

---

**ACTIVETICK MARKETIF**  
**MULTICAST INTERFACE SPECIFICATION**  
**REVISION 1.7**

---

## Table of Contents

<b>ACTIVETICK MARKETIF</b> .....	1
<b>MULTICAST INTERFACE SPECIFICATION</b> .....	1
<b>REVISION 1.5</b> .....	1
<b>1.0 Introduction</b> .....	4
<b>1.1 Background</b> .....	4
<b>1.2 Exchange-facing/Sender Mode</b> .....	4
<b>1.3 Receiver/Normalizer Mode</b> .....	4
<b>2.0 Transmission Characteristics</b> .....	5
<b>2.1 Transmission Block</b> .....	5
<b>2.2 Data Endianness</b> .....	5
<b>3.0 Message Header</b> .....	5
<b>3.1 Message Id</b> .....	6
<b>3.2 Message Size</b> .....	6
<b>3.3 Sequence Number</b> .....	7
<b>4.0 Feed Messages</b> .....	7
<b>4.1 Feed Message Header</b> .....	7
<b>4.1.1 Timestamp</b> .....	8
<b>4.1.2 Symbol</b> .....	8
<b>4.1.3 Source</b> .....	8
<b>4.2 Sequenced Message Header</b> .....	8
<b>4.3 Feed Message Top BBO Quote</b> .....	9
<b>4.4 Feed Message Trade</b> .....	9
<b>4.5 Feed Message Volume</b> .....	10
<b>4.6 Feed Message Refresh</b> .....	10
<b>4.6.1 Refresh Exchange Session State</b> .....	11

---

4.6.2	Refresh Exchange Session Data .....	11
4.6.3	Refresh Intraday Data .....	12
4.6.4	Refresh Option End Of Day.....	13
4.6.5	Refresh Option Open Interest .....	14
4.6.6	Refresh Equity Split.....	14
4.7	Feed Message Refresh Instrument Definition.....	15
4.7.1	Equity Instrument Definition.....	15
4.7.2	Option Equity Instrument Definition .....	15
4.7.3	Future Instrument Definition .....	16
4.7.5	Future Spread Instrument Definition.....	18
4.8	Feed Message Refresh Instrument Definition Continuation.....	19
4.9	Refresh Instrument Definition Appendages.....	20
4.9.1	Instrument Definition Event Type Appendage.....	20
4.9.2	Instrument Definition Feed Type Appendage.....	20
4.9.3	Instrument Definition Attribute Appendage .....	20
4.9.4	Instrument Definition Lot Type Appendage .....	20
4.9.5	Instrument Definition Underlying Appendage.....	21
4.9.6	Instrument Definition Leg Appendage .....	21
4.10	Feed Message Fundamental .....	21
4.11	Feed Message Book Insert Quote.....	21
4.12	Feed Message Book Update Quote .....	22
4.13	Feed Message Book Delete Quote .....	22
4.14	Feed Message Book Delete Quote Range .....	23
4.15	Feed Message Book Trade Execution .....	23
4.16	Feed Message Book Reset .....	24
5.0	Field Descriptions.....	24

---

## 1.0 Introduction

### 1.1 Background

ActiveTick Market Interface (MarketIf) is a software service that offers normalization of exchange's data into a proprietary messaging format. Normalized data produced by MarketIf standardizes data across all supported exchanges into a single unified format. This feed specification outlines the format of output data produced by MarketIfs.

In addition to normalization functionality, MarketIf also provides efficient transport of data feed from point A to point B using proprietary real-time compression techniques without any added latency.

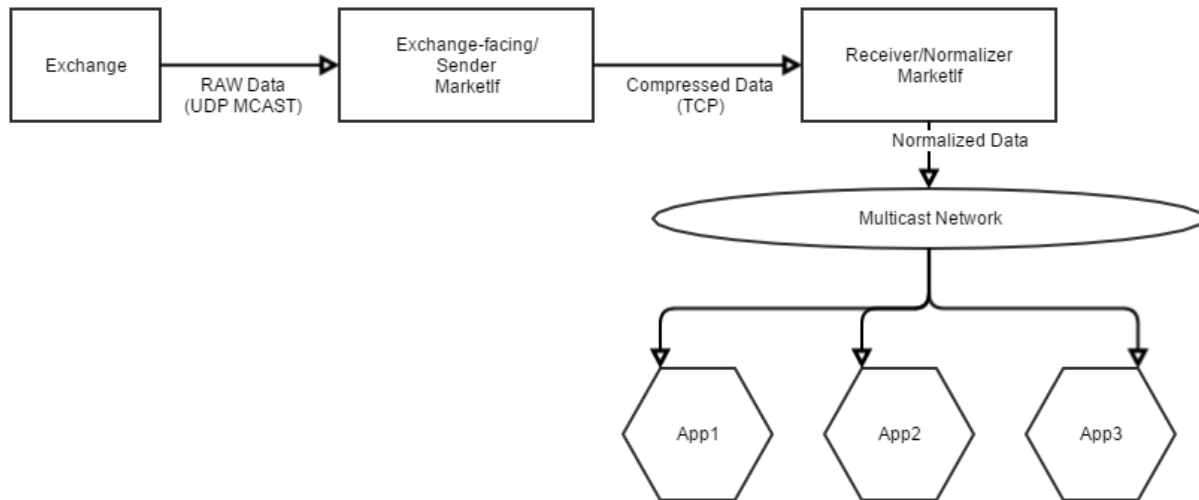
MarketIf can function in three modes, as exchange-facing/remote sender, as remote receiver/normalizer, or as exchange-facing/normalizer.

### 1.2 Exchange-facing/Sender Mode

Exchange-facing side of MarketIf receives original RAW data from exchange, typically by subscribing to exchange's multicast channels, and consumes the data into internal format. After each message is consumed, the data is compressed and transported to a remote or local MarketIf running in Receiver/Normalizer mode.

### 1.3 Receiver/Normalizer Mode

MarketIf configured as Receiver/Normalizer accepts data forwarded to it by MarketIf running as Exchange-facing/Sender. Forwarded data received by MarketIf gets uncompressed, validated, and then normalized into standard Active Tick's proprietary messaging format. After normalization, the data gets published onto predefined set of multicast channels, from where it can be picked up by multicast subscribers.

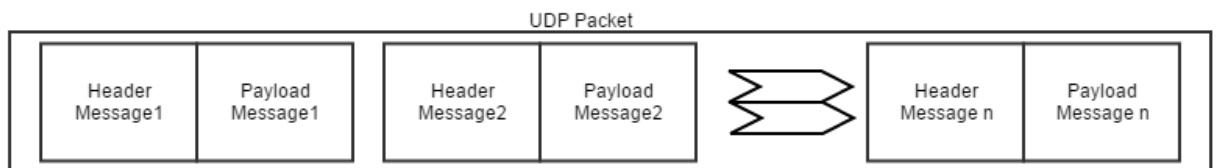


## 2.0 Transmission Characteristics

### 2.1 Transmission Block

Each normalized UDP packet send by MarketIf contains one or more messages. Messages are stacked next to each other within the packet, up to maximum allowed payload size of 1420 bytes.

Each individual message always contains a message header, followed by a message payload.



### 2.2 Data Endianness

All normalized data is sent with Little Endian byte order.

## 3.0 Message Header

Message header is placed in front of the message, and contains information about a message payload that follows it. Message header is used as a lower-level transmission header. The size of the header is 7 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Message Id</b>	UInt8	1
<b>Message Size</b>	UInt16	2
<b>Sequence Number</b>	UInt32	4

### 3.1 Message Id

The Message Id identifies type of message that follows the message header, and consists of the following IDs:

Message Id	Value
Feed Message Top BBO Quote	0
Feed Message Trade	1
Feed Message Refresh	2
Feed Message Refresh Instrument Definition	3
Feed Message Refresh Instrument Definition Continuation	4
Feed Message Fundamental	6
Feed Message Volume	7
Feed Message Book Insert Quote	51
Feed Message Book Update Quote	52
Feed Message Book Delete Quote	53
Feed Message Book Delete Quote Range	54
Feed Message Book Trade Execution	55
Feed Message Book Reset	56
Feed Message Long Top BBO Quote	100
Feed Message Long Trade	101
Feed Message Long Refresh	102
Feed Message Long Refresh Instrument Definition	103
Feed Message Long Refresh Instrument Definition Continuation	104
Feed Message Long Fundamental	105
Feed Message Long Volume	106
Feed Message Long Book Insert Quote	150
Feed Message Long Book Update Quote	151
Feed Message Long Book Delete Quote	152
Feed Message Long Book Delete Quote Range	153
Feed Message Long Book Trade Execution	154
Feed Message Long Book Reset	155

### 3.2 Message Size

The message size field indicates the size of the payload that follows the message header, excluding size of the message header. For example, if payload is 10 bytes, message size will be set to 10.

