ICE iMpact Multicast Feed Message Specification

Version 1.1.17

Table of Contents

Revisions

Version	Date	Description of Changes
1.0.01	12/05/07	Initial draft, moved some messages from the TCP Feed 1.1 specs. Added new messages.
1.0.02	02/14/08	Updated multicast product group and market type mapping
1.0.05	04/09/08	Options multicast is being enhanced for an upcoming release. Removed option related messages from specification until enhancements are ready.
1.0.06	04/21/08	• Added Market Event Message • Added SystemPricedLegType field to Trade Message • Added MinPrice, MaxPrice, ProductID, ProductName, HubID, HubAlias, StripID and StripName fields to Product Definition Response • Added TextMessageExtraFld to System Text Message
1.0.07	06/12/08	Added unknown test message, which is sent out on test envs only.
1.0.08	08/14/08	Updated Appendix E – ICE Instrument Naming Convention with support of OTC contract symbol.
1.0.09	12/01/08	Added Pre-Open Price Indicator Message
1.0.09	09/16/09	• Moved Supported Market Types list to a separate document on ICE web site. • Moved ICE Instrument Naming Convention to a separate document on ICE web site.
1.0.09	10/15/09	• Changed field name 'TotalVolume' to 'Volume' in Market Statistics Message and Market Snapshot Message.
1.1.01	11/05/09	• Added SecurityType in Product Definition Request • Added Options Product Definition Response
1.1.02	11/10/09	• Added Option Open Interest Message • Added Option Settlement Price Message
1.1.03	03/12/10	• Added 'GetStripInfoMessages' and 'ReservedField1' fields in Login Request • Added Strip Info Message • Added 'ReservedField1' field in Product Definition Response Message and Market Snapshot Message
1.1.04	09/21/10	• Added IsSerialOptionsSupported and IsTradable fields to Product Definitions Response Message. • Added IsImpliedSpreadAtMarketOpen and IsAdjustedTrade fields to Trade Message. • Added 'U' an option for 'SecurityType' field in Product Definition Request. • Added Option Strategy Definition Response Message • Added New Option Strategy Definition Message • Added RFQ Message
1.1.05	02/03/11	• Added new NumOfMarkets field (4 bytes) in Options Product Definition Response Message
1.1.06	02/28/11	• Added SerialUnderlyingMarketID field to Options Product Definition Response Message

1.1.07	03/28/11	• Added AggressorSide field to Trade Message
1.1.08	06/17/11	• Added ContractSymbolExtra to Options Product Definition Response
1.1.08	06/20/11	• Renamed “BlockTradeType” to ‘OffMarketTradeType”
1.1.09	07/08/11	• Added New Options Market Definition Message
1.1.09	08/05/11	• Renamed “TotalVolume” to “Volume” to match Market Statistics Message
1.1.10	08/26/11	• Added new ‘Extra Flags’ field to AddModifyOrder Message and Trade Message • Added OpenInterestDate to Open Interest Message, Options Open Interest Message, and Market Snapshot Message
1.1.11	10/31/11	• Added Old Style Options Trade and Market Stats Message
1.1.12	11/11/11	• Added Interval Price Limit Notification Message
1.1.12	12/02/11	• Added off market trade type for EFM trade
1.1.12	01/27/12	• Corrected field types for message ‘W’
1.1.12	02/16/12	• Added new flag IsLegDealOutsideIPL in Trade Message
1.1.12	06/08/12	• Added Block Trade Type F
1.1.12	06/12/12	• Removed UDS HedgeDelta 1-300 restriction
1.1.13	07/25/12	• Added IsSettlePriceOfficial in Market Snapshot Message
1.1.14	11/16/12	• Added SettlePriceDenominator in ProductDefinitionResponse and OptionsProductDefinitionResponse Message • Added new SettlementPrice field in Settlement Price, Options Settlement Price and MarketSnapshot Messages
1.1.15	10/16/13	• Added Delta in Option Settlement Price Message • Added SequenceWithinMillis in Add/Modify Order Message and Snapshot Order Message
1.1.16	01/24/14	• Added Side in RFQ Message • Added MICCode to Futures/OTC Product Definition Response Message • Added Top10 Price Level Messages to new options depth channels • Added Spot Market Trade Message for spot market channels • Added Futures Strategy Definition Response Message • Added New Futures Strategy Definition Message
1.1.17	03/20/14	• Added UnitQtyDenominator in product definition response messages (both futures and options)

1 Introduction

This document covers all the messages that are supported in iMpact multicast feed.

1.1 Related Documents

ICE iMpact Multicast Feed Technical Specification

2 High Level Message Specification

2.1 Complete List of Messages

2.1.1 TCP Messages

Client Messages	Type	Server Messages	Type
Login	1	Login Response	A
Product Definition	2	Futures/OTC Product Definition Response	B
		Strip Info Message (Optional)	i
		Options Product Definition Response	p
		Options Strategy Definition Response	q
		Futures Strategy Definition Response	d
Historical Replay	7	Historical Replay Response	8
Debug	5	Debug Response	P
Logout	6		
		Heartbeat	Q
		Error Response	S

2.1.2 Multicast Messages

Category	Message	Type
Common Messages	Market Snapshot (snapshot channel)	C
	Trade	G
	Spot Market Trade	Y
	Investigated Trade	H

	Cancelled Trade	I
	Market Statistics	J
	Market State Change	K
	System Text	L
	Open Interest	M
	Open Price	N
	Settlement Price	O
	Marker/Index Prices(Futures/OTC only)	z
	End Of Day Market Summary	u
	Market Event Message	f
	Pre-Open Price Indicator Message	g
	Strip Info Message	i
	Interval Price Limit Notification Message	V
	New Futures Strategy Definition Message	9
	Unknown Test Message (Test Environment Only)	?
Full Order Depth Only (Futures/OTC Only)	Snapshot Order Message (snapshot channel)	D
	Add/Modify Order	E
	Delete Order	F
	Message Bundle Marker	T
Price Level Only	Snapshot Price Level (snapshot channel)	m
	Add Price Level	t
	Change Price Level	s
	Delete Price Level	r
	New Options Strategy Definition Message	U

	New Options Market Definition Message	l
	RFQ Message	k
	Option Open Interest Message	v
	Option Settlement Price Message	w
Price Level Only	Old Style Options Trade and Market Stats Message	W

2.2 Message Type, Length and Unknown Message

The first byte of a message is always used for message type. And the next two bytes are used for the message body length, which is the message length minus 3 bytes. **Client should read the first 3 bytes, get the value of the message body length, and then use it to read the rest of the message, instead of using a hard-coded value.** Even though length is fixed for a message type at a particular time, it could change over time because new fields could be added. Reading message body using the received length value allows client to continue to work when new fields are added at the end of a message.

With message body length, client can also process unknown type because it can skip the rest of the message to get to the next one. This is very important, since new message types or new fields could be added in the future. **Client is required to handle any unknown messages or new fields. This is how iMpact feed supports backward compatibility.**

2.3 Alpha, Numeric and Price Fields

All alpha fields are in ASCII format, left justified and null character padded. Numeric fields are in binary Big Endian (Network Byte Order) format. They are all signed for consistency, even though most likely only price fields (for certain spread markets) could be negative.

In product definition, field “OrderPriceDenominator” indicates the number of decimal places that should be used for order price in a particular market. And “DealPriceDenominator” is for deal related price fields, such as deal price, high, low, etc. For majority of the markets, those two denominators are the same. But they could be different for certain crack and spread markets.

After client reads the value of a price field, it should apply the denominator to get the correct price. For example, if the value of a price field in an order message is 631400 and the order price denominator is 4, the real price is 63.1400.

