = Open 18 = Closed 21 = Pre-Open 22 = Opening Rotation 23 = Fast 24 = ON_HOLD 25 = OFF_HOLD 26 = Temporarily_Not_Available
54	→Side	N	N	Indicates if this leg of a strategy product is to be Bought or Sold. Only specified if SecurityType[167]="MLEG"
End Repeating Group				
	Standard Trailer	Y	Y	

Security Definition Message Fragmentation

The Security Definition message is designed to be able to transmit the response to a request across multiple messages. CBOEdirect will break up requests providing only 100 securities per message.

The *TotNumSecurities[393]* field will contain the total number of securities that will be provided in the response across all messages. The *NoRelatedSym[146]* field will contain the number of securities provided within a message.

Example for Security Status in a Security Definition message:

Response to a security status request, indicating that the request was a snapshot (323=7), where the total number of security status products is 103 (393=103), a 100 having been previously sent, the remaining 3 (146=3) contained in this request. Tag 322=1 indicates that this is the second message of the set (a previous one with 322=0 having been sent, containing the first 100 products). Finally, Tag 307=22 indicates that two of the three products here are in Opening Rotation, while Tag 307=17 shows that one product is open.

```
8=FIX.4.2^ 9=0654^ 35=d^ 34=83^ 49=PFIX203^ 56=SOX^ 52=20051003-12:07:36^ 320=FxMg006_0008^
322=1^ 323=7^ 336=W_MAIN^ 393=103^ 146=3^ 311=OAD^ 309=157286498^ 305=8^ 310=OPT^ 313=200701^
314=20^ 315=1^ 316=30^ 308=W^ 307=22^ 311=CBQ^ 309=111018588^ 305=8^ 310=OPT^ 313=200601^
314=21^ 315=1^ 316=35^ 308=W^ 307=22^ 311=CBQ^ 309=214567303^ 305=8^ 310=OPT^ 313=200511^
314=19^ 315=0^ 316=45^ 308=W^ 307=17^ 10=232
```

FIX Field (Tag) Dictionary

Standard FIX Fields

This table contains the usage of standard FIX 4.2 fields in the CBOE FIX 4.2 Service.

Table 29 Standard FIX Tag Dictionary

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
1	Account	N	N	String	Account number as sent by originating firm that appears on trade match reports and is passed through to OCC. Each user can use only one account per product class. Accounts must be approved by the firm's Exchange Membership Department and must be preconfigured by your API testing representative. Exact size is 3 and data type is alphabetic only.	Account mnemonic as agreed between broker and institution.	Order, Cancel, Replace, Execution Report
6	AvgPx	Y	Y	Float	Will always be 0.0. The FIX 4.2 Service requires the firm to calculate and store the average execution price. FIX 4.2 will properly calculate and distribute this value.	Calculated average price of all fills on this order. Valid values: 0 - 99999999.9999 (number of decimal places may vary and not limited to four)	Execution Report
7	BeginSeqNo	Y	Y	Int	In a Resend Request from the CBOE to the firm, this tag will indicate the beginning sequence number for the range of messages that are being requested.		
8	BeginString	Y	Y	String	Per standard. "FIX.4.2"	Identifies beginning of new message and protocol version. ALWAYS FIRST FIELD IN MESSAGE. (*Always unencrypted.*) Valid values: FIX.4.1	Standard Header of all messages
9	BodyLength	Y	Y	Int	Per standard.	Message length, in bytes, forward to the CheckSum field. ALWAYS SECOND FIELD IN MESSAGE. (*Always unencrypted.*) Valid values: 0 – 9999	Standard Header of all messages
10	CheckSum	Y	Y	String	Per standard.	Three byte, simple checksum (see Appendix B for description). ALWAYS LAST FIELD IN RECORD; i.e., serves, with the trailing <SOH>, as the end-of-record delimiter. Always defined as three characters. (*Always unencrypted.*)	Standard Trailer of all messages

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
11	ClOrdID	Y	Y	String	Unique identifier of the order as assigned by institution. This must contain the Branch + Branch Sequence Number + Order Date. Where: The Branch is 1 to 3 alphabetic only characters with no embedded blanks in the range: *A – ZZZ* (NOTE: For Market maker orders, CBOE suggests that the User ID (first half of SenderSubId) be used as the Branch). The Branch Sequence Number is a 1 to 4 numeric only value with no embedded blanks in the range: *1 – 9999* The order date is separated from the Branch + Branch Sequence Number by a hyphen"-". *AAA9999-YYYYMMDD* The order date must be provided as follows: *YYYYMMDD*	Unique identifier for Order as assigned by institution. Uniqueness must be guaranteed within a single trading day. Firms which electronically submit multi-day orders should consider embedding a date within the ClOrdID field to assure uniqueness across days.	Order, Cancel, Replace, Execution Report, Cancel Reject
14	CumQty	Y	Y	Int	In FIX 4.2 will be supported per the standard. For options this refers to the total number of contracts filled.	Total number of shares filled. Valid values: (0 – 1000000000)	Execution Report
15	Currency	N	N	Currency	Can be used to specify the currency that the price was quoted in.		
16	EndSeqNo	Y	Y	Int	In a Resend Request from the CBOE to the firm, this tag will indicate the end sequence number for the range of messages that are being requested. If this value = 0, then the firm should account for all messages from the beginning with no specific end number.		
17	ExecID	Y	Y	String	Unique serial number that is derived from the system time (GMT TIME?) in milliseconds when the Execution Report is created. CBOE Trade ID (High and Low). This value must be communicated to the CBOEdirect Administrator if the user wishes to request a trade bust.	Unique identifier of execution message as assigned by broker (will be 0 (zero) for ExecTransType=3 (Status)). Uniqueness must be guaranteed within a single trading day or the life of a multi-day order. Firms which accept multi-day orders should consider embedding a date within the ExecID field to assure uniqueness across days.	Execution Report
18	ExecInst	N	N	Multiple Value String	Refer to Contingency Mapping Table. The following values are supported in FIX 4.2 and are mapped to ORS contingencies: 1 = Not Held	Instructions for order handling on exchange trading floor. If more than one instruction is applicable to an order, this field can contain multiple instructions separated by space.	Order, Cancel, Replace, Execution Report

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
					G = All or None (AON) f = Intermarket sweep order (ISO) B = (OK to Cross) Midpoint Cross = Market + OK to Cross B = (OK to Cross) Cross=Limit+OK to Cross g=Tied Cross h=AutoLink Cross i=AutoLink Cross Match j=CROSS_WITHIN k=TIED_CROSS_WITHIN		
19	ExecRefID	N	N	String	Required for Cancel and Correct ExecTransType messages		
20	ExecTransType	Y	Y	Char	0 = New 1 = Trade Bust	Identifies transaction type. Valid values: 0 = New , 1 = Cancel, 2 = Correct, 3 = Status	Execution Report
21	HandlInst	Y	Y	Char	Must be specified per specification. Ignored by FIX 4.2. Any value is accepted. ORS decides order handling based upon order characteristics and firm specified configuration information.	Instructions for order handling on Broker trading floor. Valid values: 1 = Automated execution order, private, no Broker intervention 2 = Automated execution order, public, Broker intervention OK 3 = Manual order, best execution	Order, Cancel, Replace
22	IDSource	N	N	String	Ignored by CBOEdirect FIX 4.2 Service on messages received from firms. Field will be set to "8" (Exchange Symbol) on messages sent to firms.	Identifies class of alternative SecurityID. Valid values: 1 = CUSIP 2 = SEDOL 3 = QUIK 4 = ISIN number 5 = RIC code 6 = ISO Currency Code 7 = ISO Country Code 8 = Exchange Symbol 100+ are reserved for private security identifications	Order, Cancel, Cancel-Replace, Execution Report, Security Status.
30	LastMkt	Y	Y	Exchange	Market of execution for last fill.		
31	LastPx	Y	Y	Float	Last Sale Price. Will be set to 0.0 if Execution Report is not reporting a Fill or Partial Fill.	Price of this (last) fill. Field not required for ExecTransType = 3 (Status). Valid values: 0 - 99999999.9999 (number of decimal places may vary and not limited to four)	Execution Report