### 3.3 Sequence Number

The sequence number incrementally increases with each message sent by MarketIf. The number gets reset to 1 when it reaches 0xffffffff. A special value of 0 indicates that all consumers should reset their sequence numbers to 1.

## 4.0 Feed Messages

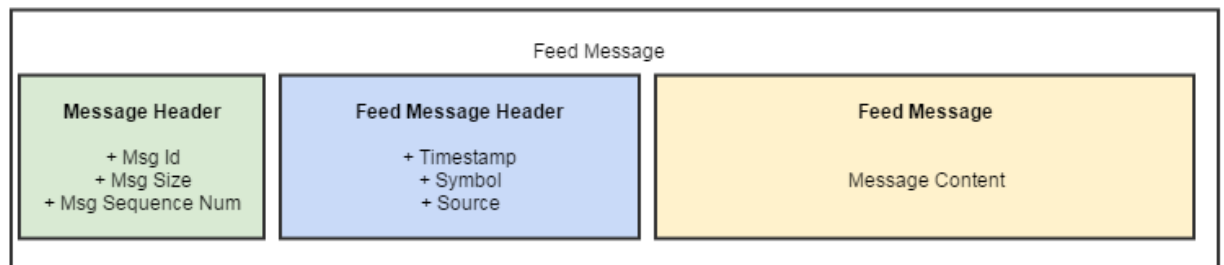
All feed messages generated by MarketIf contain a higher-level feed message header, which follows right after lower-level transmission message header. This header contains common data fields which are present in all data feed messages. In the context of a message, combined feed message header and message content is what constitutes a single message payload.

There are two versions of a feed message header, one that uses standard length symbol and another that uses a long length symbol. In order to efficiently manage bandwidth, MarketIf determines whether a message needs standard or long versions of the feed message header. Message content that follows feed message header is always the same length.

Consumers should always process both types of feed message headers.

### 4.1 Feed Message Header

There are two versions of feed message headers, short and long version. The short version of the feed message header is used for messages with symbol size up to 11 characters in length, while the long version is used with symbol sizes up to 32 characters.



Standard feed message header length is 23 bytes, whereas long feed message header length is 44 bytes. The header consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Timestamp</b>	DATETIME	8
<b>Symbol</b>	SYMBOL or LONG SYMBOL	14 or 35
<b>Source</b>	UInt8	1

#### 4.1.1 Timestamp

Timestamp is a 64-bit unsigned integer that represents time in nanoseconds since epoch, i.e. 00:00 Jan 1 1970. The time is represented in UTC. For example, timestamp value of 1513204919123456000 would be: 2017-12-13 22:41:59.123456000.

#### 4.1.2 Symbol

Symbol is used to identify the instrument for which the message is intended for. Based on length of the symbol, there are two versions of the symbol, standard and long. For detailed information, see SYMBOL under Field Descriptions.

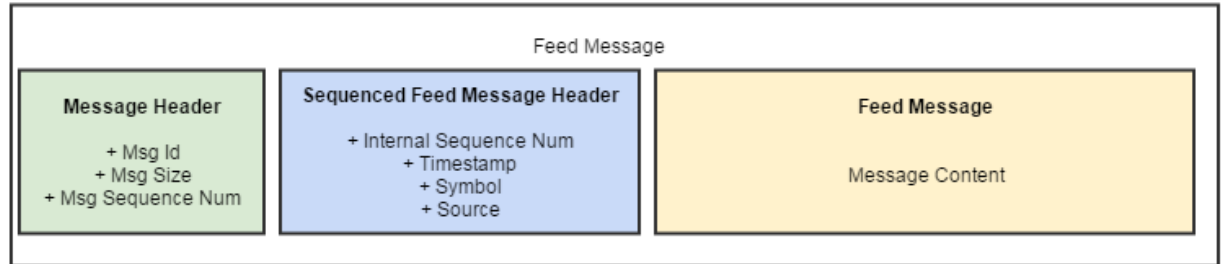
#### 4.1.3 Source

Source field identifies which MarketIf generated the message. For detailed information, see SOURCE under Field Descriptions.

## 4.2 Sequenced Message Header



Sequenced feed message headers are used when there is a strict requirement for feed message related sequencing, such as book messages.



The header contains a total of 31 or 52 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Sequence Number</b>	UInt64	8
<b>Timestamp</b>	DATETIME	8
<b>Symbol</b>	SYMBOL or LONG SYMBOL	14 or 35
<b>Source</b>	UInt8	1

#### 4.3 Feed Message Top BBO Quote

Message Id	Value
<b>Feed Message Top BBO Quote</b>	0

Top of the book best bid/offer quote message is sent whenever there is a change in BBO pricing information, contains a total of 52 or 73 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG HEADER	23 or 44
<b>Condition</b>	UInt8	1
<b>Bid Exchange</b>	UInt8	1
<b>Ask Exchange</b>	UInt8	1
<b>Bid Price</b>	PRICE	5
<b>Ask Price</b>	PRICE	5
<b>Bid Size</b>	UInt32	4
<b>Ask Size</b>	UInt32	4
<b>Reserved Field</b>	UInt8	8

#### 4.4 Feed Message Trade

Message Id	Value
<b>Feed Message Trade</b>	1

Last sale message is sent when there is a sale for a given security, contains a total of 54 or 75 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG HEADER	23 or 44
<b>Flags</b>	UInt32	4
<b>Condition1</b>	UInt8	1
<b>Condition2</b>	UInt8	1
<b>Condition3</b>	UInt8	1
<b>Condition4</b>	UInt8	1
<b>Last Exchange</b>	UInt8	1
<b>Last Price</b>	PRICE	5
<b>Last Size</b>	UInt32	4
<b>Reserved Field</b>	UInt8	13

#### 4.5 Feed Message Volume

Message Id	Value
<b>Feed Message Volume</b>	7

Volume message is sent when a trade happens outside normal trading conditions. This type of message contains last traded size and flags.

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG HEADER	23 or 44
<b>Volume Flags</b>	UInt32	4
<b>Last Size</b>	UInt32	4

#### 4.6 Feed Message Refresh

Message Id	Value
<b>Feed Message Refresh</b>	2

Refresh message is sent during a special event at the exchange, or within ActiveTick's Ticker Plant processing loop. Refresh messages should use Message Header's size to indicate the length of the feed message.

There are several refresh messages, including:

Refresh Message Type	Value
Refresh Message Intraday	0
Refresh Message Option End of Day	1
Refresh Message Option Open Interest	2
Refresh Message Exchange Session State	3
Refresh Message Exchange Session Data	4
Refresh Message Equity Split	5

#### 4.6.1 Refresh Exchange Session State

Refresh Message Type	Value
<b>Refresh Message Exchange Session State</b>	<b>3</b>

Session state is generated by MarketIf whenever exchange state change event occurs for the whole market, or for individual instrument. The state change typically occurs whenever a session is opened or closed, but can also occur intraday, whenever a new instrument begins trading on the exchange. In the latter case, MarketIf sends out a new instrument definition, followed by Session Data refresh message, followed by Session State message.

If symbol is empty for this message, then message is applied to all symbols based on message's source. Otherwise, the session state message is applicable only to the symbol provided by this message.

The message contains a total of 3 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Refresh Message Type</b>	UInt8	1
<b>Session Type</b>	UInt8	1
<b>Session State</b>	UInt8	1

#### 4.6.2 Refresh Exchange Session Data

Refresh Message Type	Value
<b>Refresh Message Session Data</b>	<b>4</b>

Exchange Session Data message is typically sent at the beginning of each exchange session to indicate information pertaining to current session. This message is also generated for a new instrument that begins trading on the exchange for the first time.

Session Data message can contain up to 10 different session events. For example, a typical equity instrument with extended hours trading would contain pre-market event, regular market event, and after-hours event, for a total of 3 events.