2.4 Request, Request Sequence ID and Error Response

Field “RequestSeqID” is specified in every request message. Client should assign a unique (per session) number so that the response message could be matched back to the request if needed.

When error occurs on the server processing an incoming request, message "Error Response" with the original request sequence ID will be sent back to client. Errors such as invalid request, invalid market type/ID or no permission to certain market type/ID could happen, though rare. "Error Response" message allows client to know what goes wrong when there is problem with a particular request.

2.5 Multicast Message Block

Each multicast packet contains a message block which could include multiple messages. Each block has a header with session number, sequence number, number of messages and sent timestamp. These are numeric fields in Big Endian format and are signed, consistent with other numeric fields in the feed messages. The following shows how a multicast datagram could look like.

Session Number (2 bytes)	Sequence Number (4 bytes)	Number of Msgs (2 bytes)	Sent DateTime (8 bytes)	Msg #1	……	Msg #n

Session Number and Sequence Number fields can be used for detecting whether a new session has started or there is a sequence gap. Please read "ICE iMpact Multicast Feed Technical Specification" for details on sequence gap detection.

"Number of Messages" field in the header indicates the number of messages contained in the block. It could be 0 in case of heartbeat. "Sent DateTime" field is the timestamp of when the message block is sent. It is the number of milliseconds since Jan 1st, 1970, 00:00:00 GMT

Message block is used for multicast only, not for TCP messages.

2.6 Messages Changed from iMpact TCP Feed 1.1

There is no change in most messages that are also supported in iMpact TCP feed 1.1, except the followings.

Message	**Changes**
Market Snapshot	RequestSeqID removed as snapshot is no longer requested MarketID and MarketType switched position LastMessageSequenceID added for synchronization
Market Snapshot Order	RequestSeqID removed as snapshot is no longer requested MarketType removed
Add/Modify Order	SentTime removed since it available in block header
Delete Order	SentTime removed since it available in block header SecurityType not supported since it is not needed
Trade	SentTime removed since it available in block header

LoginRequest	Removed fields that are no longer needed
Option Open Interest Message	Some fields are different from those in TCP Feed QV spec
Option Settlement Price Message	Some fields are different from those in TCP Feed QV spec

3 TCP Messages

With iMpact multicast feed, client still needs to connect to the TCP server for product definition download and historical replay. These are the messages used through TCP connection.

3.1 Login

3.1.1 Request Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = '1'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	Request sequence ID assigned by client, unique per session
UserName	7	30	Alpha	
Password	37	30	Alpha	
GetStripInfoMessages	67	1	Alpha	Flag to indicate whether the client wants to get Strip Info Messages or not. 'Y' or 'N'. It is 'N' by default.
ReservedField1	68	2	N/A	Reserved for future use

3.1.2 Response Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'A'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	The sequence ID in client's request.
Code	7	1	Alpha	'0' = Success '1' = Invalid Login '3' = Password Expired

Field Name	Offset	Length	Type	Notes
				'X' = Other
Text	8	120	Alpha	Success or failure messages
MarketTypesPermissioned	128	300	Alpha	Market type IDs allowed to access for the user. Char "," is used in between IDs.

3.2 Product Definitions

3.2.1 Request Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = '2'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	Request sequence ID assigned by client, unique per session
MarketType	7	2	Numeric	Market type ID, see Appendix C for the list.
SecurityType	9	1	Alpha	'F' – Futures/OTC (default, if not provided) 'O' – Options 'U' – UDS Options markets 'D' – UDS Futures markets

3.2.2 Futures/OTC Product Definition Response Message

If SecurityType is 'F' in the request, the server will return Futures/OTC Product Definition response messages.

There are multiple markets per market type. Field "NumOfMarkets" is used to identify how many messages can be expected in total for the given market type.

The multicast feed only supports pre-defined options markets, of which product definitions will be covered in the next section. In the Futures/OTC product definition response, there are some options

related fields that are not used anymore. But we keep them there (instead of removing them) for backward compatibility of fixed length TCP messages. Users should ignore values in the options related fields as stated below.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'B'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	The original request sequence ID assigned by client, unique per session
RequestMarketType	7	2	Numeric	See Appendix C for the list of market types and IDs.
NumOfMarkets	9	2	Numeric	The number of markets for the given market type
MarketID	11	4	Numeric	Unique identifier of a market
ContractSymbol	15	35	Alpha	See Naming Convention on Appendix D
TradingStatus	50	1	Alpha	See appendix A on trading status codes
OrderPriceDenominator	51	1	Alpha	Denominator for the order price fields in this market.
IncrementPrice	52	4	Numeric	Minimum increment price for this market
IncrementQty	56	4	Numeric	Minimum increment quantity for this market
LotSize	60	4	Numeric	The lot size is minimum size of contracts in lots. It is multiplier to determine the total lots.
MarketDesc	64	120	Alpha	Description of the market
MaturityYear	184	2	Numeric	4 digit year
MaturityMonth	186	2	Numeric	Month range 1-12
MaturityDay	188	2	Numeric	
IsSpread	190	1	Alpha	Indicate if the market is a spread
IsCrackSpread	191	1	Alpha	Indicate if the market is crack spread
PrimaryMarketID	192	4	Numeric	Ignored when it is not spread
SecondaryMarketID	196	4	Numeric	Ignored when it is not spread

Field Name	Offset	Length	Type	Notes
IsOptions	200	1	Alpha	Not used. Kept here for backward compatibility.
OptionType	201	1	Alpha	Not used. Kept here for backward compatibility.
StrikePrice	202	8	Numeric	Not used. Kept here for backward compatibility.
SecondStrike	210	8	Numeric	Not used. Kept here for backward compatibility.
DealPriceDenominator	218	1	Alpha	Denominator for the deal price fields in the market. For most markets, this is the same as OrderPriceDenominator. However, it could be different for some crack or spread markets.
MinQty	219	4	Numeric	Minimum quantity for this market
UnitQuantity	223	4	Numeric	The quantity in unit of measurement per lot. For example, it is 1000 barrels per lot for Brent.
Currency	227	20	Alpha	The currency that the market is traded on.
MinStrikePrice	247	8	Numeric	Not used. Kept here for backward compatibility.
MaxStrikePrice	255	8	Numeric	Not used. Kept here for backward compatibility.
IncrementStrikePrice	263	4	Numeric	Not used. Kept here for backward compatibility.
NumDecimalsStrikePrice	267	1	Alpha	Not used. Kept here for backward compatibility.
MinOptionsPrice	268	8	Numeric	Not used. Kept here for backward compatibility.
MaxOptionsPrice	276	8	Numeric	Not used. Kept here for backward compatibility.
IncrementOptionsPrice	284	4	Numeric	Not used. Kept here for backward compatibility.
NumDecimalsOptionsPrice	288	1	Alpha	Not used. Kept here for backward compatibility.