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
32	LastShares	Y	Y	Int	Last Sale Quantity. Will be set to 0 if Execution Report is not reporting a Fill or Partial Fill.	Quantity of shares bought/sold on this (last) fill. Field not required for ExecTransType = 3 (Status). Valid values: (0 - 1000000000)	Execution Report
33	LinesOfText			Int		Identifies number of lines of text body	
34	MsgSeqNum	Y	Y	Int	Per standard.	Integer message sequence number. Valid values: 0 - 999999	Standard Header of all messages
35	MsgType	Y	Y	String	Per standard.	Defines message type. ALWAYS THIRD FIELD IN MESSAGE. (*Always unencrypted.*)	Standard Header of all messages
36	NewSeqNo	N	Y	Int	Used on Resend Request Gap Fill messages		
37	OrderID	N	N	String	Will be set to the OrderID from CBOE which is of the form highCBOEId:lowCBOEid.	Unique identifier for Order as assigned by broker. Uniqueness must be guaranteed within a single trading day. Firms which accept multi-day orders should consider embedding a date within the OrderID field to assure uniqueness across days.	Execution Report
38	OrderQty	N	N	Int	Quantity specified on the original request message from the firm.	Number of shares ordered. Valid values: (0 - 1000000000)	Order, Cancel, Replace, Execution Report

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
39	OrdStatus	Y	Y	Char	Status of the order. FIX 4.2 supports the following values for OrdStatus: 0 New 1 Partial_Fill 2 Fill 3 Done_For_Day 4 Canceled 5 Replace 6 Pending_Cancel 7 Stopped 8 Rejected 9 Suspended A Pending_New B Calculated C Expired D Restated E Pending_Replace	Identifies current status of order. Valid values: 0 = New 1 = Partially filled 2 = Filled 3 = Done for day 4 = Canceled 5 = Replaced 6 = Pending Cancel/Replace 7 = Stopped 8 = Rejected 9 = Suspended A = Pending New B = Calculated C = Expired	Execution Report
40	OrdType	Y	Y	Char	FIX 4.2 supports the following order types: 1 = Market 2 = Limit 3 = Stop 4 = Stop Limit 5 = Market on close 7 = Limit or Better B = Limit on Close J = Market if Touched[4] Intermarket Sweep = 2 (Limit) Reserve = Used for Stock: Intermarket Sweep = 2 (Limit) Reserve = Midpoint Cross = 1 (Market) Cross = 2 (Limit) Tied Cross (currently unsupported) = 2 (Limit) Autolink Cross = 2 (Limit) Autolink Cross Match = 2 (Limit) Refer to Contingency Mapping Table.	Order type. Valid values: 1 = Market 2 = Limit 3 = Stop 4 = Stop limit 5 = Market on close 6 = With or without 7 = Limit or better 8 = Limit with or without 9 = On basis A = On close B = Limit on close C =Forex - Market D = Previously quoted E = Previously indicated F = Forex - Limit G = Forex - Swap H = Forex - Previously Quoted P = Pegged (requires ExecInst = L, R, M, P or O)	Order, Cancel, Replace, Execution Report
41	OrigClOrdID	N	Y	String	Per Standard – must follow format specified for field *ClOrdID[11]*.	ClOrdID of the previous order (NOT the intial order of the day) as assigned by the institution, used to identify the previous order in cancel and cancel replace requests.	Cancel, Replace, Execution Report

[4] OrdType of "J" designating Market if Touched order type is pending approval by the FIX Technical Committee as of this writing.

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
42	OrigTime			UTCTimestamp		Time of message origination (always expressed in UTC (Universal Time Coordinated, also known as "GMT"))	
43	PossDupFlag	N	N	Char	Per standard.	Indicates possible retransmission of message with this sequence number. Valid values: Y = Possible duplicate, N = Original transmission	Standard Header of all messages
44	Price	N	N	Float	Price per contract. Refer to Order Contingency Mapping Table. Required for limit OrdTypes. OrdType = 2 if no other contingency is specified. OrdType = 1 if no other contingency is specified and the Price field is not specified. 0.0 – 9999999.9999	Price per share. Valid values: 0 – 99999999.9999 (number of decimal places may vary and not limited to four)	Order, Replace, Execution Report

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
47	Rule80A (aka OrderCapacity)	N	N	Char	See the section in document FIX-03C entitled *"Specify the order capacity (origin) on an order"* for more details. ***CBOE Member Origins*** **C** Customer **F** Firm **M** Market maker **B** Broker Dealer **X** Customer Broker Dealer **N** Non member market maker / Market maker specialist at another options exchange (Not yet supported on the CBOE floor approved by EFPC) **Y** Stock Specialist registered in the underlying security (Not yet supported on the floor - approval status is unknown) ***CME Member Origins*** **V** CTI 1/Origin 1 Member, Customer Segregated Account **E** CTI 1/Origin 2 Member, House Account **Q** CTI 1/Origin 5 Member, SIPC Protected Account **G** CTI 3/Origin 1 User Proxy for trader, Customer Segregated Account **H** CTI 3/Origin 2 User Proxy for trader, House Account **R** CTI 3/Origin 5 User Proxy for trader, SIPC Protected Account **O** CTI 4/Origin 2 Non Member, House Account **T** CTI 4/Origin 5 Non Member, SIPC Protected Account ***CBSX Member Origins*** **D** Program Trade, index arb, for Member firm/org for its own account	Note that the name of this field is changing to "OrderCapacity" as Rule80A is a very US market-specific term. Other world markets need to convey similar information, however, often a subset of the US values. . See the ***"Rule80A (aka OrderCapacity) Usage by Market" appendix*** for market-specific usage of this field.Valid values: A = Agency single order B = Short exempt transaction (refer to A type) C = Program Order, non-index arb, for Member firm/org D = Program Order, index arb, for Member firm/org E = Registered Equity Market Maker trades F = Short exempt transaction (refer to W type) H = Short exempt transaction (refer to I type) I = Individual Investor, single order J = Program Order, index arb, for individual customer K = Program Order, non-index arb, for individual customer L = Short exempt transaction for member competing market-maker affiliated with the firm clearing the trade (refer to P and O types) M = Program Order, index arb, for other member N = Program Order, non-index arb, for other member O = Competing dealer trades P = Principal R = Competing dealer trades S = Specialist trades T = Competing dealer trades U = Program Order, index arb, for other agency W = All other orders as agent for other member X = Short exempt transaction for member competing market-maker not affiliated with the firm clearing the trade (refer to W and T types) Y = Program Order, non-index arb, for other agency Z = Short exempt transaction for non-member competing market-maker (refer to A and R types)	Order, Replace, Execution Report

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
					L Program Trade, non-index arb, for Member firm/org for its own account **I or M** Competing market maker and competing market maker-short exempt, for member/member org for its own account. Origin I is required to trade against odd-lots. Origin M cannot trade against odd-lots. **E** Short exempt[2] and all other orders for member/member org for its own account **H** Error (trader uses this trade to reverse out a previous trading error), for member/member org for its own account **F** Program Trade, index arb, for Member firm/org for another member's account **N** Program Trade, non-index arb, for Member firm/org for another member's account **T** Competing market maker and competing market maker short-exempt, for Member firm/org for another member's account **W** Short exempt (see footnote "2") and all other orders, for Member firm/org for another member's account **J** Program Trade, index arb, for individual as defined by Section 11(a) (1) (E) of the Securities Exchange Act of 1934 **K** Program Trade, non-index arb, for individual as defined by Section 11(a) (1) (E) of the Securities Exchange Act of 1934		

[2] For CBSX, the same account type code is used for short-exempt orders as is used for "all other orders". Short exempt orders must be separately marked as such when submitted to the Exchange.