If symbol is empty for this message, then message is applied to all symbols based on message's source. Otherwise, the session state message is applicable only to the symbol provided by this message.

The message contains a total of 192 bytes, and consists of the header and 10 session events following the header:

Field Name	Type	Length (bytes)
<b>Refresh Message Type</b>	UInt8	1
<b>Events Count</b>	UInt8	1

Session event data contains the following fields:

Field Name	Type	Length (bytes)
<b>Session Type</b>	UInt8	1
<b>Begin Time</b>	DATETIME	8
<b>End Time</b>	DATETIME	8
<b>Begin Session State</b>	UInt8	1
<b>End Session State</b>	UInt8	1

#### 4.6.3 Refresh Intraday Data

Refresh Message Type	Value
<b>Refresh Message Intraday</b>	0

Refresh Intraday message contains data fields which have to be used to overwrite any stored value. This message is used to roll previous day's closing

price. Flags field is used in a bitwise AND operation to check for existence of a Refresh Intraday Flag, and if it set, appropriate value should be used from the data structure. The message contains a total of 87 bytes, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Refresh Message Type</b>	UInt8	1
<b>Flags</b>	UInt32	4
<b>Last Trade Condition1</b>	UInt8	1
<b>Last Trade Condition2</b>	UInt8	1
<b>Last Trade Condition3</b>	UInt8	1
<b>Last Trade Condition4</b>	UInt8	1
<b>Quote Condition</b>	UInt8	1
<b>Pre Market Open Price</b>	PRICE	5
<b>Open Price</b>	PRICE	5
<b>Last Price</b>	PRICE	5
<b>High Price</b>	PRICE	5
<b>Low Price</b>	PRICE	5
<b>Close Price</b>	PRICE	5
<b>Previous Close Price</b>	PRICE	5
<b>After Market Close Price</b>	PRICE	5
<b>Bid Price</b>	PRICE	5
<b>Ask Price</b>	PRICE	5
<b>Last Exchange</b>	UInt8	1
<b>Bid Exchange</b>	UInt8	1
<b>Ask Exchange</b>	UInt8	1
<b>Bid Size</b>	UInt32	4
<b>Ask Size</b>	UInt32	4
<b>Last Size</b>	UInt32	4
<b>Volume</b>	UInt64	8
<b>Reserved Field</b>	UInt8	4

#### 4.6.4 Refresh Option End Of Day

Refresh Message Type	Value
<b>Refresh Message End Of Day</b>	1

Open EOD message is sent to summarize today's session information. The message contains a total of 41 bytes, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Refresh Message Type</b>	UInt8	1

<b>Volume</b>	UInt32	4
<b>Open Interest Volume</b>	UInt32	1
<b>Open Price</b>	PRICE	1
<b>High Price</b>	PRICE	1
<b>Low Price</b>	PRICE	1
<b>Last Price</b>	PRICE	1
<b>Underlying Price</b>	PRICE	5
<b>Bid Price</b>	PRICE	5
<b>Ask Price</b>	PRICE	5
<b>Reserved Field</b>	UInt8	16

#### 4.6.5 Refresh Option Open Interest

Refresh Message Type	Value
<b>Refresh Message Open Interest</b>	2

Option Open Interest message is sent to update open interest. The message contains a total of 21 bytes, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Refresh Message Type</b>	UInt8	1
<b>Open Interest Volume</b>	UInt32	4
<b>Reserved Field</b>	UInt8	16

#### 4.6.6 Refresh Equity Split

Refresh Message Type	Value
<b>Refresh Message Equity Split</b>	5

Equity Split message is sent when there is a split within equity. The message contains a total of 58 bytes, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Refresh Message Type</b>	UInt8	1
<b>Previous Amount</b>	UInt64	8
<b>New Amount</b>	UInt64	8
<b>Announce Date</b>	UInt64	8
<b>Pay Date</b>	UInt64	8
<b>Ex Date</b>	UInt64	8
<b>Is Optionable</b>	UInt8	1
<b>Reserved Field</b>	UInt8	16

## 4.7 Feed Message Refresh Instrument Definition

Message Id	Value
<b>Feed Message Refresh Instrument Definition</b>	3

Instrument Definition message is sent whenever there is a new instrument added at the exchange level, or is modified or deleted. In case if message is larger than maximum allowed UDP packet, the message is followed by an Instrument Definition Continuation message. The message content is unique for different types of instruments; currently MarketIf definitions support the following instruments:

Instrument Type	Value
Equity Instrument Type	0
Option Equity Instrument Type	1
Future Instrument Type	2
Future Option Instrument Type	3
Future Spread Instrument Type	4

### 4.7.1 Equity Instrument Definition

The message contains a total of 332 or 353 bytes, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG HEADER	23 or 44
<b>Instrument Type</b>	UInt8	1
<b>Instrument Definition</b>	UInt8	1
<b>Action Type</b>		
<b>Flags</b>	UInt32	4
<b>Primary Exchange</b>	UInt8	1
<b>Short Description</b>	UInt8	50
<b>Long Description</b>	UInt8	100
<b>Sector</b>	UInt8	50
<b>Industry</b>	UInt8	50
<b>SIC</b>	UInt8	10
<b>CIK</b>	UInt8	10
<b>Reserved Field</b>	UInt8	32

### 4.7.2 Option Equity Instrument Definition

The message contains a total of 102 or 123 bytes, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG HEADER	23 or 44
<b>Instrument Type</b>	UInt8	1
<b>Instrument Definition Action Type</b>	UInt8	1
<b>Underlying Symbol</b>	LONG_SYMBOL	35
<b>Expiration Date</b>	UInt32	4
<b>Strike Price</b>	PRICE	5
<b>Is Call</b>	UInt8	1
<b>Reserved Field</b>	UInt8	32

#### 4.7.3 Future Instrument Definition

The message contains a total of 161 or 182 bytes along with variable appendages, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG HEADER	23 or 44
<b>Instrument Type</b>	UInt8	1
<b>Instrument Definition Action Type</b>	UInt8	1
<b>Market Segment Id</b>	UInt8	1
<b>Security Id</b>	UInt32	4
<b>Security Id Source</b>	UInt8	1
<b>Maturity Year</b>	UInt16	2
<b>Maturity Month</b>	UInt8	1
<b>Maturity Day</b>	UInt8	1
<b>Security Group</b>	UInt8	6
<b>Asset</b>	UInt8	6
<b>Security Type</b>	UInt8	6
<b>CFI Code</b>	UInt8	6
<b>Underlying Product</b>	UInt8	1
<b>Security Exchange</b>	UInt8	1
<b>Security Trading Status</b>	UInt8	1
<b>Currency</b>	UInt8	3
<b>Is User Defined Instrument</b>	UInt8	1
<b>Match Algorithm</b>	UInt8	1
<b>Minimum Trading Volume</b>	UInt32	4
<b>Maximum Trading Volume</b>	UInt32	4
<b>Minimum Price Increment</b>	PRICE	5
<b>Minimum Price Increment Amount</b>	PRICE	5



<b>Display Factor</b>	UInt64	8
<b>Main Fraction</b>	UInt8	1
<b>Sub Fraction</b>	UInt8	1
<b>Price Display Format</b>	UInt8	1
<b>Contract Multiplier Unit</b>	UInt8	1
<b>Flow Schedule Type</b>	UInt8	1
<b>Contract Multiplier</b>	UInt32	4
<b>Unit Of Measure</b>	UInt8	30
<b>Unit Of Measure Quantity</b>	PRICE	5
<b>Decay Quantity</b>	UInt32	4
<b>Decay Start Date</b>	UInt32	4
<b>Original Contract Size</b>	UInt32	4
<b>Reserved Field</b>	UInt8	8
<b>Number Of Event Types</b>	UInt8	1
<b>Number Of Feed Types</b>	UInt8	1
<b>Number Of Instrument Attributes</b>	UInt8	1
<b>Number of Lot Type Rules</b>	UInt8	1