Field Name	Offset	Length	Type	Notes
TickValue	289	8	Numeric	OrderPriceDenominator should be applied to get the real value.
AllowOptions	297	1	Alpha	Indicate if the market supports option markets, 'Y' or 'N'
ClearedAlias	298	15	Alpha	Clearing limit admin related
AllowsImplied	313	1	Alpha	'Y' or 'N'. 'Y' indicates this is a spread market, and, implied is allowed in this market
OptionsExpirationYear	314	2	Numeric	4 digit year
OptionsExpirationMonth	316	2	Numeric	Month range 1-12
OptionsExpirationDay	318	2	Numeric	
MinPrice	320	8	Numeric	Minimum Price
MaxPrice	328	8	Numeric	Maximum Price
ProductID	336	2	Numeric	ID of the product that the contract/market is under.
ProductName	338	62	Alpha	Name of the product that the contract/market is under
HubID	400	2	Numeric	ID of the hub for the contract/market
HubAlias	402	80	Alpha	Alias of the hub for the contract/market
StripID	482	2	Numeric	ID of the strip for the contract/market
StripName	484	39	Alpha	Name of the strip for the contract/market
ReservedField1	523	1	N/A	Reserved for future use
IsSerialOptionsSupported	524	1	Alpha	Indicate if serial options is supported. 'Y' or 'N'.
IsTradable	525	1	Alpha	Indicate if the contract is tradable. 'Y' or 'N'.
SettlePriceDenominator	526	1	Alpha	Denominator for the settlement price fields in the market. For most markets, this is the same as DealPriceDenominator.
MICCode	527	4	Alpha	Market Identifier Code for the market.
UnitQtyDenominator	531	1	Alpha	Denominator for UnitQuantity. Clients should also apply UnitQtyDenominator when calculating LotSize. This field will be '0' for

Field Name	Offset	Length	Type	Notes
				most of the markets.

When there is an error on the server side (one likely reason would be user is not allowed to access a market type), "Error Response" message will be sent to the client. Please see the section about "Error Response" for details on the message format.

It is possible that error occurs for one but not another market type, especially in case of permission issue. On the server side, error for one request doesn't affect the handling of another request. It is up to the client to decide how it would process the error response.

3.2.3 Strip Info Message

This message is returned after Product Definition Response messages if 'GetStripInfoMessages' was set to 'Y' in the login request for the session. Client can expect to receive the same number of Strip Info messages as that for Product Definition Response messages.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'i'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
StripID	3	2	Numeric	
StripType	5	20	Alpha	
BeginYear	25	2	Numeric	4 digit year
BeginMonth	27	2	Numeric	Month range 1-12
BeginDay	29	2	Numeric	
EndYear	31	2	Numeric	4 digit year
EndMonth	33	2	Numeric	Month range 1-12
EndDay	35	2	Numeric	
StripName	37	50	Alpha	

3.2.4 Options Product Definition Response Message

If SecurityType is 'O' in the request, the server will return Options Product Definition response messages. Each option is related to a single underlying instrument. Users should utilize the underlying market id to link to a Futures/OTC product definition response for details on the instrument the option is derived from. All options for a given market type are returned in the response.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'p'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	The original request sequence ID assigned by client, unique per session
RequestMarketType	7	2	Numeric	See Appendix C for the list of market types and IDs.
NumOfMarketsObsolete	9	2	Numeric	**Obsolete**. Clients should use the new 'NumOfMarkets' field (at offset 247), which supports bigger value.
MarketID	11	4	Numeric	Unique identifier of the option market
UnderlyingMarketID	15	4	Numeric	Underlying Futures/OTC market id. This market id links to the product definition of the futures market.
ContractSymbol	19	35	Alpha	See Naming Convention on Appendix D
TradingStatus	54	1	Alpha	See appendix A on trading status codes
OrderPriceDenominator	55	1	Alpha	Denominator for the order price fields in this market.
IncrementQty	56	4	Numeric	Minimum increment quantity for this market
LotSize	60	4	Numeric	The lot size is minimum size of contracts in lots. It is multiplier to determine the total lots.
MarketDesc	64	120	Alpha	Description of the market
OptionType	184	1	Alpha	"C" – Call "P" – Put
StrikePrice	185	8	Numeric	Strike Price of the option. Used in

Field Name	Offset	Length	Type	Notes
				conjunction with the NumDecimalsStrikePrice. This is often different from the premium price decimals.
DealPriceDenominator	193	1	Alpha	Denominator for the deal price fields in the market. For most markets, this is the same as OrderPriceDenominator.
MinQty	194	4	Numeric	Minimum quantity for this market
Currency	198	20	Alpha	The currency that the market is traded on.
NumDecimalsStrikePrice	218	1	Alpha	Denominator for the strike price field.
MinOptionsPrice	219	8	Numeric	Minimum premium price for the option.
MaxOptionsPrice	227	8	Numeric	Maximum premium price for the option.
IncrementPremiumPrice	235	4	Numeric	Price increment for the option market.
OptionsExpirationYear	239	2	Numeric	4 digit year
OptionsExpirationMonth	241	2	Numeric	Month range 1-12
OptionsExpirationDay	243	2	Numeric	Day of the month.
OptionsSettlmentType	245	1	Alpha	'A' – American 'E' – European
OptionsExpirationType	246	1	Alpha	'M' – Monthly 'D' – Daily
NumOfMarkets	247	4	Numeric	The number of options markets for the given market type
SerialUnderlyingMarketID	251	4	Numeric	The underlying futures market ID for a serial option. The serial option market may or may not be a valid futures month and option will expire/exercise into a position held in this underlying market. It will be set to -1 when not applicable.
ContractSymbolExtra	255	35	Alpha	Extra contract symbol. Some contract symbols might contain more than 35 characters. Clients should append this field to ContractSymbol (Offset 19) to get the complete contract symbol.

Field Name	Offset	Length	Type	Notes
SettlePriceDenominator	290	1	Alpha	Denominator for the settlement price fields in the market. For most markets, this is the same as DealPriceDenominator.
UnitQtyDenominator	291	1	Alpha	Denominator for UnitQuantity. This field will be '0' for most of the markets.

3.2.5 Options Strategy Definition Response Message

If SecurityType is 'U' in the request, the server will return Options Strategy Definition Response messages.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'q'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	The original request sequence ID assigned by client, unique per session
RequestMarketType	7	2	Numeric	See Appendix C for the list of market types and IDs.
NumOfMarkets	9	2	Numeric	The number of markets for the given market type.
MarketID	11	4	Numeric	Unique identifier of the market
UnderlyingMarketID	15	4	Numeric	Unique identifier of the underlying market
ContractSymbol	19	35	Alpha	
TradingStatus	54	1	Alpha	See appendix A on trading status codes
OrderPriceDenominator	55	1	Alpha	Denominator for the order price fields in this market.
IncrementPrice	56	4	Numeric	Minimum increment premium price for this market
IncrementQty	60	4	Numeric	Minimum increment quantity for this market

Field Name	Offset	Length	Type	Notes
MinQty	64	4	Numeric	Minimum quantity for this market
NumberOfLegDefinition	68	1	Numeric	Number of strategy leg definitions. The leg info are in repeating group followed
-> LegBodyLength		1	Numeric	Message length, including this field, for a leg. Client should get this value and read the repeating group based on this. New field could be added to the leg definition repeating group and client should be able to handle that. Currently the LegBodyLength is 12 bytes.
-> LegMarketID		4	Numeric	Market Id of the option leg market
-> LegUnderlyingMarketID		4	Numeric	Futures market id of the underlying futures market
-> LegRatio		2	Numeric	Number of option contracts per increment quantity.
-> LegSide		1	Alpha	'1' – Buy '2' – Sell
NumberOfHedgeDefinition		1	Numeric	Number of strategy hedge definitions. The hedge info are in repeating group followed
-> HedgeBodyLength		1	Numeric	Message length, including this field, for a hedge. Client should get this value and read the repeating group based on this. New field could be added to the hedge definition repeating group and client should be able to handle that. Currently the HedgeBodyLength is 18 bytes.
-> HedgeMarketID		4	Numeric	Future's market id of the hedge
-> HedgeSecurityType		1	Alpha	'F' – Future
-> HedgeSide		1	Alpha	'1' – Buy '2' – Sell
-> HedgePrice		8	Numeric	
-> HedgePriceDenominator		1	Alpha	
-> HdegeDelta		2	Numeric	Value between 1 – 300