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
					C Short exempt (see footnote "2") and all other orders, for individual as defined by Section 11(a) (1) (E) of the Securities Exchange Act of 1934 **U** Program Trade, index arb, for non-member/non-member org that is not an individual as defined by Section 11(a) (1) (E) of the Securities Exchange Act of 1934 **Y** Program Trade, non-index arb, for non-member/non-member org that is not an individual as defined by Section 11(a) (1) (E) of the Securities Exchange Act of 1934 **R** Competing market maker and competing market maker short-exempt, for non-member/non-member org that is not an individual as defined by Section 11(a) (1) (E) of the Securities Exchange Act of 1934 **B** Short exempt (see footnote "2") and all other orders, for non-member/non-member org that is not an individual as defined by Section 11(a) (1) (E) of the Securities Exchange Act of 1934		
48	SecurityID			String	The CBOEdirect product key is used to identify all products traded on CBOE and ONEChicago markets and their underlying products.	CUSIP or other alternate security identifier	
49	SenderCompID	Y	Y	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify firm sending message.	Standard Header of all messages
50	SenderSubID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify specific message originator (desk, trader, etc.).	Standard Header of all messages
52	SendingTime	Y	Y	time	Per standard. Time upon input is expected to be in Universal Time Coordinated (UTC) – see FIX specification. Time will be mapped to Central Time Zone when message is input to ORS. Outbound messages are mapped from Central Time Zone to UTC.	Time of message transmission (always expressed in GMT).	Standard Header of all messages

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
54	Side	Y	Y	Char	Side for the product that was specified on the request message from the firm. For Options or Futures orders, CBOE supports the following values: 1 = Buy 2 = Sell For orders of type MLEG (strategies), CBOE supports the following values: 1 = Same ***or*** B = Same 2 = Opposite ***or*** C=Opposite	Side of order. Valid values: 1 = Buy 2 = Sell 3 = Buy minus 4 = Sell plus 5 = Sell short 6 = Sell short exempt 7 = Undisclosed (valid for IOI message only) 8 = Cross (orders where counterparty is an exchange, valid for all messages except IOIs)	Order, Cancel, Replace, Execution Report
55	Symbol	Y	Y	Char	Used for the trading symbol for the product. For **options** this is the one to three character ticker symbol (referred to within CBOEdirect as the Reporting Class), examples: IBM, IBZ, MSQ, DLQ, OEX, OEY, DJX. For **futures** this is a one to four character ticker symbol, combined with a contract specification designation between 0 and 9 inclusive, followed by one character that represents OneChicago, for example: MSFT1C, IBM1C. For **underlying** it is the ticker symbol for the underlying product, such as IBM, T, MSFT, OEX, SPX.	Ticker symbol.	Order, Cancel, Replace, Execution Report
56	TargetCompID	Y	Y	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify receiving firm.	Standard Header of all messages
57	TargetSubID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify specific individual or unit intended to receive message. "ADMIN" reserved for administrative messages not intended for a specific user.	Standard Header of all messages
58	Text	N	N	Char	A text string that will vary in functionality depending on the type of message. CBOE ignores this field on new order messages sent to CBOE.	Free format text string. (Note: this field does not have a specified maximum length.)	Order, Replace, Execution Report, Reject, Logout, Email., Security Definition

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
59	TimeInForce	N	N	Char	Absence of this field indicates Day order. FIX 4.2 supports the following values for TimeInForce: 0 = Day 1 = Good Till Cancel (GTC) 2 = At the Opening (OPG) – not currently supported 3 = Immediate or Cancel (IOC) 4 = Fill or Kill (FOK) Refer to Contingency Mapping Table. **Options and Futures**: Specifies how long the order remains in effect. Absence of this tag is interpreted as DAY. Valid values: 0 = Day 1 = GTC 2 = OPG (at the open) 3 = IOC (Immediate or CXL) 4 = FOK (Fill or Kill)	Specifies how long the order remains in effect. Absence of this field is interpreted as DAY. Valid values: 0 = Day 1 = Good Till Cancel (GTC) 2 = At the Opening (OPG)-not currently supported 3 = Immediate or Cancel (OC) 4 = Fill or Kill (FOK) 5 = Good Till Crossing (GTX) 6 = Good Till Date (GTD)	Order, Replace
60	TransactTime	N	N	Char	GMT time the execution report was created.	Time of execution/order creation (expressed in GMT).	Execution Report
66	ListID	Y	N		Must be unique, by customer, for the day		
67	ListSeqNo	Y	Y		Order number within the list.		
68	TotNoOrders	Y	Y		Used to support fragmentation. Sum of NoOrders across all messages with the same ListID. Used for internalization Value=2		
73	NoOrders	Y	Y		Number of orders in the list message which is used for internalization orders Value=2		
76	ExecBroker	N	Y	Char	Firm Number – as assigned by the OCC (numeric – 3 digits)	Identifies executing / give-up broker. Standard NASD market-maker mnemonic is preferred.	Order, Cancel, Replace, Execution Report
77	OpenClose	N	N	Char	Required for customer orders by FIX 4.2.	For options only. Valid Values: O=Open, C=Close	Order, Cancel, Replace, Execution Report
84	CxlQty	N	N	Qty	Cancel or bust quantity. Can be aggregate depending on the message.	Total number of shares canceled for this order. (Prior to FIX 4.2 this field was of type int)	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
90	SecureDataLen	N	N	Int	Required to identify length of encrypted section of message. (Always unencrypted.) CBOE does not support encryption in FIX 4.2 at this time.	Length of encrypted message.	Standard Header of all messages
91	SecureData	N	N	Data	Required when message body is encrypted. Always immediately follows SecureDataLen field. CBOE does not support encryption in FIX 4.2 at this time.	Actual encrypted data stream.	Standard Header of all messages
94	EmailType			Char		Email message type. Valid values: 0 = New 1 = Reply 2 = Admin Reply	
97	PossResend	N	N	Boolean	Per standard.	Indicates that message may contain information that has been sent under another sequence number. Valid Values: Y=Possible resend; N=Original transmission	Standard Header of all messages
99	StopPx	N		Price	Required for OrdType = "Stop" or OrdType = "Stop limit." Refer to Contingency Mapping Table.	Price per share. Valid values: 0- 99999999.9999 (number of decimal places may vary and not limited to four)	Order, Replace, Execution Report
100	ExDestination	N	Y	Char	Should be set to "W" For W_STOCK, set to "CBOE"	Execution destination as defined by institution when order is entered. Valid values: See Appendix C in the FIX 4.2 Specification	Order, Replace, Execution Report
102	CxlRejReason	N	N	Char	0 = Too Late to Cancel 1 = Unknown order	Code to identify reason for cancel rejection. Valid values: 0 = Too late to cancel; 1 = Unknown order	Cancel Reject
103	OrdRejReason	N	N	Int	Per Standard	Code to identify reason for order rejection: Valid values: 0 – Broker or Exchange Option 1 – Unknown Symbol 2 – Exchange Closed 3 – Order Exceed Limit 4 – Too late to enter 5 – Unknown Order 6 – Duplicate order (e.g. dupe ClOrdID) 7 – Duplicate of verbally communicated order 8 – Stale Order	Execution Report
106	Issuer	N	N	String	Per FIX 4.2 Standard	Company name of security issuer (e.g. International Business Machines)	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
107	SecurityDesc			String	**Future, Option, Index, or Underlying Products**: Contains the CFICode for the underlying product **Strategy Products**: Strategy product names. The definitions of each strategy type are described in the CBOEdirect W_MAIN / ONE Strategy Order Test Plan. Straddle Pseudo Straddle Vertical Ratio Time Diagonal Combo Unknown Spread Type	Security description.	
108	HeartBtInt		Y	Int	The heart beat interval is set by the initiator at logon with this tag. CBOEdirect sets the heartbeat interval to be whatever the firm specifies.	Heartbeat interval (seconds)	
109	ClientID	N	Y	Char	Correspondent Firm that was specified on the request message (New Order, Cancel, Replace) from the firm.	Firm identifier used in third party-transactions.	Order, Cancel, Replace, Execution Report, Cancel Reject
110	MinQty	N	N	Int	Refer to Contingency Mapping Table. Used to specify a minimum quantity that can be filled on the order.	Minimum quantity of an order to be executed. Valid values: (0 - 1000000000)	Order, Cancel, Replace, Execution Report
115	OnBehalfOfCompID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify firm originating message if the message was delivered by a third party, i.e., the third party firm identifier would be delivered in the SenderCompID field and the firm originating the message in this field.	Standard Header of all messages
116	OnBehalfOfSubID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify specific message originator (i.e., trader) if the message was delivered by a third party.	Standard Header of all messages
117	QuoteID	Y	Y	String	Users should provide a quote identifier QuoteID[117] on each quote. A firm can have only one two sided quote in effect for a product at a time. Because of this, QuoteIDs do not have to be unique. You have the choice to keep the same QuoteID across multiple submissions of quotes for a specific product.	Unique identifier for quote	
122	OrigSendingTime	N	N	Time	Per standard.	Original time of message transmission (always expressed in GMT) when transmitting orders as the result of a resend request.	Standard Header of all messages