#### 4.7.4 Future Option Instrument Definition

The message contains a total of 160 or 181 bytes along with variable appendages, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG HEADER	23 or 44
<b>Instrument Type</b>	UInt8	1
<b>Instrument Definition Action Type</b>	UInt8	1
<b>Market Segment Id</b>	UInt8	1
<b>Security Id</b>	UInt32	4
<b>Security Id Source</b>	UInt8	1
<b>Maturity Year</b>	UInt16	2
<b>Maturity Month</b>	UInt8	1
<b>Maturity Day</b>	UInt8	1
<b>Security Group</b>	UInt8	6
<b>Asset</b>	UInt8	6
<b>Security Type</b>	UInt8	6
<b>CFI Code</b>	UInt8	6
<b>Underlying Product</b>	UInt8	1
<b>Security Exchange</b>	UInt8	1
<b>Security Trading Status</b>	UInt8	1
<b>Strike Price</b>	PRICE	5
<b>Min Cab Price</b>	PRICE	5

<b>Strike Currency</b>	UInt8	3
<b>Currency</b>	UInt8	3
<b>Settle Currency</b>	UInt8	3
<b>Match Algorithm</b>	UInt8	1
<b>Minimum Trading Volume</b>	UInt32	4
<b>Maximum Trading Volume</b>	UInt32	4
<b>Minimum Price Increment</b>	PRICE	5
<b>Minimum Price Increment Amount</b>	PRICE	5
<b>Display Factor</b>	UInt64	8
<b>Tick Rule</b>	UInt8	1
<b>Main Fraction</b>	UInt8	1
<b>Sub Fraction</b>	UInt8	1
<b>Price Display Format</b>	UInt8	1
<b>Unit Of Measure</b>	UInt8	30
<b>Unit Of Measure Quantity</b>	PRICE	5
<b>Reserved Field</b>	UInt8	8
<b>Number Of Event Types</b>	UInt8	1
<b>Number Of Feed Types</b>	UInt8	1
<b>Number Of Instrument Attributes</b>	UInt8	1
<b>Number Of Lot Type Rules</b>	UInt8	1
<b>Number Of Underlyings</b>	UInt8	1

#### 4.7.5 Future Spread Instrument Definition

The message contains a total of 145 or 166 bytes along with variable appendages, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG HEADER	23 or 44
<b>Instrument Type</b>	UInt8	1
<b>Instrument Definition Action Type</b>	UInt8	1
<b>Market Segment Id</b>	UInt8	1
<b>Security Id</b>	UInt32	4
<b>Security Id Source</b>	UInt8	1
<b>Maturity Year</b>	UInt16	2
<b>Maturity Month</b>	UInt8	1
<b>Maturity Day</b>	UInt8	1
<b>Security Group</b>	UInt8	6
<b>Asset</b>	UInt8	6
<b>Security Type</b>	UInt8	6
<b>Security Sub Type</b>	UInt8	5
<b>CFI Code</b>	UInt8	6

<b>Underlying Product</b>	UInt8	1
<b>Security Exchange</b>	UInt8	1
<b>Security Trading Status</b>	UInt8	1
<b>Currency</b>	UInt8	3
<b>Is User Defined Instrument</b>	UInt8	1
<b>Match Algorithm</b>	UInt8	1
<b>Minimum Trading Volume</b>	UInt32	4
<b>Maximum Trading Volume</b>	UInt32	4
<b>Minimum Price Increment</b>	PRICE	5
<b>Minimum Price Increment Amount</b>	PRICE	5
<b>Display Factor</b>	UInt64	8
<b>Tick Rule</b>	UInt8	1
<b>Main Fraction</b>	UInt8	1
<b>Sub Fraction</b>	UInt8	1
<b>Price Display Format</b>	UInt8	1
<b>Unit Of Measure</b>	UInt8	30
<b>Reserved Field</b>	UInt8	8
<b>Number Of Event Types</b>	UInt8	1
<b>Number Of Feed Types</b>	UInt8	1
<b>Number Of Instrument Attributes</b>	UInt8	1
<b>Number Of Lot Type Rules</b>	UInt8	1
<b>Number Of Legs</b>	UInt8	1

#### 4.8 Feed Message Refresh Instrument Definition Continuation

Message Id	Value
<b>Feed Message Refresh Instrument Definition Continuation</b>	4

Instrument Definition Continuation message is sent whenever there is a new instrument added at the exchange level, or is modified or deleted, and its size was not enough to fit onto a single UDP message packet. Only appendages are included with the continuation message.

The message contains a total of 32 or 53 bytes along with variable appendages, and consists of the following fields:

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG HEADER	23 or 44
<b>Instrument Type</b>	UInt8	1
<b>Instrument Definition Action Type</b>	UInt8	1
<b>Number Of Event Types</b>	UInt8	1

<b>Number Of Feed Types</b>	UInt8	1
<b>Number Of Instrument Attributes</b>	UInt8	1
<b>Number Of Lot Type Rules</b>	UInt8	1
<b>Number Of Underlyings</b>	UInt8	1
<b>Number Of Legs</b>	UInt8	1
<b>Is Last</b>	UInt8	1

## 4.9 Refresh Instrument Definition Appendages

### 4.9.1 Instrument Definition Event Type Appendage

The appendage contains a total of 9 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Event Type</b>	UInt8	1
<b>Event Time</b>	UInt64	8

### 4.9.2 Instrument Definition Feed Type Appendage

The appendage contains a total of 4 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Feed Type</b>	UInt8	3
<b>Market Depth</b>	UInt8	1

### 4.9.3 Instrument Definition Attribute Appendage

The appendage contains a total of 5 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Attribute Type</b>	UInt8	1
<b>Attribute Value</b>	UInt32	4

### 4.9.4 Instrument Definition Lot Type Appendage

The appendage contains a total of 9 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Lot Type</b>	UInt8	1
<b>Minimum Lot Size</b>	Double	8

#### 4.9.5 Instrument Definition Underlying Appendage

The appendage contains a total of 40 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Underlying Symbol</b>	LONG_SYMBOL	35
<b>Underlying Security Id</b>	UInt32	4
<b>Underlying Security Id Source</b>	UInt8	1

#### 4.9.6 Instrument Definition Leg Appendage

The appendage contains a total of 55 bytes, and consists of the following data fields:

Field Name	Type	Length (bytes)
<b>Leg Security Symbol</b>	LONG_SYMBOL	35
<b>Leg Security Id</b>	UInt32	4
<b>Leg Security Id Source</b>	UInt8	1
<b>Leg Side</b>	UInt8	1
<b>Leg Ratio Quantity</b>	UInt8	1
<b>Leg Price</b>	PRICE	5
<b>Leg Option Delta</b>	Double	8

#### 4.10 Feed Message Fundamental

Coming Soon.

#### 4.11 Feed Message Book Insert Quote

Message Id	Value
<b>Feed Message Book Insert Quote</b>	51

Book Insert Quote message is sent when a new entry needs to be inserted into order book at specified index. The message is typically generated when a new order is added to the book. The message contains a total of 57 or 78 bytes.

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG SEQUENCED HEADER	31 or 52
<b>Side</b>	UInt8	1
<b>Index</b>	UInt32	4
<b>Flags</b>	UInt32	4
<b>Quantity</b>	UInt32	4
<b>Number of Orders</b>	UInt32	4
<b>Price</b>	PRICE	5
<b>Attribution</b>	UInt8	4

#### 4.12 Feed Message Book Update Quote

Message Id	Value
<b>Feed Message Book Update Quote</b>	52

Book Update Quote message is sent when existing book entry needs to be updated with newer information. The message contains an index at which information must be overwritten with contents of the update message. The message contains a total of 57 or 78 bytes.

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG SEQUENCED HEADER	31 or 52
<b>Side</b>	UInt8	1
<b>Index</b>	UInt32	4
<b>Flags</b>	UInt32	4
<b>Quantity</b>	UInt32	4
<b>Number of Orders</b>	UInt32	4
<b>Price</b>	PRICE	5
<b>Attribution</b>	UInt8	4

#### 4.13 Feed Message Book Delete Quote

Message Id	Value
<b>Feed Message Book Delete Quote</b>	53

Book Delete Quote message is sent when existing entry at specified index needs to be deleted from order book. The message contains a total of 40 or 61 bytes.

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG SEQUENCED HEADER	31 or 52
<b>Side</b>	UInt8	1
<b>Index</b>	UInt32	4
<b>Flags</b>	UInt32	4

#### 4.14 Feed Message Book Delete Quote Range

Message Id	Value
<b>Feed Message Book Delete Quote Range</b>	54

Book Delete Quote Range is sent when existing entries need to be removed from the book. The message contains a total of 44 or 65 bytes.