3.2.6 Futures Strategy Definition Response Message

If SecurityType is 'D' in the request, the server will return Futures Strategy Definition Response messages.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'd'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	The original request sequence ID assigned by client, unique per session
RequestMarketType	7	2	Numeric	See Appendix C for the list of market types and IDs.
NumOfMarkets	9	2	Numeric	The number of markets for the given market type.
MarketID	11	4	Numeric	Unique identifier of the market
ContractSymbol	15	70	Alpha	
TradingStatus	85	1	Alpha	See appendix A on trading status codes
OrderPriceDenominator	86	1	Alpha	Denominator for the order price fields in this market.
IncrementPrice	87	4	Numeric	Minimum increment premium price for this market
IncrementQty	91	4	Numeric	Minimum increment quantity for this market
MinQty	95	4	Numeric	Minimum quantity for this market
NumberOfLegDefinition	99	1	Numeric	Number of strategy leg definitions. The leg info are in repeating group followed
-> LegBodyLength		1	Numeric	Message length, including this field, for a leg. Client should get this value and read the repeating group based on this. New field could be added to the leg definition repeating group and client should be able to handle that. Currently the LegBodyLength is 8 bytes.
-> LegMarketID		4	Numeric	Market Id of the futures leg market

Field Name	Offset	Length	Type	Notes
-> LegRatio		2	Numeric	Number of futures contracts per increment quantity.
-> LegSide		1	Alpha	'1' – Buy '2' – Sell

3.3 Historical Replay

3.3.1 Request Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = '7'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	Request sequence ID assigned by client, unique per session. This has nothing to do with the sequence number in multicast.
MulticastGroupAddr	7	15	Alpha	The multicast group address of the channel in which we want to get the historical messages.
MulticastPort	22	2	Numeric	The multicast group port of the channel in which we want to get the historical messages.
SessionID	24	2	Numeric	The ID of the multicast session in which we want to get the historical messages.
StartSequenceNumber	26	4	Numeric	
EndSequenceNumber	30	4	Numeric	

3.3.2 Response Message

If the server finds the messages for the requested sequence gap, it will send the following response and then those messages. Otherwise, Error Response message is sent to the client. Be aware that it is considered an error if the server can only find some but not all the messages requested.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = '8'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	Request sequence ID assigned by client, unique per session
MulticastGroupAddr	7	15	Alpha	The multicast group address of the channel in which we want to get the historical messages.
MulticastPort	22	2	Numeric	The multicast group port of the channel in which we want to get the historical messages.
SessionID	24	2	Numeric	The ID of the multicast session in which we want to get the historical messages.
StartSequenceNumber	26	4	Numeric	
EndSequenceNumber	30	4	Numeric	

3.4 Debug Message

Debug request could be used programmatically by client, or something as simple as telnet into the server port for troubleshooting connectivity related issues.

3.4.1 Request Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = '5'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	Request sequence ID assigned by client, unique per session

3.4.2 Response Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'P'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	The original request sequence ID assigned by client, unique per session.
Text	7	60	Alpha	Debug text message from server

3.5 HeartBeat

This message is for TCP Only. For multicast, a heartbeat is just a message block with only the header.

3.5.1 HeartBeat Message (TCP only)

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'Q'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
DateTime	3	8	Numeric	Date time the message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT

3.6 Logout

3.6.1 Request Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = '6'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
RequestSeqID	3	4	Numeric	Request sequence ID assigned by client, unique per session

3.6.2 Response Message

There is no dedicated response message to Logout request. The server simply logout the user from the system and closes the connection.

3.7 Error Response Message

3.7.1 Error Response Message

Error response message is sent to client when there is error processing a request. Client feed handler should be ready to handle error response after it submits a request. At minimum, it should log the error messages.

Field Name	Offset	Length	Type	Notes
`MessageType`	0	1	Alpha	Value = 'S'
`MessageBodyLength`	1	2	Numeric	Message body length, excluding 1st and this field
`RequestSeqID`	3	4	Numeric	The original request sequence ID assigned by client, unique per session.
`Code`	7	1	Alpha	'1' – Unknown request '2' – Invalid market type '3' – Market type access denied '4' – Login session required for the request 'X' – Other error
`Text`	8	100	Alpha	The error message

4 Multicast Messages

These are the messages used in the multicast channels.

4.1 Common Messages

These are the messages common to all multicast channels, regardless it is full order depth or price level.

4.1.1 Market Snapshot Message

The market snapshot message is the same for full order depth and price level snapshot channel. The field "NumOfBookEntries" indicates the number of book entries in the snapshot for a given market. It is the number of MarketSnapshotOrder messages that will follow in case of full order depth snapshot channel, and the number of MarketSnapshotPriceLevel messages in case of price level snapshot channel.

Note: for any given market, if the "NumOfBookEntries" is greater than 0, it is possible to receive the entire market snapshot (which comprises of Market Snapshot Message and, MarketSnapshotOrder or MarketSnapshotPriceLevel Messages) in multiple multicast message blocks. Clients should NOT assume the entire market snapshot would be contained in one message block.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'C'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
MarketType	7	2	Numeric	
TradingStatus	9	1	Alpha	See Appendix A on the trading status codes
Volume	10	4	Numeric	Electronic trade volume only, excluding block and other volumes.
BlockVolume	14	4	Numeric	
EFSVolume	18	4	Numeric	
EFPVolume	22	4	Numeric	
OpenInterest	26	4	Numeric	
OpeningPrice	30	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.

Field Name	Offset	Length	Type	Notes
SettlementPriceWithDealPricePrecision	38	8	Numeric	DealPriceDenominator for the market should be applied to get this price. This field is kept here for backward compatibility. Client should use the new SettlementPrice field (added in 1.1.14) for better precision. DealPriceDenominator and SettlePriceDenominator might be different for some markets.
High	46	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
Low	54	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
VWAP	62	8	Numeric	Weighted Average Price. DealPriceDenominator for the market should be applied to get the real price.
NumOfBookEntries	70	4	Numeric	Number of book entries in the market. It is the number of order messages followed for full order depth snapshot channel. In case of price level snapshot, it is the number of price level messages that followed for the market.
LastTradePrice	74	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
LastTradeQuantity	82	4	Numeric	
LastTradeDateTime	86	8	Numeric	Milliseconds since Jan 1st, 1970, 00:00:00 GMT
SettlePriceDateTime	94	8	Numeric	Milliseconds since Jan 1st, 1970, 00:00:00 GMT. If there is no settlement price, the value is -1.
LastMessageSequenceID	102	4	Numeric	This should be used for synchronization with live update messages. Please see the main tech spec for details on how it can be done.
ReservedField1	106	2	N/A	Reserved for future use
OpenInterestDate	108	10	Alpha	The date Open Interest is effective for, in the format of YYYY-MM-DD. It will be blank is there is no Open Interest for the market.
IsSettlePriceOfficial	118	1	Alpha	Indicate if the SettlementPrice is official, 'Y' or 'N'.

Field Name	Offset	Length	Type	Notes
SettlementPrice	119	8	Numeric	SettlePriceDenominator for the market should be applied to get the real settlement price.