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
123	GapFillFlag	N	Y	Boolean	When a firm sends a Sequence Reset message, this tag should be set to = Y.	Indicates that the Sequence Reset message is replacing administrative or application messages which will not be resent. Valid values: Y = Gap Fill message, MsgSeqNum field valid N = Sequence Reset, ignore MsgSeqNum	
128	DeliverToCompID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify the firm targeted to receive the message if the message is delivered by a third party, i.e., the third party firm identifier would be delivered in the TargetCompID field and the ultimate receiver firm ID in this field.	Standard Header of all messages
129	DeliverToSubID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify specific message recipient (i.e., trader) if the message is delivered by a third party	Standard Header of all messages
131	QuoteReqID		Y	String		Unique identifier for quote request	
141	ResetSeqNumFlag	N	N	Boolean	CBOE does not support this tag.		
142	SenderLocationID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify specific message originator's location (i.e., geographic location and/or desk, trader).	Standard Header of all messages
143	TargetLocationID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify specific message originator's location (i.e., geographic location and/or desk, trader).	Standard Header of all messages
144	OnBehalfOfLocationID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify specific message originator's location (i.e., geographic location and/or desk, trader) if the message was delivered by a third party.	Standard Header of all messages
145	DeliverToLocationID	N	N	Char	Refer to FIX-03A for details on CBOE usage	Assigned value used to identify specific message originator's location (i.e., geographic location and/or desk, trader) if the message was delivered by a third party.	Standard Header of all messages
146	NoRelatedSym	Y	Y	Int	Number of securities returned as part of the repeating group.	Specifies the number of repeating symbols specified.	
147	Subject			String		The subject of an Email message	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
150	ExecType	Y	Y	Char	**Orders**: Same values as OrdStatus[39] **Quotes**: Same values as Quotestatus[9312]	Describes the specific ExecutionRpt (i.e., Pending Cancel) while OrdStatus will always identify the current order status (i.e., Partially Filled). Valid values: 0 = New 1 = Partial fill 2 = Fill 3 = Done for day 4 = Canceled 5 = Replace 6 = Pending Cancel/Replace 7 = Stopped 8 = Rejected 9 = Suspended A = Pending New B = Calculated C = Expired D = Restarted E = Pending_Replace	Execution Report
151	LeavesQty	Y	Y	Int	Quantity open for further execution.	Amount of shares open for further execution. If the OrdStatus is Canceled, DoneForTheDay, Expired, Calculated, or Rejected (in which case the order is no longer active) then LeavesQty could be 0, otherwise LeavesQty = OrderQty - CumQty. Valid values:(0 - 1000000000)	Execution Report
164	EmailThreadID			String		Unique identifier for an email thread (new and chain of replies)	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
167	SecurityType	N	Y	Char	**OPT** = Stock or Index options. **CS** = Type used to identify all stock products within CBOEdirect. The CBOEdirect system does not differentiate the type of stock (equity). For this reason, all stocks that serve as underlyings with CBOEdirect will be designated as Fix SecurityType[167] = "CS". **FUT** = Futures, including single stock (securities) futures. **INDX** = Underlying Index for an Index Option product, such as OEX, SPX, NDX, VIX. **MLEG** = Multi-leg security, such as an option strategy or a combination of underlying and derivative (buy write for example). **USTB** = The CBOEdirect system does not differentiate the type of debt For this reason, all debt products that serve as underlyings with CBOEdirect will be designated as Fix SecurityType[167] = "USTB" (eg. US Treasury underlying the Interest Rate products)	Indicates type of security (ISITC spec). Valid values: see FIX 4.1 Specification	Order, Cancel, Replace, Execution Report, Cancel Reject
198	SecondaryOrderID	N	N	Char	Will be set to the CBOE ORS system order Id– if it is available- for orders that are routed to the CBOE trading floor. Will be provided on fill reports, cancel reports, and nothing done reports.	Assigned by the party which accepts the order. Can be used to provide the OrderID used by an exchange or executing system.	Execution Report
200	MaturityMonthYear	N	Y	Month-year	Must be set to valid CBOEdirect Expiration Year and Month. Required if SecurityID[tag 48] is not used.	Month and Year of the maturity for SecurityType=FUT or SecurityType=OPT. Format: YYYYMM (e.g., 200209)	Order, Cancel, Replace, Execution Report, Cancel Reject
201	PutOrCall	N	Y	Int		Indicates whether an Option is for a put or call. Valid values: 0 = Put, 1 = Call	Order, Cancel, Replace, Execution Report, Cancel Reject
202	StrikePrice	N	Y	Float		Strike Price for an Option. Valid values: 0 - 99999999.9999 (number of decimal places may vary and not limited to four)	Order, Cancel, Replace, Execution Report, Cancel Reject
203	CoveredOrUncovered	N	N	Char	CBOE users: Required for Customer Sell orders only, optional for all other orders.	Used for options. Valid values: 0 = Covered, 1 = Uncovered	Order, Replace, Execution Report
204	CustomerOrFirm	N	N	Char	CBOE has the following expanded values for this field: 0 = Customer 1 = Firm 2 = Broker Dealer (CBOE only) 3 = Customer Broker Dealer (CBOE only) 4 = Market Maker (CBOE only)	Used for options when delivering the order to an execution system/exchange to specify if the order is for a customer or the firm placing the order itself. Valid values: 0 = Customer,1 = Firm	Order, Replace, Execution Report