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG SEQUENCED HEADER	31 or 52
<b>Side</b>	UInt8	1
<b>Index From</b>	UInt32	4
<b>Index To</b>	UInt32	4
<b>Flags</b>	UInt32	4

#### 4.15 Feed Message Book Trade Execution

Message Id	Value
<b>Feed Message Book Trade Execution</b>	55

Book Trade Execution is sent when a trade executes and modifies one of the sides of the book. The message contains a total of 49 or 68 bytes.

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG SEQUENCED HEADER	31 or 52
<b>Flags</b>	UInt32	4
<b>Quantity</b>	UInt32	4

<b>Number of Orders</b>	UInt32	4
<b>Aggressor Side</b>	UInt8	1
<b>Price</b>	PRICE	5

#### 4.16 Feed Message Book Reset

Message Id	Value
<b>Feed Message Book Order Replaced</b>	56

Reset message is sent to completely clear the book for specified source.

Field Name	Type	Length (bytes)
<b>Header</b>	SHORT or LONG SEQUENCED HEADER	31 or 52

## 5.0 Field Descriptions

Field Name	Length (bytes)	Description								
<b>AGGRESSOR SIDE</b>	1	Aggressor side consists of the following values: <table border="1" data-bbox="519 1176 1396 1333"> <thead> <tr> <th>Aggressor Side Type</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>Aggressor Side None</b></td> <td>0</td> </tr> <tr> <td><b>Aggressor Side Buy</b></td> <td>1</td> </tr> <tr> <td><b>Aggressor Side Sell</b></td> <td>2</td> </tr> </tbody> </table>	Aggressor Side Type	Value	<b>Aggressor Side None</b>	0	<b>Aggressor Side Buy</b>	1	<b>Aggressor Side Sell</b>	2
Aggressor Side Type	Value									
<b>Aggressor Side None</b>	0									
<b>Aggressor Side Buy</b>	1									
<b>Aggressor Side Sell</b>	2									
<b>BOOK ATTRIBUTION</b>	4	Book Attribution field contains 4-character identifier that is quoting the price. If no identifier exists, the field is empty.								
<b>BOOK ID</b>	1	Book ID contains book identifier, and consists of the following values: <table border="1" data-bbox="519 1480 1396 1596"> <thead> <tr> <th>Book ID</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>NASDAQ TotalView</b></td> <td>0</td> </tr> <tr> <td><b>CME</b></td> <td>1</td> </tr> </tbody> </table>	Book ID	Value	<b>NASDAQ TotalView</b>	0	<b>CME</b>	1		
Book ID	Value									
<b>NASDAQ TotalView</b>	0									
<b>CME</b>	1									
<b>BOOK SIDE</b>	1	Book side type consists of the following values: <table border="1" data-bbox="519 1669 1396 1816"> <thead> <tr> <th>Side Type</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>Side None</b></td> <td>0</td> </tr> <tr> <td><b>Side Bid</b></td> <td>1</td> </tr> <tr> <td><b>Side Ask</b></td> <td>2</td> </tr> </tbody> </table>	Side Type	Value	<b>Side None</b>	0	<b>Side Bid</b>	1	<b>Side Ask</b>	2
Side Type	Value									
<b>Side None</b>	0									
<b>Side Bid</b>	1									
<b>Side Ask</b>	2									



		<b>Side Implied Bid</b>	3
		<b>Side Implied Ask</b>	4
<b>DATETIME</b>	8	Date time is a 64 bit unsigned integer and represents time in nanoseconds since epoch, ie, 00:00 Jan 1 1970.	
<b>EXCHANGE TYPE</b>	1	Exchange Types consist of the following alphanumeric types:	
		<b>Exchange Type</b>	<b>Value</b>
		<b>CONSOLIDATED</b>	Space
		<b>USA NYSE MKT</b>	A
		<b>USA NASDAQ OMX BX</b>	B
		<b>USA NSE</b>	C
		<b>USA FINRA ADF</b>	D
		<b>USA CQS</b>	E
		<b>USA ISE</b>	I
		<b>USA EDGA</b>	J
		<b>USA EDGX</b>	K
		<b>USA CHICAGO STOCK EXCHANGE</b>	M
		<b>USA NYSE</b>	N
		<b>USA NYSE ARCA</b>	P
		<b>USA NASDAQ</b>	Q
		<b>USA CTS</b>	S
		<b>USA CTA NASDAQ</b>	T
		<b>USA OTC</b>	U
		<b>USA CBOE</b>	W
		<b>USA IEX</b>	V
		<b>USA NASDAQ PHLX</b>	X
		<b>USA BATS Y</b>	Y
		<b>USA BATS</b>	Z
		<b>CANADA TSX</b>	T
		<b>CANADA VENTURE</b>	V
		<b>USA OPTION NYSE AMEX</b>	A
		<b>USA OPTION BOSTON</b>	B
		<b>USA OPTION CBOE</b>	C
		<b>USA OPTION EDGX</b>	E
		<b>USA OPTION ISE GEMINI</b>	H
		<b>USA OPTION ISE</b>	I
		<b>USA OPTION ISE MERCURY</b>	J
		<b>USA OPTION MIAMI</b>	M
		<b>USA OPTION NYSE ARCA</b>	N
		<b>USA OPTION C2</b>	W
		<b>USA OPTION OPRA</b>	O
		<b>USA OPTION MIAX PEARL</b>	P

		USA OPTION NASDAQ	Q
		USA OPTION NASDAQ OMX BX	T
		USA OPTION C2	W
		USA OPTION NASDAQ PHLX	X
		USA OPTION BATS	Z
		USA FUTURE CBOT	O
		USA FUTURE CME	C
		USA FUTURE NYMEX	N
		USA FUTURE COMEX	X
		USA FUTURE KCBT	K
		FOREX	F
<b>QUOTE CONDITION</b>	1	Quote conditions map to exchange-specific quote conditions. The conditions consist of the following values:	
		Quote Condition	Value
		Regular Quote	0
		Regular Two Sided Open Quote	1
		Regular One Sided Open Quote	2
		Slow Ask Quote	3
		Slow Bid Quote	4
		Slow Bid Ask Quote	5
		Slow Due LRP Bid Quote	6
		Slow Due LRP Ask Quote	7
		Slow Due NYSE LRP Quote	8
		Slow Due Set Slow List Bid Ask Quote	9
		Manual Ask Automatic Bid Quote	10
		Manual Bid Automatic Ask Quote	11
		Manual Bid and Ask Quote	12
		Opening Quote	13
		Closing Quote	14
		Closed Quote	15
		Resume Quote	16
		Fast Trading Quote	17
		Trading Range Indication Quote	18
		Market Maker Quotes Closed Quote	19
		Non Firm Quote	20
		News Dissemination Quote	21
		Order Influx Quote	22
		Order Imbalance Quote	23
		Due To Related Security News Dissemination Quote	24
		Due To Related Security News Pending Quote	25
		Additional Information Quote	26