4.1.2 Trade Message

A trade with IsSystemPricedLeg equal to 'Y' should not be used for the last price. Neither should you use off market trade (OffMarketTradeType not equal to ' ') for that. This message will not be sent for trades in spot markets.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'G'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	MarketID of the instrument that was traded.
TradeID	7	8	Numeric	Unique identifier of the trade message, unique per market.
IsSystemPricedLeg	15	1	Alpha	Indicate if it is a system priced leg, 'Y' or 'N'
Price	16	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
Quantity	24	4	Numeric	
OffMarketTradeType	28	1	Alpha	Only for off market trade. The value is ' ' when it is a regular trade. See Appendix B for the codes and descriptions.
TransactDateTime	29	8	Numeric	Deal date time. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
SystemPricedLegType	37	1	Alpha	'C' – System Priced Crack Spread Leg 'S' – System Priced Leg This can be ignored if IsSystemPricedLeg='N'
IsImpliedSpreadAtMarketOpen	38	1	Alpha	Indicate if the trade happens at market open due to spread implied. When set to 'Y', such deal should not be included in market stats.
IsAdjustedTrade	39	1	Alpha	Indicate if the trade is an adjusted trade, 'Y' or 'N'

Field Name	Offset	Length	Type	Notes
AggressorSide	40	1	Alpha	‘ ‘ – No Aggressor ‘1’ – Buy ‘2’ – Sell
ExtraFlags	41	1	Numeric	Bit 0 (Least Significant Bit): IsRFCCrossing – indicate this is a RFC Crossing Deal if set to 1. Bit 1: IsLegDealOutsideIPL – indicate the deal is outside of IPL (when IPL is enabled) if set to 1. When set to 1, such deal should not be included in market stats. This could only happen in leg markets due to implied orders. Bit 2 thru 7: Reserved for future use. For backward compatibility, client should always look at each individual bit for the corresponding flag. Otherwise problems might occur when bits 2 thru 7 start to be utilized.

4.1.3 Spot Market Trade Message

This message will be sent only upon trade in spot markets and it will be rendered on spot market channels.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = ‘Y’
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	MarketID of the instrument that was traded.
TradeID	7	8	Numeric	Unique identifier of the trade message, unique per market.
Price	15	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
Quantity	23	4	Numeric	
TransactDateTime	27	8	Numeric	Deal date time. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
ExtraFlags	35	1	Numeric	For Future use
DeliveryBeginDateTime	36	8	Numeric	Delivery begin date time. Milliseconds since Jan

Field Name	Offset	Length	Type	Notes
				1st, 1970, 00:00:00 GMT
DeliveryEndDateTime	44	8	Numeric	Delivery end date time. Milliseconds since Jan 1st, 1970, 00:00:00 GMT

4.1.4 Investigated Trade Message

This message is sent when a trade is put under investigation or the investigation is completed. Client can use the market ID and order ID to find and flag the original trade if needed.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'H'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
TradeID	7	8	Numeric	
Price	15	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
Quantity	23	4	Numeric	
OffMarketTradeType	27	1	Alpha	Only for block trade. The value is ' ' when it is a regular trade. See Appendix B for the codes and descriptions.
DateTime	28	8	Numeric	Date time the trade was investigated. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
Status	36	1	Alpha	'1' – Under Investigation '2' – Investigation Completed

4.1.5 Cancelled Trade Message

This message is sent when a trade is cancelled. Client can use the market ID and order ID to find and update the original trade if needed. But it has no effect on the book.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'I'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
TradeID	7	8	Numeric	
Price	15	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
Quantity	23	4	Numeric	
OffMarketTradeType	27	1	Alpha	Only for block trade. The value is ' ' when it is a regular trade. See Appendix B for the codes and descriptions.
DateTime	28	8	Numeric	Date time the trade was cancelled. Milliseconds since Jan 1st, 1970, 00:00:00 GMT

4.1.6 Market Statistics Message

This is usually sent after a trade or cancelled trade message. Sometimes, you may get a statistics message without getting a trade message.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'J'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
Volume	7	4	Numeric	Electronic trade volume only, excluding block and other volumes.
BlockVolume	11	4	Numeric	
EFSVolume	15	4	Numeric	
EFPVolume	19	4	Numeric	
High	23	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.

Field Name	Offset	Length	Type	Notes
Low	31	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
VWAP	39	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
DateTime	47	8	Numeric	Date time the stat was updated. Milliseconds since Jan 1st, 1970, 00:00:00 GMT

4.1.7 Market State Change Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'K'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
TradingStatus	7	1	Alpha	See Appendix A on the trading status codes
DateTime	8	8	Numeric	Date time the message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT

4.1.8 System Text Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'L'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
TextMessage	3	200	Alpha	
DateTime	203	8	Numeric	Date time the message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
TextMessageExtraFld	211	800	Alpha	Extra field for text message when TextMessage field is not big enough. This should be appended to TextMessage if it is not empty.

4.1.9 Open Interest Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'M'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
OpenInterest	7	4	Numeric	
OpenInterestChange	11	4	Numeric	
DateTime	15	8	Numeric	Date time the message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
OpenInterestDate	23	10	Alpha	The date this Open Interest is effective for, in the format of YYYY-MM-DD

4.1.10 Open Price Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'N'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
OpenPrice	7	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
DateTime	15	8	Numeric	Date time the message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT

4.1.11 Settlement Price Message

Settlement prices could be official or unofficial. For a given market, the exchange usually sends out unofficial price before the official one.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'O'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
SettlementPriceWithDealPricePrecision	7	8	Numeric	DealPriceDenominator for the market should be applied to get this price. This field is kept here for backward compatibility. Client should use the new SettlementPrice field (added in 1.1.14) for better precision. DealPriceDenominator and SettlePriceDenominator might be different for some markets.
DateTime	15	8	Numeric	Date time the message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
IsOfficial	23	1	Alpha	Flag to indicate this is official settlement price or not. 'Y' or 'N'.
ValuationDateTime	24	8	Numeric	Date time the settlement price is for. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
SettlementPrice	32	8	Numeric	SettlePriceDenominator for the market should be applied to get the actual settlement price.

4.1.12 Marker/Index Prices

The exchange sends out Marker/Index Prices when there is an update. It is possible that the same Marker/Index price is sent out more than once for a market sometime. Client can compare messages with PublishedDateTime for a given market and valuation date, and only does update when a message is the latest.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'z'

Field Name	Offset	Length	Type	Notes
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
Price	7	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
ShortName	15	30	Alpha	The short name date of the Marker/Index. For example *"Morn5Min"*
PublishedDateTime	45	8	Numeric	The date and time the marker was put into. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
ValuationDate	53	10	Alpha	The date this price is effective for, in the format of YYYY-MM-DD

4.1.13 End of Day Market Summary Message

The message is streamed to client when market is closed and settlement price is available for the current trading day. This message is supported for both options and non-options markets.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'u'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
Volume	7	4	Numeric	
BlockVolume	11	4	Numeric	
EFSVolume	15	4	Numeric	
EFPVolume	19	4	Numeric	
OpeningPrice	23	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
High	31	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
Low	39	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.

Field Name	Offset	Length	Type	Notes
VWAP	47	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
SettlementPrice	55	8	Numeric	DealPriceDenominator for the market should be applied to get the real price.
OpenInterest	63	4	Numeric	
DateTime	67	8	Numeric	Date time this message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT

4.1.14 Market Event Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'f'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
EventType	7	1	Alpha	'A' – Implication Disabled for the Market
DateTime	8	8	Numeric	Date time this message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT

4.1.15 Pre-Open Price Indicator Message

This message contains the estimate of what the opening price could be, based on the orders in the market or previous settlement price. Currently it is sent out every 1 minute during pre-open.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'g'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
PreOpenPrice	7	8	Numeric	DealPriceDenominator for the market should be

Field Name	Offset	Length	Type	Notes
				applied to get the real price.
DateTime	15	8	Numeric	Date time the message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT

4.1.16 Strip Info Message

This message will be sent out if there is strip date change during the trading session. If client does not care about strip date changes, client should ignore these messages.
Please refer to section 3.2.3 for the message format.