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
205	MaturityDay	N	Y	Day of the month	Required by CBOE. Used to specify the actual expiration date. The exact MaturityDay is provided on all messages sent from CBOE back to the firm.	Day of month used in conjunction with MaturityMonthYear to specify the maturity date for SecurityType=FUT or SecurityType=OPT. Valid values: 1-31	
207	SecurityExchange	N	N	Char	If used, should be set to "W" for CBOE or "50" for OneChicago. Ignored on input. W = CBOE C2OX= CBOE 2 (C2) 50 = OneChicago	Market used to help identify a security. Valid values: See Appendix C of FIX 4.1 Specification.	Order, Cancel, Replace, Execution Report
262	MDReqID	Y	Y	String	Will be used to track subscriptions – must be unique per request.	Unique identifier for Market Data Request	
263	SubscriptionRequestType	N	N	Char	SubcriptionRequestType indicates to the other party what type of response is expected. A snapshot request ("0") only asks for current information. A subscribe request ("1") asks for updates as the status changes. Unsubscribe ("2") will cancel a subscription. ("3") subscribe to auction. ("4") unsubscribe from auction.	Subscription Request Type Valid Values: 0 – Snapshot 1 – Snapshot + Update (Subscribe) 2 – Disable previous Snapshot+Update Request (Unsubscribe)	Security Status Request, Trading Session Status Request, Market Data Request, Quote Status Request (Non Standard)
264	MarketDepth	N	N	int	Per Standard	Depth of market for book snapshot Valid values: 0 – Full book 1 – Top of book N>1 report best N price tiers of data	
265	MDUpdateType	N	N	int	CBOE only supports 0-Full Refresh. Any other enumeration will be ignored.	Specifies the type of Market Data update. Valid values: 0 = Full Refresh 1 = Incremental Refresh	
266	AggregatedBook	N	N	Boolean	At this time CBOE does not provide aggregated book – multiple entries per side per price will be provided when available.	Specifies whether or not book entries should be aggregated. Valid values: Y = one book entry per side per price N = Multiple entries per side per price allowed (Not specified) = broker option	
267	NoMDEntryTypes	Y	Y	Int	Specifies the number of market data entries to be provided.	Number of MDEntryType fields requested.	
268	NoMDEntries	Y	Y	Int	Number of entries following in the repeating group for a Market Data message.	Number of entries in Market Data message.	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
269	MDEntryType	Y	Y	Char	Must be the first field in this repeating group. CBOE supports the following MDEntryTypes at this time: 0-Bid 1-Offer 2-Trade 3-Index Value 4-Opening Price 5-Closing Price 7-Session High Price 8-Session Low Price	Type Market Data entry. Valid values: 0 = Bid 1 = Offer 2 = Trade 3 = Index Value 4 = Opening Price 5 = Closing Price 6 = Settlement Price 7 = Trading Session High Price 8 = Trading Session Low Price 9 = Trading Session VWAP Price	
270	MDEntryPx	Y	Y	Price		Price of the Market Data Entry.	
271	MDEntrySize	N	N	Qty	Conditionally required if MDEntryType = Bid(0), Offer(1), or Trade(2)	Number of shares represented by the Market Data Entry.	
272	MDEntryDate	N	N	UTCDate		Date of Market Data Entry.	
273	MDEntryTime	N	N	UTCTime Only		Time of Market Data Entry.	
274	TickDirection	N	N	Char		Direction of the "tick". Valid values: 0 = Plus Tick 1 = Zero-Plus Tick 2 = Minus Tick 3 = Zero-Minus Tick	
275	MDMkt	N	N	Exchange	"W". Market posting quote / trade. Valid values: See Appendix C of the FIX specification	Market posting quote / trade. Valid values: ***See Appendix C** of the FIX specification*	
276	QuoteCondition	N	N	MultipleValueString	Space-delimited list of conditions describing a quote.	Space-delimited list of conditions describing a quote. Valid values: A = Open / Active B = Closed / Inactive C = Exchange Best D = Consolidated Best E = Locked F = Crossed G = Depth H = Fast Trading I = Non-Firm	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
277	TradeCondition	N	N	MultipleValueString	Space-delimited list of conditions describing a trade	Space-delimited list of conditions describing a trade Valid values: A = Cash (only) Market B = Average Price Trade C = Cash Trade (same day clearing) D = Next Day (only) Market E = Opening / Reopening Trade Detail F = Intraday Trade Detail G = Rule 127 Trade (NYSE) H = Rule 155 Trade (Amex) I = Sold Last (late reporting) J = Next Day Trade (next day clearing) K = Opened (late report of opened trade) L = Seller M = Sold (out of sequence) N = Stopped Stock (guarantee of price but does not execute the order)	
281	MDReqRejReason			Char	This will state the reason for the reject of the market data request.	Reason for the rejection of a Market Data request. Valid values: 0 = Unknown symbol 1 = Duplicate MDReqID 2 = Insufficient Bandwidth 3 = Insufficient Permissions 4 = Unsupported SubscriptionRequestType 5 = Unsupported MarketDepth 6 = Unsupported MDUpdateType 7 = Unsupported AggregatedBook 8 = Unsupported MDEntryType	
282	MDEntryOriginator	N	N	String	Originator of a Market Data Entry	Originator of a Market Data	
283	LocationID	N	N	String		Identification of a Market Maker's location	
284	DeskID	N	N	String		Identification of a Market Maker's desk	
286	OpenCloseSettleFlag	N	N	Char	Used if MDEntryType = Opening Price(4) or Closing Price(5). Settlement Price(6) is not supported by CBOEdirect or OneChicago and can be obtained from the Options Clearing Corporation (OCC) or CBOE Financial Network (CFN).	Flag that identifies a price. Valid values: 0 = Daily Open / Close / Settlement price 1 = Session Open / Close / Settlement price 2 = Delivery Settlement price	
287	SellerDays	N	N	Int		Specifies the number of days that may elapse before delivery of the security	
288	MDEntryBuyer	N	N	String	For optional use in reporting Trades	Buying party in a trade	
289	MDEntrySeller	N	N	String	For optional use in reporting Trades	Selling party in a trade	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
290	MDEntryPositionNo	N	N	Int	Display position of a bid or offer, numbered from most competitive to least competitive, per market side, beginning with 1	Display position of a bid or offer, numbered from most competitive to least competitive, per market side, beginning with 1.	
291	FinancialStatus			Char		Identifies a firm's financial status. Valid values: 1 = Bankrupt	
292	CorporationAction			Char		Identifies the type of Corporate Action. Valid values: A = Ex-Dividend B = Ex-Distribution C = Ex-Rights D = New E = Ex-Interest	
295	NoQuoteEntries	Y	Y	Int	The number of quotes for this Symbol (QuoteSet) that follow in this message.		
296	NoOutofQuoteSets		Y				
298	QuoteCancelType	Y	Y	Int	Dictates the type of quote cancellation desired 1 = Cancel for Symbol (cancel specific product) 3 = Cancel for Underlying Symbol (cancel class) 4 = Cancel All Quotes NOTE: CBOE does not permit cancellation by Security Type QuoteCancelType[298]=2.	Identifies the type of quote cancel. Valid Values: 1 – Cancel for Symbol(s) 2 – Cancel for Security Type(s) 3 – Cancel for Underlying Symbol 4 – Cancel All Quotes	
299	QuoteEntryID	N	N	String	For optional use when this Bid, Offer, or Trade represents a quote. Uniquely identifies the quote as part of a QuoteSet.	Uniquely identifies the quote as part of a QuoteSet.	
301	QuoteResponseLevel	N	N	int	Applications can optionally support acknowledgement of quotes using this tag. This tag is used to specify the level of acknowledgement requested from the counterparty. 0 = no acknowledgement is requested 1 = requests acknowledgement of invalid or erroneous quotes 2 = requests acknowledgement of each Mass Quote message	Level of Response requested from receiver of quote messages. Valid Values: 0 – No Acknowledgement (Default) 1 – Acknowledge only negative or erroneous quotes 2 – Acknowledge each quote messages	
302	QuoteSetID	Y	Y	String	Sequential number for the Quote Set. For a given QuoteID – assumed to start at 1. Must be used if NoQuoteSets is used.	Unique ID for the Quote Set.	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
304	TotQuoteEntries	Y	Y	int	Total number of quotes for the quote set across all messages. Should be the sum of all NoQuoteEntries in each message that has repeating quotes that are part of the same quote set. Required if NoQuoteEntries > 0	Total number of quotes for the quote set across all messages. Should be the sum of all NoQuoteEntries in each message that has repeating quotes that are part of the same quote set.	
305	UnderlyingIDSource	N	N	String		Underlying security's IDSource. Valid values: see IDSource field	
306	UnderlyingIssuer			String		Underlying security's Issuer. See Issuer field for description	
307	UnderlyingSecurityDesc			String		Underlying security's SecurityDesc. See SecurityDesc field for description	
308	UnderlyingSecurityExchange			Exchange	The exchange on which the underlying security trades.	Underlying security's SecurityExchange. Can be used to identify the underlying security. Valid values: see SecurityExchange	
309	UnderlyingSecurityID			String		Underlying security's SecurityID. See SecurityID field for description	
310	UnderlyingSecurityType	N	Y	String		Underlying security's SecurityType. Valid values: see SecurityType field	
311	UnderlyingSymbol	Y	Y	String	The symbol of the underlying product. Examples: IBM, MSFT, T, OEX, DJX	Underlying security's Symbol. See Symbol field for description	
312	UnderlyingSymbolSfx			String		Underlying security's SymbolSfx. See SymbolSfx field for description	
313	UnderlyingMaturityMonthYear		Y	month-year	This field would only be used for debt underlyings.	Underlying security's MaturityMonthYear. See MaturityMonthYear field for description	
314	UnderlyingMaturityDay	N	Y	day-of-month	This field would only be used for debt underlyings.	Underlying security's MaturityDay. See MaturityDay field for description	
315	UnderlyingPutOrCall		Y	Int		Underlying security's PutOrCall. See PutOrCall field for description	
316	UnderlyingStrikePrice		Y	Price		Underlying security's StrikePrice. See StrikePrice field for description	
318	UnderlyingCurrency			Currency		Underlying security's Currency. See Currency field for description and valid values	
319	RatioQty			float	Quantity of particular leg in the Security	Quantity of a particular leg in the security. Note: the RatioQty values of all the legs for a strategy product are reduced to their lowest common denominator before being used as a multiplier.	
320	SecurityReqID			String		Unique ID of a Security Definition Request.	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
321	SecurityRequestType			Int		Type of Security Definition Request. Valid values: 0 = Request Security identity and specifications 1 = Request Security identity for the specifications provided (Name of the security is not supplied) 2 = Request List Security Types 3 = Request List Securities (Can be qualified with Symbol, SecurityType, TradingSessionID, SecurityExchange is provided then only list Securities for the specific type)	
322	SecurityResponseID			Int	Identifier for the Security Definition message	Unique ID of a Security Definition message.	
323	SecurityResponseType			Int		Type of Security Definition message response. Valid values: 1 = Accept security proposal as is 2 = Accept security proposal with revisions as indicated in the message 3 = List of security types returned per request 4 = List of securities returned per request 5 = Reject security proposal 6 = Can not match selection criteria	
324	SecurityStatusReqID			Int	Must be unique, or the ID of previous Security Status Request to disable if SubscriptionRequestType = Disable previous Snapshot + Updates Request (2).	Unique ID of a Security Status Request message.	
325	UnsolicitedIndicator			Boolean	'Y' if message is sent unsolicited as a result of a previous subscription request.	Indicates whether or not message is being sent as a result of a subscription request or not. Valid values: Y = Message is being sent unsolicited N = Message is being sent as a result of a prior request	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
326	SecurityTradingStatus	N	N	Int	The CBOEdirect system only supports the following security trading states: 2 = Halted 17 = Open 18 = Closed 21 = Pre-Open 22 = Opening Rotation 23 = Fast	Identifies the trading status applicable to the transaction. 1 = Opening Delay 2 = Trading Halt 3 = Resume 4 = No Open/No Resume 5 = Price Indicaiton 6 = Trading Range Indication 7 = Market Imbalance Buy 8 = Market Imbalance Sell 9 = Market On Close Imbalance Buy 10 = Market On Close Imbalance Sell 11 = (not assigned) 12 = No Market Imbalance 13 = No Market On Close Imbalance 14 = ITS Pre-Opening 15 = New Price Indication 16 = Trade Dissemination Time 17 = Ready To Trade (start of session) 18 = Not Available For Trading (end of session) 19 = Not Traded On This Market 20 = Unknown or Invalid	
327	HaltReason			Char	Denotes the reason for the Opening Delay or Trading Halt.	Denotes the reason for the Opening Delay or Trading Halt. Valid values: I = Order Imbalance X = Equipment Changeover P = News Pending D = News Dissemination E = Order Influx M = Additional Information	
328	InViewOfCommon			Boolean		Indicates whether or not the halt was due to Common Stock trading being halted. Valid values: Y = Halt was due to common stock being halted N = Halt was not related to a halt of the common stock	
329	DueToRelated			Boolean		Indicates whether or not the halt was due to the Related Security being halted. Valid values: Y = Halt was due to related security being halted N = Halt was not related to a halt of the related security	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
330	BuyVolume			Qty		Number of shares bought.	
331	SellVolume			Qty		Number of shares sold.	
332	HighPx			Price		Represents an indication of the high end of the price range for a security prior to the open or reopen	
333	LowPx			Price		Represents an indication of the low end of the price range for a security prior to the open or reopen	
334	Adjustment			Int		Identifies the type of adjustment. Valid values: 1 = Cancel 2 = Error 3 = Correction	
335	TradSesReqID			String	Must be unique, or the ID of previous Market Data Request to disable if SubscriptionRequestType = Disable previous Snapshot + Updates Request (2).	Unique ID of a Trading Session Status message.	
336	TradingSessionID	N	N	String	"**W_MAIN**" Primary trading session for CBOE products "**C2_MAIN**" fully electronic Options trading session "**W_STOCK**" CBSX trading session "**ONE_MAIN**" OneChicago futures market If the TradingSessionID is not specified then the default will be the primary trading session. "**Underlying**" This will be the session where market data will be available for all products that are underlying CBOEdirect traded products.	Identifier for Trading Session Can be used to represent a specific market trading session Values should be bi-laterally agreed to between counterparties.	
337	ContraTrader	N	N	String	Optional part of repeating group. Contrabroker information is only available for members of the CBOE for CBOE products traded on the trading floor (W_MAIN)	Identifies the trader (e.g. "badge number") of the ContraBroker.	Execution Report
340	TradSesStatus			Int		State of the trading session. Valid values: 1 = Halted 2 = Open 3 = Closed 4 = Pre-Open 5 = Pre-Close	
341	TradSesStartTime			UTCTimestamp		Time of the opening of the trading session	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
345	TradSesEndTime			UTCTimestamp		Closing time of the trading session	
346	NumberOfOrders	N	N	Int	In an Aggregated Book, used to show how many individual orders make up an MDEntry	Number of orders in the market.	
354	EncodedTextLen			Int		Byte length of encoded (non-ASCII characters) EncodedText field.	
355	EncodedText			Data		Encoded (non-ASCII characters) representation of the Text field in the encoded format specified via the MessageEncoding field. If used, the ASCII (English) representation should also be specified in the Text field.	
375	ContraBroker	N	N	String	First field in repeating group – required if NoContraBrokers > 0. Contrabroker information is only available for CBOE products traded on the trading floor (W_MAIN)	Identifies contra broker. Standard NASD market-maker mnemonic is preferred.	Execution Report
378	ExecRestatementReason		Y	Int	4 = Broker Option This indicates that it was the broker/markets option to reinstate the busted trade quantity to the order.		
380	BusinessRejectReason					Code to identify reason for a Business Message Reject message. Valid values: 0 = Other 1 = Unkown ID 2 = Unknown Security 3 = Unsupported Message Type 4 = Application not available 5 = Conditionally Required Field Missing	
382	NoContraBrokers	N	N	Int	A maximum of 8 contraparties can be reported by FIX 4.2. Contrabroker information is only available for CBOE products traded on the trading floor (W_MAIN)	The number of ContraBroker entries	Execution Report
386	NoTradingSessions	N	N	Int	Indicates the number of TradingSessionIDs specified. For FIX 4.2 a maximum of 1 trading session can be specified.	Number of TradingSessionIDs in repeating group.	
387	TotalVolumeTraded	N	N	Int	Total volume traded in this trading session for this security.	Total volume (quantity) traded.	