		<b>News Pending Quote</b>	27																																		
		<b>Additional information Due To Related Security Quote</b>	28																																		
		<b>Due To Related Security Quote</b>	29																																		
		<b>In View of Common Quote</b>	30																																		
		<b>Equipment Changeover Quote</b>	31																																		
		<b>No Open No Resume Quote</b>	32																																		
		<b>Sub Penny Trading Quote</b>	33																																		
		<b>Automated Bid No Offer Quote</b>	34																																		
		<b>Luld Price Band Quote</b>	35																																		
		<b>Market Wide Circuit Breaker Level 1</b>	36																																		
		<b>Market Wide Circuit Breaker Level 2</b>	37																																		
		<b>Market Wide Circuit Breaker Level 3</b>	38																																		
		<b>Republished Luld Price Band Quote</b>	39																																		
<b>PRICE</b>	5	<p>Price data structure contains information pertaining to price and decimal location. The Price field contains whole price multiplied by the 10 to the power of precision. For example, for the price of 50.25, the Price field will be set to 5025, and Decimal Precision to 2.</p> <p>The data consists of the following types:</p> <table border="1"> <thead> <tr> <th>Field Name</th> <th>Type</th> <th>Length (bytes)</th> </tr> </thead> <tbody> <tr> <td><b>Price</b></td> <td>UInt32</td> <td>4</td> </tr> <tr> <td><b>Price Decimal Precision</b></td> <td>UInt8</td> <td>1</td> </tr> </tbody> </table>		Field Name	Type	Length (bytes)	<b>Price</b>	UInt32	4	<b>Price Decimal Precision</b>	UInt8	1																									
Field Name	Type	Length (bytes)																																			
<b>Price</b>	UInt32	4																																			
<b>Price Decimal Precision</b>	UInt8	1																																			
<b>INSTRUMENT ATTRIBUTE FLAGS</b>	4	<p>Instrument Attribute Flags are OR'ed into flags value, and consist of the following flags:</p> <table border="1"> <thead> <tr> <th>Flag</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>Electronic Match Eligible</b></td> <td>0x0</td> </tr> <tr> <td><b>Order Cross Eligible</b></td> <td>0x1</td> </tr> <tr> <td><b>Block Trade Eligible</b></td> <td>0x2</td> </tr> <tr> <td><b>EFP Eligible</b></td> <td>0x4</td> </tr> <tr> <td><b>EBF Eligible</b></td> <td>0x8</td> </tr> <tr> <td><b>EFS Eligible</b></td> <td>0x10</td> </tr> <tr> <td><b>EFR Eligible</b></td> <td>0x20</td> </tr> <tr> <td><b>OTC Eligible</b></td> <td>0x40</td> </tr> <tr> <td><b>Link Mass Quoting Eligible</b></td> <td>0x80</td> </tr> <tr> <td><b>Negative Strike Eligible</b></td> <td>0x100</td> </tr> <tr> <td><b>Negative Price Eligible</b></td> <td>0x200</td> </tr> <tr> <td><b>Is Fractional</b></td> <td>0x400</td> </tr> <tr> <td><b>RFQ Cross Eligible</b></td> <td>0x1000</td> </tr> <tr> <td><b>Zero Price Eligible</b></td> <td>0x2000</td> </tr> <tr> <td><b>Decaying Product Eligible</b></td> <td>0x4000</td> </tr> <tr> <td><b>Variable Product Eligible</b></td> <td>0x8000</td> </tr> </tbody> </table>		Flag	Value	<b>Electronic Match Eligible</b>	0x0	<b>Order Cross Eligible</b>	0x1	<b>Block Trade Eligible</b>	0x2	<b>EFP Eligible</b>	0x4	<b>EBF Eligible</b>	0x8	<b>EFS Eligible</b>	0x10	<b>EFR Eligible</b>	0x20	<b>OTC Eligible</b>	0x40	<b>Link Mass Quoting Eligible</b>	0x80	<b>Negative Strike Eligible</b>	0x100	<b>Negative Price Eligible</b>	0x200	<b>Is Fractional</b>	0x400	<b>RFQ Cross Eligible</b>	0x1000	<b>Zero Price Eligible</b>	0x2000	<b>Decaying Product Eligible</b>	0x4000	<b>Variable Product Eligible</b>	0x8000
Flag	Value																																				
<b>Electronic Match Eligible</b>	0x0																																				
<b>Order Cross Eligible</b>	0x1																																				
<b>Block Trade Eligible</b>	0x2																																				
<b>EFP Eligible</b>	0x4																																				
<b>EBF Eligible</b>	0x8																																				
<b>EFS Eligible</b>	0x10																																				
<b>EFR Eligible</b>	0x20																																				
<b>OTC Eligible</b>	0x40																																				
<b>Link Mass Quoting Eligible</b>	0x80																																				
<b>Negative Strike Eligible</b>	0x100																																				
<b>Negative Price Eligible</b>	0x200																																				
<b>Is Fractional</b>	0x400																																				
<b>RFQ Cross Eligible</b>	0x1000																																				
<b>Zero Price Eligible</b>	0x2000																																				
<b>Decaying Product Eligible</b>	0x4000																																				
<b>Variable Product Eligible</b>	0x8000																																				

		<table border="1"> <tr> <td><b>Daily Product Eligible</b></td> <td>0x10000</td> </tr> <tr> <td><b>Implied Matching Eligible</b></td> <td>0x20000</td> </tr> </table>	<b>Daily Product Eligible</b>	0x10000	<b>Implied Matching Eligible</b>	0x20000																				
<b>Daily Product Eligible</b>	0x10000																									
<b>Implied Matching Eligible</b>	0x20000																									
<b>INSTRUMENT EVENT TYPE</b>	1	<p>Instrument Event Type contains the following values:</p> <table border="1"> <thead> <tr> <th>Event Type</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>Event Activation</b></td> <td>0</td> </tr> <tr> <td><b>Event Last Eligible Trade Date</b></td> <td>1</td> </tr> </tbody> </table>	Event Type	Value	<b>Event Activation</b>	0	<b>Event Last Eligible Trade Date</b>	1																		
Event Type	Value																									
<b>Event Activation</b>	0																									
<b>Event Last Eligible Trade Date</b>	1																									
<b>INSTRUMENT DEFINITION ACTION TYPE</b>	1	<p>Instrument Definition Action Type contains the type of action that should be performed on the instrument, and consists of the following values:</p> <table border="1"> <thead> <tr> <th>Action Type</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>Action Add</b></td> <td>0</td> </tr> <tr> <td><b>Action Modify</b></td> <td>1</td> </tr> <tr> <td><b>Action Delete</b></td> <td>2</td> </tr> </tbody> </table>	Action Type	Value	<b>Action Add</b>	0	<b>Action Modify</b>	1	<b>Action Delete</b>	2																
Action Type	Value																									
<b>Action Add</b>	0																									
<b>Action Modify</b>	1																									
<b>Action Delete</b>	2																									
<b>INSTRUMENT LEG SIDE TYPE</b>	1	<p>Leg Side Type contains the following values:</p> <table border="1"> <thead> <tr> <th>Leg Side Type</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>Side Buy</b></td> <td>0</td> </tr> <tr> <td><b>Side Sell</b></td> <td>1</td> </tr> <tr> <td><b>Side NULL</b></td> <td>2</td> </tr> </tbody> </table>	Leg Side Type	Value	<b>Side Buy</b>	0	<b>Side Sell</b>	1	<b>Side NULL</b>	2																
Leg Side Type	Value																									
<b>Side Buy</b>	0																									
<b>Side Sell</b>	1																									
<b>Side NULL</b>	2																									
<b>INSTRUMENT LOT TYPE</b>	1	<p>Instrument Lot Type contains the following values:</p> <table border="1"> <thead> <tr> <th>Lot Type</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>Minimum Order Entry Quantity</b></td> <td>0</td> </tr> <tr> <td><b>Minimum Block Trade Quantity</b></td> <td>1</td> </tr> <tr> <td><b>Round Lot</b></td> <td>2</td> </tr> </tbody> </table>	Lot Type	Value	<b>Minimum Order Entry Quantity</b>	0	<b>Minimum Block Trade Quantity</b>	1	<b>Round Lot</b>	2																
Lot Type	Value																									
<b>Minimum Order Entry Quantity</b>	0																									
<b>Minimum Block Trade Quantity</b>	1																									
<b>Round Lot</b>	2																									
<b>INSTRUMENT SECURITY TRADING STATUS</b>	1	<p>Instrument Security Trading Status contains the following values:</p> <table border="1"> <thead> <tr> <th>Trading Status</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>Status Unknown</b></td> <td>0</td> </tr> <tr> <td><b>Status Trading Halt</b></td> <td>1</td> </tr> <tr> <td><b>Status Close</b></td> <td>2</td> </tr> <tr> <td><b>Status New Price Indication</b></td> <td>3</td> </tr> <tr> <td><b>Status Ready To Trade</b></td> <td>4</td> </tr> <tr> <td><b>Status Not Available for Trading</b></td> <td>5</td> </tr> <tr> <td><b>Status Pre Open</b></td> <td>6</td> </tr> <tr> <td><b>Status Pre Cross</b></td> <td>7</td> </tr> <tr> <td><b>Status Cross</b></td> <td>8</td> </tr> <tr> <td><b>Status Post Close</b></td> <td>9</td> </tr> <tr> <td><b>Status No Change</b></td> <td>10</td> </tr> </tbody> </table>	Trading Status	Value	<b>Status Unknown</b>	0	<b>Status Trading Halt</b>	1	<b>Status Close</b>	2	<b>Status New Price Indication</b>	3	<b>Status Ready To Trade</b>	4	<b>Status Not Available for Trading</b>	5	<b>Status Pre Open</b>	6	<b>Status Pre Cross</b>	7	<b>Status Cross</b>	8	<b>Status Post Close</b>	9	<b>Status No Change</b>	10
Trading Status	Value																									
<b>Status Unknown</b>	0																									
<b>Status Trading Halt</b>	1																									
<b>Status Close</b>	2																									
<b>Status New Price Indication</b>	3																									
<b>Status Ready To Trade</b>	4																									
<b>Status Not Available for Trading</b>	5																									
<b>Status Pre Open</b>	6																									
<b>Status Pre Cross</b>	7																									
<b>Status Cross</b>	8																									
<b>Status Post Close</b>	9																									
<b>Status No Change</b>	10																									
<b>REFRESH</b>	4	The flags indicate which data within the Refresh Intraday data structure is																								