4.1.17 Interval Price Limit Notification Message

Interval Price Limit (IPL) might be enabled for certain markets. IPL check is to prevent sudden movements (in both directions) in the market during a short period of time. If IPL is violated, there will be a Hold period where prices outside of IPL will not be allowed. IPL notifications will be sent out to market participants about such violation (IPL Hold Start). Notifications will be sent out after the Hold period expires (IPL Hold End). Note that trading within the IPL limit is still allowed during IPL Hold period.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'V'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
IPLHoldType	7	1	Alpha	IPL Hold Type: 'S' – IPL Hold Start 'E' – IPL Hold End
NotificationDateTime	8	8	Numeric	Date time of the IPL Hold notification. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
IsUp	16	1	Alpha	'Y' – IPL Upper bound violation (Bidding too high) 'N' – IPL Lower bound violation (Asking too low) N/A when IPLHoldType = 'E'
IPLHoldDuration	17	4	Numeric	Hold duration, in milliseconds.

Field Name	Offset	Length	Type	Notes
				N/A when IPLHoldNotifyType = 'E'
IPLUp	21	8	Numeric	IPL upper bound. OrderPriceDenominator for the market should be applied to get the real price limit. N/A when IPLHoldNotifyType = 'E'
IPLDown	29	8	Numeric	IPL lower bound. OrderPriceDenominator for the market should be applied to get the real price limit. N/A when IPLHoldNotifyType = 'E'

4.1.18 New Futures Strategy Definition Message

New Futures Strategy Definition messages will be sent out when new UDS markets for futures are created. These messages can be ignored if clients are not interested in the UDS markets.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = '9'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	Unique identifier of the market
ContractSymbol	7	70	Alpha	
TradingStatus	77	1	Alpha	See appendix A on trading status codes
OrderPriceDenominator	78	1	Alpha	Denominator for the order price fields in this market.
IncrementPrice	79	4	Numeric	Minimum increment premium price for this market
IncrementQty	83	4	Numeric	Minimum increment quantity for this market
MinQty	87	4	Numeric	Minimum quantity for this market
NumberOfLegDefinition	91	1	Numeric	Number of strategy leg definitions. The leg info are in repeating group followed
-> LegBodyLength	92	1	Numeric	Message length, including this field, for a leg. Client should get this value and read the repeating group based on this. New field could

Field Name	Offset	Length	Type	Notes
				be added to the leg definition repeating group and client should be able to handle that. Currently the LegBodyLength is 8 bytes.
-> LegMarketID		4	Numeric	Market Id of the futures leg market
-> LegRatio		2	Numeric	Number of futures contracts per increment quantity.
-> LegSide		1	Alpha	'1' – Buy '2' – Sell

4.1.19 Unknown Test Message (For Test Environments Only)

Client is required to handle any new type of messages that could be added in the future. Please read section 2.2 on how to process unknown messages.

To ensure that client is able to handle unknown messages, we broadcast an unknown test message in all test environments periodically (currently every 5 minutes).

4.2 Messages for Full Order Depth Only (Futures/OTC)

The messages under this section are for full order depth channels only. You can ignore them if you don't subscribe to those channels.

4.2.1 Market Snapshot Order Message

This message is for orders in snapshot only. It is different from the order message for incremental updates. For a given market, these messages follow right after Market Snapshot Message.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'D'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
OrderID	7	8	Numeric	Unique per market
OrderSequenceID	15	2	Numeric	Sequence ID of the order. When an order is modified, this will be incremented while OrderID remains the same. It is for legacy reason and can be ignored.
Side	17	1	Alpha	1 = Bid, 2 = Offer
Price	18	8	Numeric	OrderPriceDenominator for the market should be applied to get the real price.
Quantity	26	4	Numeric	
IsImplied	30	1	Alpha	Indicate if this is an implied order or not
IsRFQ	31	1	Alpha	Indicate whether it is just an RFQ or not. **Client should filter the order if it doesn't care about RFQ.**
OrderEntryDateTime	32	8	Numeric	Order entry date time. Milliseconds since Jan 1st, 1970, 00:00:00 GMT.
SequenceWithinMillis	40	4	Numeric	Can be used in conjunction with "OrderEntryDate Time" field for priority of orders within same milliseconds time.

4.2.2 Add/Modify Order Message

Both add and modify order notifications use one message format. Client should add the order to book if it is not there already. Otherwise, just overwrite the existing order.

Orders outside daily price limit are not in the matching engine, and thus are excluded.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'E'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
OrderID	7	8	Numeric	Unique per market
OrderSequenceID	15	2	Numeric	Sequence ID of the order. When an order is modified, this will be incremented while OrderID remains the same. It is for legacy reason and can be ignored.
Side	17	1	Alpha	1 = Bid, 2 = Offer
Price	18	8	Numeric	OrderPriceDenominator for the market should be applied to get the real price.
Quantity	26	4	Numeric	
IsImplied	30	1	Alpha	Indicate if this is an implied order or not
IsRFQ	31	1	Alpha	Indicate whether it is just an RFQ or not. **Client should filter the order if it doesn't care about RFQ.**
OrderEntryDateTime	32	8	Numeric	Order entry date time. Milliseconds since Jan 1st, 1970, 00:00:00 GMT.
ExtraFlags	40	1	Numeric	Least Significant Bit (Bit 0): IsModifyOrder – indicate this is to Modify existing order if set to 1. Bit 1 thru 7: Reserved for future use. For backward compatibility, client should always look at each individual bit for the corresponding flag. Otherwise problems might occur when bits 1 thru 7 start to be utilized.

Field Name	Offset	Length	Type	Notes
SequenceWithinMillis	41	4	Numeric	Can be used in conjunction with "OrderEntryDate Time" field for priority of orders within same milliseconds time.

4.2.3 Delete Order Message

Upon receipt of this message, client should remove the order from its local book. Under certain scenarios, the message could be sent from backend with an OrderID that doesn't exist on client's book, in which case client can just ignore it.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'F'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
OrderID	7	8	Numeric	

4.2.4 Trade Message

Trade message was defined in "Common Messages" section**. Upon receipt of this message, client is required to remove the order referenced (of which order ID equals to trade ID in the trade message)**, instead of just deducting the quantity, because in case of partial fill for a resting order, the backend creates a new order with the remaining quantity using a new ID (though priority and entry timestamp remains the same). Read the section on message bundle marker if you want to do special processing in the case of partial fill.

4.2.5 Message Bundle Marker

This message indicates where a bundle of messages starts or ends. For example, if it is the start marker, the messages followed in the stream are part of a bundle, until the end marker. The messages in the same bundle are results from one transaction in the backend.

Message bundle is specifically added so that customers can process messages resulted from partial fill differently if they want to. Currently, when a resting order is partially filled, a new order is created for the remaining quantity using a new ID (though priority and entry timestamp remain the same). Client gets a trade message and add/modify order message. Instead of treating them independently, which would

remove the whole quantity of the order (because of the trade) first and then add the remaining quantity back, some customers prefer to process them in one transaction.

This is a workaround message that might not be needed in future releases. And not all customers care about it, depending on the specific applications. You can ignore the message if it is applicable for your application.

Note: a bundle could span across multiple multicast blocks.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'T'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
StartOrEnd	3	1	Alpha	'S' – Start of a message bundle 'E' – End of a message bundle

4.3 Messages for Price Level Only

The messages under this section are for price level channels only. You can ignore them if you don't subscribe to those channels.

Appendix E includes some price level related scenarios and demonstrates how client should handle different messages to update the book accordingly.