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
388	DiscretionInst	N	N	Char	Only value supported is 0 - (Related to displayed price). The DiscretionOffset will be related to the Price(44). Field should only be included when a discretionary price is to be specified. Refer to Contingency Mapping Table.	Code to identify the price a DiscretionOffset is related to and should be mathematically added to. Valid values: 0 = Related to displayed price 1 = Related to market price 2 = Related to primary price 3 = Related to local primary price 4 = Related to midpoint price 5 = Related to last trade price	Order, Replace
389	DiscretionOffset	N	N	PriceOffset	Values < 1.0 that can be added/subtracted from the limit order price for a with discretion order Refer to Contingency Mapping Table.	Amount (signed) added to the "related to" price specified via DiscretionInst.	Order, Replace
393	TotalNumSecurities	N	N	Int		Total number of securities	
394	BidType	Y	Y		Used for internalization orders using the New Order-List message, with the "disclosed" convention 2=Disclosed		
424	DayOrderQty			Qty		For GT (GTC and GTD) orders, the OrderQty less all shares (adjusted for stock splits) that traded on previous days DayOrderQty = OrderQty – (CumQty - DayCumQty)	
425	DayCumQty			Qty		The number of shares on a GT order that have traded today	
426	DayAvgPx			Price		The average price of shares on a GT order that have traded today.	
432	ExpireDate	N	N	LocalMktDate	For optional use when this Bid or Offer represents an order. ExpireDate and ExpireTime cannot both be specified in one Market Data Entry.	Date of order expiration (last day the order can trade), always expressed in terms of the local market date. The time at which the order expires is determined by the local market's business practices	
436	UnderlyingContractMultiplier	N	N	float	When a firm sends the CBOE a Security Definition Request to obtain all classes in a particular trading session, CBOE will publish a Security Definition message containing the Contract Size (shares per contract, or SPC) in the repeating group for each class. This value will usually = 100 unless corporate actions (stock splits, etc.) have caused it to change.		
437	ContraTradeQty	N	N	Qty	Optional part of repeating group. Contrabroker information is only available for CBOE products traded on the trading floor (W_MAIN)	Quantity traded with the ContraBroker.	Execution Report
438	ContraTradeTime	N	N	UTCTimestamp	Optional part of repeating group. Contrabroker information is only available for CBOE products traded on the trading floor (W_MAIN)	Identifes the time of the trade with the ContraBroker. (Always expressed in UTC (Universal Time Coordinated, also known as "GMT").)	Execution Report