<b>INTRADAY FLAGS</b>		present. The flags values are OR'ed into flags field, and have to be checked for presence of data.																																								
		The flags consists of the following values:																																								
		<table border="1"> <thead> <tr> <th>Flag</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>Last Condition</td><td>0x1</td></tr> <tr><td>Quote Condition</td><td>0x2</td></tr> <tr><td>Open Price</td><td>0x4</td></tr> <tr><td>Last Price</td><td>0x8</td></tr> <tr><td>High Price</td><td>0x10</td></tr> <tr><td>Low Price</td><td>0x20</td></tr> <tr><td>Close Price</td><td>0x40</td></tr> <tr><td>Previous Close Price</td><td>0x80</td></tr> <tr><td>After Market Close Price</td><td>0x100</td></tr> <tr><td>Bid Price</td><td>0x200</td></tr> <tr><td>Ask Price</td><td>0x400</td></tr> <tr><td>Last Exchange</td><td>0x800</td></tr> <tr><td>Bid Exchange</td><td>0x1000</td></tr> <tr><td>Ask Exchange</td><td>0x2000</td></tr> <tr><td>Bid Size</td><td>0x4000</td></tr> <tr><td>Ask Size</td><td>0x8000</td></tr> <tr><td>Last Size</td><td>0x10000</td></tr> <tr><td>Volume</td><td>0x20000</td></tr> <tr><td>Pre Market Open Price</td><td>0x40000</td></tr> </tbody> </table>	Flag	Value	Last Condition	0x1	Quote Condition	0x2	Open Price	0x4	Last Price	0x8	High Price	0x10	Low Price	0x20	Close Price	0x40	Previous Close Price	0x80	After Market Close Price	0x100	Bid Price	0x200	Ask Price	0x400	Last Exchange	0x800	Bid Exchange	0x1000	Ask Exchange	0x2000	Bid Size	0x4000	Ask Size	0x8000	Last Size	0x10000	Volume	0x20000	Pre Market Open Price	0x40000
Flag	Value																																									
Last Condition	0x1																																									
Quote Condition	0x2																																									
Open Price	0x4																																									
Last Price	0x8																																									
High Price	0x10																																									
Low Price	0x20																																									
Close Price	0x40																																									
Previous Close Price	0x80																																									
After Market Close Price	0x100																																									
Bid Price	0x200																																									
Ask Price	0x400																																									
Last Exchange	0x800																																									
Bid Exchange	0x1000																																									
Ask Exchange	0x2000																																									
Bid Size	0x4000																																									
Ask Size	0x8000																																									
Last Size	0x10000																																									
Volume	0x20000																																									
Pre Market Open Price	0x40000																																									
<b>SESSION TYPE</b>	1	Session type consists of the following types:																																								
		<table border="1"> <thead> <tr> <th>Session Type</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>Regular Market Session</td><td>0</td></tr> <tr><td>Pre Market Session</td><td>1</td></tr> <tr><td>After Market Session</td><td>2</td></tr> <tr><td>Unknown</td><td>255</td></tr> </tbody> </table>	Session Type	Value	Regular Market Session	0	Pre Market Session	1	After Market Session	2	Unknown	255																														
Session Type	Value																																									
Regular Market Session	0																																									
Pre Market Session	1																																									
After Market Session	2																																									
Unknown	255																																									
<b>SESSION STATE</b>	1	Session state consists of the following values:																																								
		<table border="1"> <thead> <tr> <th>Session State</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>Day Open State</td><td>0</td></tr> <tr><td>Day Close State</td><td>1</td></tr> <tr><td>Open State</td><td>2</td></tr> <tr><td>Close State</td><td>3</td></tr> <tr><td>Halt State</td><td>4</td></tr> <tr><td>Resume State</td><td>5</td></tr> <tr><td>Unknown</td><td>255</td></tr> </tbody> </table>	Session State	Value	Day Open State	0	Day Close State	1	Open State	2	Close State	3	Halt State	4	Resume State	5	Unknown	255																								
Session State	Value																																									
Day Open State	0																																									
Day Close State	1																																									
Open State	2																																									
Close State	3																																									
Halt State	4																																									
Resume State	5																																									
Unknown	255																																									

<b>SOURCE</b>	1	Source of originating market, consists of the following codes:																																						
<table border="1"> <thead> <tr> <th>Source Code</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>Unknown</td><td>0</td></tr> <tr><td>CME</td><td>1</td></tr> <tr><td>CME NYMEX</td><td>2</td></tr> <tr><td>CME COMEX</td><td>3</td></tr> <tr><td>CME CBOT</td><td>4</td></tr> <tr><td>CME KCBT</td><td>5</td></tr> <tr><td>CTA</td><td>20</td></tr> <tr><td>CTA NYSE</td><td>21</td></tr> <tr><td>CTA NYSE MKT</td><td>22</td></tr> <tr><td>UTP NASDAQ</td><td>23</td></tr> <tr><td>OTC MARKETS</td><td>24</td></tr> <tr><td>OPRA</td><td>26</td></tr> <tr><td>CBOE CSMI INDEX</td><td>50</td></tr> <tr><td>NASDAQ ITCH</td><td>70</td></tr> <tr><td>FOREX</td><td>80</td></tr> <tr><td>CALC INDEX</td><td>100</td></tr> <tr><td>S&amp;P DJI INDEX</td><td>101</td></tr> <tr><td>NASDAQ GIDS INDEX</td><td>102</td></tr> </tbody> </table>			Source Code	Value	Unknown	0	CME	1	CME NYMEX	2	CME COMEX	3	CME CBOT	4	CME KCBT	5	CTA	20	CTA NYSE	21	CTA NYSE MKT	22	UTP NASDAQ	23	OTC MARKETS	24	OPRA	26	CBOE CSMI INDEX	50	NASDAQ ITCH	70	FOREX	80	CALC INDEX	100	S&P DJI INDEX	101	NASDAQ GIDS INDEX	102
Source Code	Value																																							
Unknown	0																																							
CME	1																																							
CME NYMEX	2																																							
CME COMEX	3																																							
CME CBOT	4																																							
CME KCBT	5																																							
CTA	20																																							
CTA NYSE	21																																							
CTA NYSE MKT	22																																							
UTP NASDAQ	23																																							
OTC MARKETS	24																																							
OPRA	26																																							
CBOE CSMI INDEX	50																																							
NASDAQ ITCH	70																																							
FOREX	80																																							
CALC INDEX	100																																							
S&P DJI INDEX	101																																							
NASDAQ GIDS INDEX	102																																							
<b>SYMBOL or LONG_SYMBOL</b>	14 or 35	Identifies the instrument symbol. Symbol consists of the following data fields:																																						
<table border="1"> <thead> <tr> <th>Field Name</th> <th>Type</th> <th>Length (bytes)</th> </tr> </thead> <tbody> <tr><td>Symbol Body</td><td>UInt8</td><td>11 or 32</td></tr> <tr><td>Symbol Type</td><td>UInt8</td><td>1</td></tr> <tr><td>Exchange Type</td><td>UInt8</td><td>1</td></tr> <tr><td>Country Type</td><td>UInt8</td><td>1</td></tr> </tbody> </table>			Field Name	Type	Length (bytes)	Symbol Body	UInt8	11 or 32	Symbol Type	UInt8	1	Exchange Type	UInt8	1	Country Type	UInt8	1																							
Field Name	Type	Length (bytes)																																						
Symbol Body	UInt8	11 or 32																																						
Symbol Type	UInt8	1																																						
Exchange Type	UInt8	1																																						
Country Type	UInt8	1																																						
<p>Symbol Body contains different data fields for different symbol types. The symbol body is left-aligned and padded with 0's to the right. Some numeric data fields use partial bits within the same data type, reducing footprint of the data structure.</p>																																								
<p><b>1. Equity/Index/Bond/Mutual Fund/Currency/Future Spread Instrument</b></p>																																								
<table border="1"> <thead> <tr> <th>Field Name</th> <th>Type</th> <th>Length (bytes)</th> </tr> </thead> <tbody> <tr><td>Symbol Body</td><td>UInt8</td><td>11 or 32</td></tr> </tbody> </table>			Field Name	Type	Length (bytes)	Symbol Body	UInt8	11 or 32																																
Field Name	Type	Length (bytes)																																						
Symbol Body	UInt8	11 or 32																																						
<p>Alphanumeric text specific to an instrument simply fills the symbol body for these types of instruments.</p>																																								