4.3.1 Market Snapshot Price Level Message

This message is for price level in snapshot only. For a given market, these messages follow right after Market Snapshot Message.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'm'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
Side	7	1	Alpha	1 = Bid, 2 = Offer
PriceLevelPosition	8	1	Numeric	Position of the price level
Price	9	8	Numeric	OrderPriceDenominator for the market should be applied to get the real price.
Quantity	17	4	Numeric	Total quantity at the price level
OrderCount	21	2	Numeric	Number of all orders at the price level
ImpliedQuantity	23	4	Numeric	Quantity that were implied
ImpliedOrderCount	27	2	Numeric	Number of implied orders at the price level

4.3.2 Add Price Level Message

Upon receipt of this message, client should add/insert a price level at the specified position in the book for the given market, and push down the price levels that were previously at or below that position. After that, if the total number of levels exceeds what is supported (e.g. Top 5), client should remove the bottom level. The system doesn't send out Delete Price Level message in that scenario.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 't'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
Side	7	1	Alpha	1 = Bid, 2 = Offer
PriceLevelPosition	8	1	Numeric	Position of the price level
Price	9	8	Numeric	OrderPriceDenominator for the market should be applied to get the real price.
Quantity	17	4	Numeric	Total quantity at the price level
OrderCount	21	2	Numeric	Number of all orders at the price level
ImpliedQuantity	23	4	Numeric	Quantity that were implied
ImpliedOrderCount	27	2	Numeric	Number of implied orders at the price level

4.3.3 Change Price Level Message

Upon receipt of this message, client should update the price level at the specified position in its book for the given market.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 's'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
Side	7	1	Alpha	1 = Bid, 2 = Offer
PriceLevelPosition	8	1	Numeric	Position of the price level
Price	9	8	Numeric	OrderPriceDenominator for the market should be applied to get the real price.
Quantity	17	4	Numeric	Total quantity at the price level
OrderCount	21	2	Numeric	Number of all orders at the price level

Field Name	Offset	Length	Type	Notes
ImpliedQuantity	23	4	Numeric	Quantity that were implied
ImpliedOrderCount	27	2	Numeric	Number of implied orders at the price level

4.3.4 Delete Price Level Message

Upon receipt of this message, client should remove the price level at the specified position in its book for the given market. And it should pull up all the levels that were below that position.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'r'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
Side	7	1	Alpha	1 = Bid, 2 = Offer
PriceLevelPosition	8	1	Numeric	Position of the price level

4.3.5 Trade Message

Trade message was defined in "Common Messages" section. Unlike for Full Order Depth, trade message should not be used for price level book related processing.

4.3.6 New Options Strategy Definition Message

New Options Strategy Definition messages will be sent out when new UDS markets for options are created. These messages can be ignored if clients are not interested in the UDS markets.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'U'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field

Field Name	Offset	Length	Type	Notes
MarketID	3	4	Numeric	Unique identifier of the market
UnderlyingMarketID	7	4	Numeric	Unique identifier of the underlying market
ContractSymbol	11	35	Alpha	
TradingStatus	46	1	Alpha	See appendix A on trading status codes
OrderPriceDenominator	47	1	Alpha	Denominator for the order price fields in this market.
IncrementPrice	48	4	Numeric	Minimum increment premium price for this market
IncrementQty	52	4	Numeric	Minimum increment quantity for this market
MinQty	56	4	Numeric	Minimum quantity for this market
NumberOfLegDefinition	60	1	Numeric	Number of strategy leg definitions. The leg info are in repeating group followed
-> LegBodyLength	61	1	Numeric	Message length, including this field, for a leg. Client should get this value and read the repeating group based on this. New field could be added to the leg definition repeating group and client should be able to handle that. Currently the LegBodyLength is 12 bytes.
-> LegMarketID		4	Numeric	Market Id of the option leg market
-> LegUnderlyingMarketID		4	Numeric	Futures market id of the underlying futures market
-> LegRatio		2	Numeric	Number of option contracts per increment quantity.
-> LegSide		1	Alpha	'1' – Buy '2' – Sell
NumberOfHedgeDefinition		1	Numeric	Number of strategy hedge definitions. The hedge info are in repeating group followed
-> HedgeBodyLength		1	Numeric	Message length, including this field, for a hedge. Client should get this value and read the repeating group based on this. New field could be added to the hedge definition repeating group and client should be able to handle that. Currently the HedgeBodyLength is 18 bytes.

Field Name	Offset	Length	Type	Notes
-> HedgeMarketID		4	Numeric	Future's market id of the hedge
-> HedgeSecurityType		1	Alpha	'F' – Future
-> HedgeSide		1	Alpha	'1' – Buy '2' – Sell
-> HedgePrice		8	Numeric	
-> HedgePriceDenominator		1	Alpha	
-> HdegeDelta		2	Numeric	

4.3.7 New Options Market Definition Message

New Options Market Definition messages will be sent out when new options markets are created. Once created, these new options markets should be treated no differently than those that are pre-defined.
In case of missing this message(s) in live updates, all (new) options market definitions are available via the Options Product Definition Requests.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'l'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	Unique identifier of the option market
UnderlyingMarketID	7	4	Numeric	Underlying Futures/OTC market id. This market id links to the product definition of the futures market.
ContractSymbol	11	70	Alpha	See Naming Convention on Appendix D
TradingStatus	81	1	Alpha	See appendix A on trading status codes
OrderPriceDenominator	82	1	Alpha	Denominator for the order price fields in this market.
IncrementQty	83	4	Numeric	Minimum increment quantity for this market

Field Name	Offset	Length	Type	Notes
LotSize	87	4	Numeric	The lot size is minimum size of contracts in lots. It is multiplier to determine the total lots.
MarketDesc	91	120	Alpha	Description of the market
OptionType	211	1	Alpha	"C" – Call "P" – Put
StrikePrice	212	8	Numeric	Strike Price of the option. Used in conjunction with the NumDecimalsStrikePrice. This is often different from the premium price decimals.
DealPriceDenominator	220	1	Alpha	Denominator for the deal price fields in the market. For most markets, this is the same as OrderPriceDenominator.
MinQty	221	4	Numeric	Minimum quantity for this market
Currency	225	20	Alpha	The currency that the market is traded on.
NumDecimalsStrikePrice	245	1	Alpha	Denominator for the strike price field.
MinOptionsPrice	246	8	Numeric	Minimum premium price for the option.
MaxOptionsPrice	254	8	Numeric	Maximum premium price for the option.
IncrementPremiumPrice	262	4	Numeric	Price increment for the option market.
OptionsExpirationYear	266	2	Numeric	4 digit year
OptionsExpirationMonth	268	2	Numeric	Month range 1-12
OptionsExpirationDay	270	2	Numeric	Day of the month.
OptionsSettlmentType	272	1	Alpha	'A' – American 'E' – European
OptionsExpirationType	273	1	Alpha	'M' – Monthly 'D' – Daily
SerialUnderlyingMarketID	274	4	Numeric	The underlying futures market ID for a serial option. The serial option market may or may not be a valid futures month and option will expire/exercise into a position held in this underlying market.

Field Name	Offset	Length	Type	Notes
				It will be set to -1 when not applicable.

4.3.8 RFQ Message

Request for Quote Message.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'k'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
MessageTimestamp	7	8	Numeric	Date time of the RFQ; milliseconds since Jan 1st, 1970, 00:00:00 GMT
RFQSystemID	15	8	Numeric	System ID of the RFQ
MarketTypeID	23	2	Numeric	
UnderlyingMarketID	25	4	Numeric	
Quantity	29	4	Numeric	
Side	33	1	Alpha	' ' – N/A '1' – Bid '2' – Offer

4.3.9 Option Open Interest Message

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'v'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
OpenInterest	7	4	Numeric	

DateTime	11	8	Numeric	Date time the message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
OpenInterestDate	19	10	Alpha	The date this Open Interest is effective for, in the format of YYYY-MM-DD

4.3.10 Option Settlement Price Message

Option settlement prices could be official or unofficial. For a given market, the exchange usually sends out unofficial price before the official one.