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
439	ClearingFirm	N	N	String	CMTA OCC clearing firm number	Firm that will clear the trade. Used if different from the executing firm.	Order, Replace
440	ClearingAccount	N	N	String	Q Account of market maker sub account information sent to OCC	Supplemental accounting information forwarded to clearing house / firm	Order, Replace, Execution Report
442	MultilegReportingType	N	N	char	Per Standard	Used to indicate what an Execution Report represents (e.g. used with multi-leg securities, such as option strategies, spreads, etc.) Valid Values: 1- Single Leg Security (default\ 2 – Individual leg of a multi-leg security 3 – Multi-leg security	Execution Report
555	NoLegs	N	Y		Required for One-Step Strategy Order		Not part of the FIX 4.2 Standard, but are FIX 4.3 tags that CBOEdirect has adopted to implement the One-Step Strategy Order Feature
609	LegSecurityType	N	Y		Required for One-Step Strategy Order		Not part of the FIX 4.2 Standard, but are FIX 4.3 tags that CBOEdirect has adopted to implement the One-Step Strategy Order Feature
600	LegSymbol	N	Y		Required for One-Step Strategy Order		Not part of the FIX 4.2 Standard, but are FIX 4.3 tags that CBOEdirect has adopted to implement the One-Step Strategy Order Feature
610	LegMaturityMonthYear	N	Y		Required for One-Step Strategy Order		Not part of the FIX 4.2 Standard, but are FIX 4.3 tags that CBOEdirect has adopted to implement the One-Step Strategy Order Feature
611	LegMaturityDate	N	Y		Required for One-Step Strategy Order		Not part of the FIX 4.2 Standard, but are FIX 4.3 tags that CBOEdirect has adopted to implement the One-Step Strategy Order Feature
612	LegStrikePrice	N	Y		Required for One-Step Strategy Order		Not part of the FIX 4.2 Standard, but are FIX 4.3 tags that CBOEdirect has adopted to implement the One-Step Strategy Order Feature

Tag #	Tag Name	FIX Req'd	CBOE Req'd	Data Type	CBOE Usage	Standard Definition	FIX Messages
613	LegOptAttribute	N	Y		Required for One-Step Strategy Order Put or Call		Not part of the FIX 4.2 Standard, but are FIX 4.3 tags that CBOEdirect has adopted to implement the One-Step Strategy Order Feature
623	LegRatioQty	N	Y		Required for One-Step Strategy Order		Not part of the FIX 4.2 Standard, but are FIX 4.3 tags that CBOEdirect has adopted to implement the One-Step Strategy Order Feature
624	LegSide	N	Y		Required for One-Step Strategy Order		Not part of the FIX 4.2 Standard, but are FIX 4.3 tags that CBOEdirect has adopted to implement the One-Step Strategy Order Feature

Custom Defined Fields

CBOE wherever possible has worked with the FIX Protocol Group to adhere to the FIX protocol standard. Where CBOE requirements differed from the FIX specification, CBOE worked to have these deficiencies addressed in future versions of the FIX Protocol. CBOE registers their user defined fields. This table contains the user-defined tags that are part of the FIX 4.2 service.

Table 30 Custom Defined Tags used in the CBOE FIX 4.2 Service

Tag #	Tag Name	DataType	CBOE Usage	FIX Messages (required=Y, not required=N)	Standard Adoption Status
5941	DirectedFirm	string	Required for Directed AIM auctions		
6706	NoOfLegsList		Required for options, futures, the common stock leg of buy_writes		
6707	NestedPartyIdList		For MultiLegStockClearingFirm. Optional for all product types		
6708	LegMaturityDayList		Required for options, futures, the common stock leg of buy_writes		
6711	LegSecurityTypeList		Required for options, futures, the common stock leg of buy_writes		
6712	LegSymbolList		This is the reporting class of the leg (IBM, IBJ, IBW, MSFTIC, etc) For Buy writes it would be stock symbol. Required for options, futures, the common stock leg of buy_writes.		
6713	LegSecurityIdList		Optional for options, futures, the common stock leg of buy_writes		
6714	LegSideList		Required for options, futures, the common stock leg of buy_writes		
6715	LegRatioQtyList		Required for options, futures, the common stock leg of buy_writes		
6716	LegPriceList		Optional for options, futures, the common stock leg of buy_writes		
6717	LegMaturityMonthYearList		Required for options, futures, the common stock leg of buy_writes		
6718	LegStrikePriceList		Required for options		
6719	LegOptAttributeList		For LegPutOrCall. Required for options		
6720	LegCoveredUncoveredList		Optional for options and futures. N/A for the common stock leg of buy_writes.		
6721	LegPositionEffectList		Optional for options and futures. N/A for the common stock leg of buy_writes		
6722	LegRefIdList		Optional for options, futures, the common stock leg of buy_writes		
6818	BrokerRoutingID		Optional for options, futures, equity		

Tag #	Tag Name	DataType	CBOE Usage	FIX Messages (required=Y, not required=N)	Standard Adoption Status
9192	ConcurrentOrder/QuoteIndicator	String	Optional field. Used to select the usage of the concurrent threading model. Valid values are: 1=enable concurrent quote model 2=enable concurrent order model 3=enable both concurrent order and quote model		
9221	AuxAuctionInfo	String	This field may not always be present, and may contain multiple data values concerning the auction. Specifically, in the case of a COA auction, the NBBO Bid and NBBO Ask values at the time of the auction solicitation may be provided. For COA and other auction types; each of the executing firm, correspondent firm, and CMTA firm values of the order(s) being auctioned may be provided (these settings are configurable by firm). In order to present these new data values, AuxAucionInfo will be formatted as a comma-delimited series of name=value pairs, where the names will be: • NBBOBID • NBBOASK • FIRM • CORRESFIRM • CMTAFIRM The comma-delimited list will be surrounded by "{" "}" characters	Quote Request	
9302	OrderPrice		The starting price for the auction. This tag is part of the Quote Request message.	Quote Request	
9310	CancelOpenQty	String	Cancel/Replace message to specify the required cancel quantity on the light order cancel replace		
9312	QuoteStatus	String	Status of Quote – a subset of OrdStatus[39] Booked Busted Cancelled Filled Locked (W_MAIN only) New Open Outcry Possible Resend Query Reinstated Removed Updated	Quote	Adopted FIX 4.3

Tag #	Tag Name	DataType	CBOE Usage	FIX Messages (required=Y, not required=N)	Standard Adoption Status
9313	QuoteRequestSubscription	int	Presence of this tag on the Quote Status Request – indicates that user wants to subscribe for Quote Requests.	Quote Status Request	Adopted FIX 4.3 via the usage of the SubscriptionRequestType field on the Quote Status Request
9314	OpenInterest	Qty	OpenInterest is not supported by CBOEdirect or OneChicago and can be obtained from the Options Clearing Corporation (OCC) or CBOE Financial Network (CFN).	Market Data – Full Refresh	Adopted FIX 4.3 as OpenInterest[
9315	MDScope	int	Indicates scope of the market data: 1-Local 2-National 3-Global	Market Data Request, Market Data – Full Refresh	
9316	LegalMarket	boolean	Indicates if the market for a product is within exchange presribed widths	Market Data – Full Refresh for Current Market (Top of Book)	
9317	LightOrderIndicator	String	Valid value for Light Orders is “1”	N	
9318	OrderTypeLogonIndicator	String	Used to specify whether the user will send light orders, regular orders or both.	N	
9322	MultilegPriceIncrement	Price	Used to defined the price increment for generation of a multileg instrument. The price increment is used to indicate the increment to the price of the instrument defined in the security block for the next leg of the multileg security	c(n), d(n)	
9323	MultilegMonthIncrement	Price	Number of months to increment the next leg of a multileg instrument from an anchor leg. Used for option strategy definition.	c(n), d(n)	
9324	ClearingOptionalData	String	Contains optional data that will be passed on to trade match and will be part of clearing information sent to the OCC.	8(n) ,D(n), G(n)	
9321	SecondaryClOrdID	String	Used when counterparties require a secondary client order id. Will be replaced by FIX 4.3 field of the same name.	8(n), 9(n), D(n), F(n), G(n)	Added to FIX 4.3 as Tag 526)
9365	PremPriceTickBreakPoint	Price	Price at which the Premium Price Tick changes from the PremPriceTickBelow and the PremPriceTickAbove	c(n), d(n)	
9366	PremPriceTickAbove	Price	Premium Price Tick Size above the PremPriceTickBreakPoint	c(n), d(n)	
9367	PremPriceTickBelow	Price	Premium Price Tick Size Below the PremPriceTickBreakPoint	c(n), d(n)	
9368	LastBustShares	Qty	The number of shares reported as part of a trade bust	8(n)	