## 2. Option Instrument

Field Name	Type	Length
<b>Symbol Body</b>	UInt8	5 or 26 bytes
<b>Strike Price</b>	UInt32 (1)	27 bits
<b>Expiration Value</b>	UInt32 (1)	5 bits
<b>Expiration Year</b>	UInt8 (2)	6 bits
<b>Option Type</b>	UInt8 (2)	1 bit
<b>Expiration Value Type</b>	UInt8 (2)	1 bit
<b>Expiration Month</b>	UInt8 (3)	4 bits
<b>Strike Price Precision</b>	UInt8 (3)	4 bits

Option Strike Price Precision indicates the decimal location for the Strike Price, and consists of the following codes:

Strike Price Precision	Value
<b>Whole</b>	0
<b>Decimal 1</b>	1
<b>Decimal 2</b>	2
<b>Decimal 3</b>	3
<b>Decimal 4</b>	4
<b>Decimal 5</b>	5
<b>Decimal 6</b>	6
<b>Decimal 7</b>	7
<b>Negative Whole</b>	8
<b>Negative Decimal 1</b>	9
<b>Negative Decimal 2</b>	10
<b>Negative Decimal 3</b>	11
<b>Negative Decimal 4</b>	12
<b>Negative Decimal 5</b>	13
<b>Negative Decimal 6</b>	14
<b>Negative Decimal 7</b>	15

Option Type indicates whether the option is a call or a put, and consists of the following values:

Option Type	Value
<b>Call</b>	0
<b>Put</b>	1

Expiration Value indicates how to use Expiration Value, and consists of Day or Week types. If Expiration Value Type is Day, then Expiration Value field should be used as a day. In case if Expiration Value Type is a Week, then Expiration Value field should indicate

week number, ie, week 1, week 2, etc., with maximum week number 5.

Expiration Value Type	Value
Day	0
Week	1

Expiration Year contains year starting from 2000. For example, year 2017 is set to 17.

Expiration Month contains months with range of 1-12.

### 3. Future Instrument

Field Name	Type	Length
Symbol Body	UInt8	9 or 30 bytes
Expiration Day	UInt16 (1)	6 bits
Expiration Year	UInt16 (1)	4 bits
Expiration Month	UInt16 (1)	5 bits

Expiration Year contains year starting from 2000. For example, year 2017 is set to 17.

Expiration Month contains months with range of 1-12.

Expiration Day is set to expiry date, or 0 if the exchange does not supply one.

SymbolType is an alphanumeric value, and consists of the following types:

SymbolType	Value
Equity	S
Index	I
Equity Option	O
Bond	B
Mutual Fund	M
Currency	C
Future	F
Future Option	P
Future Spread	D

See Exchange Type field description for Exchange Type.

CountryType is an alphanumeric value, and consists of the following types:



		Country Type	Value
		United States	U
		Canada	C
		International	I

<b>TRADE</b>	1	Trade conditions map to exchange-specific trade conditions. The conditions consist of the following values:
<b>CONDITINONS</b>		

Trade Condition	Value
Regular Trade	0
Acquisition Trade	1
Average Price Trade	2
Automatic Execution Trade	3
Bunched Trade	4
Bunch Sold Trade	5
CAP Election Trade	6
Cash Trade	7
Closing Trade	8
Cross Trade	9
Derivatively Price Trade	10
Distribution Trade	11
Form T Trade	12
Form T Out of Sequence Trade	13
Inter Market Sweep Trade	14
Market Center Official Close Trade	15
Market Center Official Open Trade	16
Market Center Opening Trade	17
Market Center Re Opening Trade	18
Market Center Closing Trade	19
Next Day Trade	20
Price Variation Trade	21
Prior Reference Price Trade	22
NYSE Rule 155 Trade	23
NYSE Rule 127 Trade	24
Opening Trade	25
Opened Trade	26
Regular Stopped Stock Trade	27
Re Opening Trade	28
Seller Trade	29
Sold Last Trade	30
Sold Last Stopped Stock Trade	31
Sold Out of Sequence Trade	32
Sold Out of Sequence Stopped Stock Trade	33

		<b>Split Trade</b>	34
		<b>Stock Option Trade</b>	35
		<b>Yellow Flag Trade</b>	36
		<b>Odd Lot Trade</b>	37
		<b>Corrected Consolidated Close Price</b>	38
		<b>Unknown</b>	39
		For Equity Options, trade conditions map to the following values:	
		<b>Option Trade Condition</b>	<b>Value</b>
		<b>Regular</b>	0
		<b>Canc</b>	1
		<b>Oseq</b>	2
		<b>Cncl</b>	3
		<b>Late</b>	4
		<b>Cnco</b>	5
		<b>Open</b>	6
		<b>Cnol</b>	7
		<b>Opnl</b>	8
		<b>Auto</b>	9
		<b>Reop</b>	10
		<b>Ajst</b>	11
		<b>Sprd</b>	12
		<b>StdI</b>	13
		<b>Stpd</b>	14
		<b>Cstp</b>	15
		<b>Bwrt</b>	16
		<b>Cmbo</b>	17
		<b>Spim</b>	18
		<b>Isoi</b>	19
		<b>Bnmt</b>	20
		<b>Xmpt</b>	21
<b>TRADE FLAGS</b>	4	Trade flags are processed and set by MarketIfs regardless of exchanges' flags, and indicate a presence of a certain condition within a trade message. All trade flags are OR'ed into the flag. The flag consists of the following values:	
		<b>Trade Flag</b>	<b>Value</b>
		<b>Regular Market Last Price</b>	0x1
		<b>Regular Market Volume</b>	0x2
		<b>Regular Market High</b>	0x4
		<b>Regular Market Low</b>	0x8
		<b>Day Market High</b>	0x10
		<b>Day Market Low</b>	0x20

		<table border="1"> <tr> <td><b>Extended Market Last Price</b></td> <td>0x40</td> </tr> <tr> <td><b>Pre Market Volume</b></td> <td>0x80</td> </tr> <tr> <td><b>After Market Volume</b></td> <td>0x100</td> </tr> <tr> <td><b>Close Price</b></td> <td>0x200</td> </tr> <tr> <td><b>Open Price</b></td> <td>0x400</td> </tr> </table>	<b>Extended Market Last Price</b>	0x40	<b>Pre Market Volume</b>	0x80	<b>After Market Volume</b>	0x100	<b>Close Price</b>	0x200	<b>Open Price</b>	0x400
<b>Extended Market Last Price</b>	0x40											
<b>Pre Market Volume</b>	0x80											
<b>After Market Volume</b>	0x100											
<b>Close Price</b>	0x200											
<b>Open Price</b>	0x400											
<b>VOLUME FLAGS</b>	4	<p>Volume flags indicate conditions whether the volume can be used or not in specific trading sessions. Volume flags are individual bits set by Marketifs, and should be checked by performing a bitwise AND. The flag consists of the following values:</p> <table border="1"> <thead> <tr> <th>Volume Flag</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td><b>Regular Market Volume</b></td> <td>0x1</td> </tr> <tr> <td><b>Pre Market Volume</b></td> <td>0x2</td> </tr> <tr> <td><b>After Market Volume</b></td> <td>0x4</td> </tr> </tbody> </table>	Volume Flag	Value	<b>Regular Market Volume</b>	0x1	<b>Pre Market Volume</b>	0x2	<b>After Market Volume</b>	0x4		
Volume Flag	Value											
<b>Regular Market Volume</b>	0x1											
<b>Pre Market Volume</b>	0x2											
<b>After Market Volume</b>	0x4											