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'w'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
MarketID	3	4	Numeric	
SettlementPriceWithDealPricePrecision	7	8	Numeric	DealPriceDenominator for the market should be applied to get this price. This field is kept here for backward compatibility. Client should use the new SettlementPrice field (added in 1.1.14) for better precision. DealPriceDenominator and SettlePriceDenominator might be different for some markets.
DateTime	15	8	Numeric	Date time the message was sent. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
IsOfficial	23	1	Alpha	Flag to indicate this is official settlement price or not. 'Y' or 'N'.
ValuationDateTime	24	8	Numeric	Date time the settlement price is for. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
Volatility	32	8	Numeric	Apply 2 as the denominator to get the real value. For example, volatility of 3.00 will be sent as 300.
SettlementPrice	40	8	Numeric	SettlePriceDenominator for the market should be applied to get the actual settlement price.
Delta	48	8	Numeric	Apply 2 as the denominator to get the real value. For example, delta of 3.00 will be sent as 300.

4.3.11 Old Style Options Trade and Market Stats Message

Old style options markets are ***not*** pre-defined and do ***not*** have their own market IDs. Deals and market statistics for old style options will be sent out via this message type. These messages can be ignored if clients are not interested in old style options. Currently it is only used for OTC options (except Henry Hub).

Field Name	Offset	Length	Type	Notes
MessageType	0	1	Alpha	Value = 'W'
MessageBodyLength	1	2	Numeric	Message body length, excluding 1st and this field
UnderlyingMarketID	3	4	Numeric	The underlying market ID of this options market
TradeID	7	8	Numeric	
Price	15	8	Numeric	NumDecimalsOptionsPrice from the underlying market should be applied to get the real price.
Quantity	23	4	Numeric	
OffMarketTradeType	27	1	Alpha	Only for off market trade. The value is ' ' when it is a regular trade. See Appendix B for the codes and descriptions.
TransactDateTime	28	8	Numeric	Deal date time. Milliseconds since Jan 1st, 1970, 00:00:00 GMT
OptionType	36	1	Alpha	'1' – Call '2' – Put
StrikePrice	37	8	Numeric	NumDecimalsStrikePrice from the underlying market should be applied to get the real strike price
EventCode	45	1	Alpha	'0' – Normal trade '1' – Cancelled trade '2' – Adjusted trade

TotalVolume	46	4	Numeric	N/A if set to -1
BlockVolume	50	4	Numeric	N/A if set to -1
EFSVolume	54	4	Numeric	N/A if set to -1
EFPVolume	58	4	Numeric	N/A if set to -1
High	62	8	Numeric	NumDecimalsOptionsPrice from the underlying market should be applied to get the real price. N/A if set to -1
Low	70	8	Numeric	NumDecimalsOptionsPrice from the underlying market should be applied to get the real price. N/A if set to -1
VWAP	78	8	Numeric	NumDecimalsOptionsPrice from the underlying market should be applied to get the real price. N/A if set to -1

5 Options Support

For options, top of book as well as Top 10 Price levels are supported. And thus price level messages should be used.

6 Programming Guidelines

The followings are the list of things that we recommend for message processing, many of which were already mentioned in earlier sections.

- Client should request product definitions only once a day and cache the data, so that it can be quicker to recover in the middle of a trading day.

- Client should handle unknown message type. Client should work without change if new field is added to a message.

- Client should implement some kind of queuing mechanism for storing messages so that it can read incoming messages quicker. Consumption of the messages can be done in a separate thread.

- Error response could be expected for a TCP request. On the server side, error in handling of one request doesn't affect the processing of another request. It is up to the client to decide how it wants to handle an error response. But at minimum, the error response should be logged.

- There are two denominators for price fields, one for orders and the other for deals. They are the same for majority of the markets. However, they could be different for certain crack and spread markets. "DealPriceDenominator" should be used for deal price, market high, low, vwap, opening price and settlement price.

7 Appendices

Appendix A: Trading Status Codes

Code	Description
O	Open
C	Close
E	Expired
1	Pre-Open
2	Pre-Close

Appendix B: Off Market Trade Type

Code	Description
K	Block Trade
E	EFP Trade
S	EFS Trade
F	EFP/EFS Contra Trade
V	Bilateral Off-Exchange Trade
O	NG EFP/EFS Trade
9	CCX EFP Trade
J	EFR Trade
Q	EOO Trade (for US Futures Options only)
I	EFM Trade

Appendix C: Supported Market Types

Market types that ICE currently supports can be found at the URL below. They can be used in Product Definition and Market Data Requests.

https://www.theice.com/publicdocs/technology/Supported_Market_Types_on_ICE_API.pdf

Appendix D: ICE Instrument Naming Convention

ICE Instrument Naming Convention document can be found at the following URL.

https://www.theice.com/publicdocs/technology/Instrument_Naming_Convention.pdf

Appendix E: Price Level Scenarios

The followings include a few typical price level related scenarios and demonstrate how client should handle the messages and update its local book correctly.

1. Initial Book for Market 234678

Book on the Bid side

PriceLevelPosition	Bid
1	5 @ 78.15
2	10 @ 78.10
3	10 @ 78.00
4	15 @ 77.95
5	5 @ 77.90

2. Price Level Added for Market 234678

A new bid of 20 is floated into the system with price 78.05. The system sends out the following Add Price Level Message.

MessageType	't'
MessageBodyLength	29
Market ID	234678
Side	'1'
PriceLevelPosition	**3**
Price	**7805**
Quantity	**20**
OrderCount	1
ImpliedQuantity	0
ImpliedOrderCount	0

** 2 is the value of OrderPriceDenomintor for market 234678.

Upon receipt of this message, client should insert the price level at position 3 of the book on the bid side, and move previous level 3 and 4 down 1 position. Previous entry at position 5 should be deleted.

Book on the Bid side after Update

PriceLevelPosition	Bid
1	5 @ 78.15
2	10 @ 78.10
3	**20 @ 78.05**
4	10 @ 78.00
5	15 @ 77.95

3. Price Level Changed for Market 234678

Another bid of 10 is floated into the system with price 78.05. Since the price level was there, the system sends out the following Change Price Level Message.

MessageType	's'
MessageBodyLength	29
Market ID	234678
Side	'1'
PriceLevelPosition	**3**
Price	**7805**
Quantity	**30**
OrderCount	2
ImpliedQuantity	0
ImpliedOrderCount	0

Upon receipt of this message, client should update the price level at position 3 of the book on the bid side.

Book on the Bid side after Update

PriceLevelPosition	Bid
1	5 @ 78.15
2	10 @ 78.10
3	**30 @ 78.05**
4	10 @ 78.00

5	15 @ 77.95

4. Price Level Deleted for Market 234678

Orders at 78.00 are withdrawn and the server sends out the following Delete Price Level message.

MessageType	'r'
MessageBodyLength	9
Market ID	234678
Side	'1'
PriceLevelPosition	4

Upon receipt of this message, client deletes price level entry at position 4, and move entry at position 5 up 1 level.

Book on the Bid side after Delete

PriceLevelPosition	Bid
1	5 @ 78.15
2	10 @ 78.10
3	20 @ 78.05
4	**15 @ 77.95**

If there are more price levels below position 4 after the delete on the server side, the system would send out the following Add Price Level message for position 5.

MessageType	't'
MessageBodyLength	29
Market ID	234678
Side	'1'
PriceLevelPosition	**5**
Price	**7790**
Quantity	**5**

OrderCount	1
ImpliedQuantity	0
ImpliedOrderCount	0

Upon receipt of this message, client should add price level at position 5.

Book on the Bid side after Update

PriceLevelPosition	Bid
1	5 @ 78.15
2	10 @ 78.10
3	20 @ 78.05
4	15 @ 77.95
5	**5 @ 77.90**