Tag #	Tag Name	DataType	CBOE Usage	FIX Messages (required=Y, not required=N)	Standard Adoption Status
9369	PriceProtectionScope	int	Defines the type of price protection the customer requires on their order Valid values: 0 = None 1 = Local (Exchange, ECN, ATS) 2 = National (Across all national markets) 3 = Global (Across all markets)	8(n), D(n), G(n)	Proposed for future version of FIX (post FIX 4.3)
9370	MultiLegPositionEffects	Multi Value String	Array of open close codes for multileg orders	8(n), D(n), G(n)	OBSOLETE with FIX 4.3 Multileg Order (MsgType=AB) LegPostionEffect (tag 564) field
9371	MultilegCoveredOrUncovered	Multi Value String	Field containing the CoveredUncovered constants for the legs of a multileg instrument. Added for FIX 4.2 complex order support. Same values as CoveredOrUncovered (tag 203) 0 – Covered, 1-Uncovered.	8(n), D(n), G(n)	OBSOLETE with FIX 4.3 Multileg Order (MsgType=AB) LegCoveredOrUncovered(tag 565) field
9372	MultilegStockClearingFirm	String	The Clearing firm for the stock leg of a multileg option strategy added for complex order support in FIX 4.2. Used to specify the stock clearing firm for the stock leg of a combination stock – derivative strategy, such as a buy-write where the option clearing firm number and stock clearing firm number are different.	8(n), D(n), G(n)	OBSOLETE with FIX 4.3 with the Multileg Order (MsgType=AB) Nested Parties block clearing firm party role
9379	MultilegPricePerLeg	String	Used for submitting “Delta-Neutral” Orders. This field is used to specify individual leg prices as a comma delimited list of prices. It is important to know whether the strategy was created with the legs “As Defined” or “Opposite.” The leg prices will be applied to the systems leg’s order descriptions, irrespective of the order in the Security Definition Request		
9380	StockFirmName	String	Buy_writes: used to specify where the equity leg of the buy_write strategy is to be executed. This field is recommended for buy_writes, but not required. Used to specify the equity clearing firm number for cross-product buy_writes when the stock clearing firm number is different from the option clearing firm number.		
9381	StockFirmNameKey	String	A text field that is used to specify the contact at the specified Exchange that will facilitate the trading of the equity portion of the buy_write		
9382	MatchType	Int	Internalizing Firm can set the price for its side of the order. 1=guaranteed price (not currently supported), 2=limit price, 3=auto match Values from cmiOrder::MatchType.		
9383	AuctionType	String	CBOE sends this tag to solicit participation in an auction through the Quote Request message. Values: Order, Strategy, Internalization		

Tag #	Tag Name	DataType	CBOE Usage	FIX Messages (required=Y, not required=N)	Standard Adoption Status
9384	AuctionContingency	Int	CBOE sends this tag to solicit participation in an auction through the Quote Request message. Same values as cmiconstants::ContingencyTypes. 1=none, 2=AON, 3=FOK, 4=IOC		
9385	AuctionID	String	Taken from tag 131 (QuoteReqID) in the Quote Request message.		
9433	ExecutionInformation	String	Custom tag originally defined by SIAC for NYSE FCS reports. This field will contain the three character acronym of the broker involved in the trade on behalf of the order originator. If the order is filled in CBOE's hybrid limit order book, the tag will contain "XXH."	8(n)	Added to FIX 4.3 Party Block as Party Role Executing System
9463	SubscriptionRequestType	Int	This tag would be specified in a Security Definition Request if the firm wants to subscribe to security status (product state callbacks). If not specified, the firm would be subscribed to security status by default. 0 = SNAPSHOT only 1 = SNAPSHOT + UPDATES 500=subscribe for Auction participation 501=unsubscribe from Auction participation		

9465	OrderOrigin	String	**Options and Futures**: If a Broker_Dealer enters an order on behalf of a market-maker in the W_MAIN, CFE_MAIN, or ONE_MAIN sessions, the market-maker's acronym must be entered in this tag. The default exchange is CBOE, but an exchange may be specified. This is also called "Originator".
			
			`CBOE:ABC`
			-or you may enter-
			`ABC`		

9467	EquitySession	String	For Buy_writes. Used to specify which "Session" the Equity Leg of the strategy product can be found. Currently, there is only one supported EquitySession value, which is the default ("Underlying").		
9469	ExtendedPriceType	Int	Used to enter Cabinet orders. 4=Cabinet		
6699	ApplicationQueueDepth	Int	A User defined field that provides the number of application level events that are queued for processing behind this current message. For instance, when ApplicationQueueDepth > 0 on a corresponding application response message sent from CBOE to the firm, this indicates that there are still ApplicationQueueDepth # of reports that have yet to be generated and transmitted to the user. This information is provided to help counter parties manage throughput and backlog issues.	N	

Tag #	Tag Name	DataType	CBOE Usage	FIX Messages (required=Y, not required=N)	Standard Adoption Status
6700	ApplicationQueueActionRequest	Char	Optional user defined field that indicates the action that should be taken to resolve the Application queue depth (backlog). 0- No action taken 1- Queue flushed 2- Overlay last 3- End subscription This tag will be applicable ONLY on Market Data subscription request messages. It will be rejected on all the other messages. For additional information regarding this tag, please review document FIX-03b.	N	
6701	ApplicationQueueActionTaken	Char	Optional application message field that indicates the action taken to resolve the Application queue depth (backlog). 0- No action taken 1- Queue flushed 2- Overlaid last 3- Subscription Ended This tag will be applicable ONLY on Market Data response messages.	N	

Tag #	Tag Name	DataType	CBOE Usage	FIX Messages (required=Y, not required=N)	Standard Adoption Status
9730	TradeLiquidityIndicator		This tag is used in the W_STOCK session for trade type billing indicators. • **Maker = 'A'**: refers to the person adding liquidity to the market, by basically having an order or quote resting in the book to be traded against. (i.e. they are establishing the price) • **Taker = 'R'**: refers to the person taking liquidity from the market, by basically sending an order to trade against the book. (i.e. they are coming in and taking out the best price) • **Flash= 'F'**: refers to an order that is being presented to the dealers for a short-term auction for step-up, before the order is routed to an away exchange for a fill. • **Flash Response= 'E'**: refers to the dealer responding to a flash and effectively stepping up to improve the CBSX market to the prevailing price and fulfilling the customer here. • **Linked Away= 'X'**: refers to an order that was sent to another market for execution. • **Linked Away Response= 'L'**: refers to the response from the other exchange filling the CBSX order sent to them. • **Opening = 'O'**: refers to all executions that take place as part of the opening rotation process itself. • **Cross= 'C'**: refers to a trade whereby both buyer and seller are represented on a single transaction. Thus, neither is really a maker or taker per se, but rather virtually meet one another. • **ODD_LOT_FLASH='N'**: used for CBSX odd lot orders or the odd lot portion of a mixed lot order that is being flashed. • **ODD_LOT_RESPONSE='B'**: used for all responses to odd lot orders that are being flashed. • **RESTING = 'Q'** • **CROSS_PRICE_IMP = 'S'** • **FLASH_PRICE_IMP = 'T'** • **FLASH_RESPONSE_PRICE_IMP = 'U'** • **MAKER_TURNER = 'V'** • **RESTING_TURNER = 'W'**		
9743	PIPManagementType		1=solicitation to participate in an auction. Sent by CBOE.		

Tag #	Tag Name	DataType	CBOE Usage	FIX Messages (required=Y, not required=N)	Standard Adoption Status
20010	TradingGroupInfo	String	Provides trading group information by class. The value will equal the trade server environment. Example: 20010=test1EquityTradeServer1	N	
20101	ShortSaleIndicator		Valid values for short sale positions for the CBSX session: 1 = Buy 2 = Sell 5 = Sell_Short 6 = Sell_Short_Exempt	N	
20102	PendingFillQty	String	If the light order is filled right away this field will be filled with the fill quantity and will be followed by a normal fill report	N	
20103	PendingCxlQty	String	If the light order is cancelled right away this field will have the cancelled quantity and a cancel report will follow this.	N	
20104	TradingSessionID	String	Trading Session	N	