



Cboe Options Exchange FLEX Feed Specification

Version 1.0.9

October 11, 2019

Contents

| | | |
|----------|----------------------------------------------------|-----------|
| 1 | Introduction | 5 |
| 1.1 | Overview | 5 |
| 1.2 | Feed Connectivity Requirements | 5 |
| 1.3 | FLEX Specific Symbol Processing | 7 |
| 1.4 | Gap Request Proxy and Message Retransmission | 7 |
| 1.5 | Spin Servers..... | 8 |
| 2 | Protocol | 9 |
| 2.1 | Message Format | 9 |
| 2.2 | Data Types | 9 |
| 2.3 | Message Framing..... | 10 |
| 2.4 | Sequenced Unit Header..... | 10 |
| 2.5 | Heartbeat Messages..... | 10 |
| 3 | FLEX Feed Messages | 11 |
| 3.1 | Time | 11 |
| 3.2 | FLEX Instrument Definition..... | 11 |
| 3.3 | Complex FLEX Instrument Definition | 12 |
| 3.4 | Auction Notification | 13 |
| 3.5 | Auction Cancel..... | 14 |
| 3.6 | Auction Trade | 15 |
| 3.7 | Trade..... | 15 |
| 3.8 | Trade Break | 16 |
| 3.9 | Trading Status | 17 |
| 3.10 | End of Session | 17 |
| 3.11 | Login | 18 |
| 3.12 | Login Response | 18 |
| 3.13 | Gap Request | 20 |
| 3.14 | Gap Response..... | 20 |
| 4 | Spin Messages..... | 21 |
| 4.1 | Login | 21 |
| 4.2 | Login Response | 21 |
| 4.3 | Spin Image Available..... | 21 |
| 4.4 | Spin Request..... | 21 |
| 4.5 | Spin Response | 22 |
| 4.6 | Spin Finished | 22 |
| 4.7 | Instrument Definition Request | 22 |
| 4.8 | Instrument Definition Response..... | 23 |
| 4.9 | Instrument Definition Finished..... | 23 |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| | | |
|----------|-------------------------------------------------------------------|-----------|
| 5 | Message Types | 24 |
| 5.1 | Gap Request Proxy Messages | 24 |
| 5.2 | Spin Server Messages..... | 24 |
| 5.3 | FLEX Feed Messages..... | 24 |
| 6 | Example Messages..... | 25 |
| 6.1 | Login Message | 25 |
| 6.2 | Login Response Message | 25 |
| 6.3 | Gap Request Message | 25 |
| 6.4 | Gap Response Message..... | 25 |
| 6.5 | Spin Image Available Message..... | 25 |
| 6.6 | Spin Request Message..... | 25 |
| 6.7 | Spin Response Message | 26 |
| 6.8 | Spin Finished Message | 26 |
| 6.9 | Instrument Definition Request | 26 |
| 6.10 | Instrument Definition Response..... | 26 |
| 6.11 | Instrument Definition Finished..... | 26 |
| 6.12 | Time Message | 26 |
| 6.13 | FLEX Instrument Definition Message..... | 26 |
| 6.14 | Complex FLEX Instrument Definition Message | 27 |
| 6.15 | Auction Notification Message | 28 |
| 6.16 | Auction Cancel Message..... | 28 |
| 6.17 | Auction Trade Message | 28 |
| 6.18 | Trade (long) Message | 28 |
| 6.19 | Trade (short) Message..... | 29 |
| 6.20 | Trade Break Message | 29 |
| 6.21 | Trading Status Message | 29 |
| 6.22 | End of Session | 29 |
| 7 | Multicast Configuration | 30 |
| 7.1 | Production Environment Configuration..... | 30 |
| 7.1.1 | Limitations/Configurations..... | 30 |
| 7.1.2 | Unit/Symbol Distribution..... | 31 |
| 7.1.3 | Cboe Options FLEX Multicast Routing Parameters | 32 |
| 7.1.4 | Cboe Options FLEX Multicast Addresses | 33 |
| 7.2 | Certification Environment Configuration..... | 35 |
| 7.2.1 | Unit/Symbol Distribution..... | 35 |
| 7.2.2 | Cboe Options FLEX Certification Multicast Routing Parameters..... | 36 |
| 7.2.3 | Cboe Options FLEX Certification Multicast Addresses | 36 |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

8 Connectivity 37
8.1 Supported Extranet Carriers 37
8.2 Bandwidth Recommendation..... 37

9 References 38

10 Support..... 38

1 Introduction

1.1 Overview

Note that this specification will be the standard specification used for market data for FLEX Options on the Cboe Options (“C1”) Exchange platform.

Cboe customers may use the FLEX feed to receive real-time auction updates and execution information.

Options FLEX feed cannot be used to enter orders. For order entry, refer to the US Options FIX Specification.

A WAN-Shaped version of the FLEX feed is available from both of Cboe’s datacenters. Customers may choose to take one or more of the following FLEX feed options depending on their location and connectivity to Cboe.

| Exchange | Shaping (WAN) | Served From Data Center (Primary/Secondary) | Multicast Feed ID |
|------------|---------------|---------------------------------------------|-------------------|
| C1 Options | WAN | Primary | CAF |
| C1 Options | WAN | Primary | CBF |
| C1 Options | WAN | Secondary | CEF |

1.2 Feed Connectivity Requirements

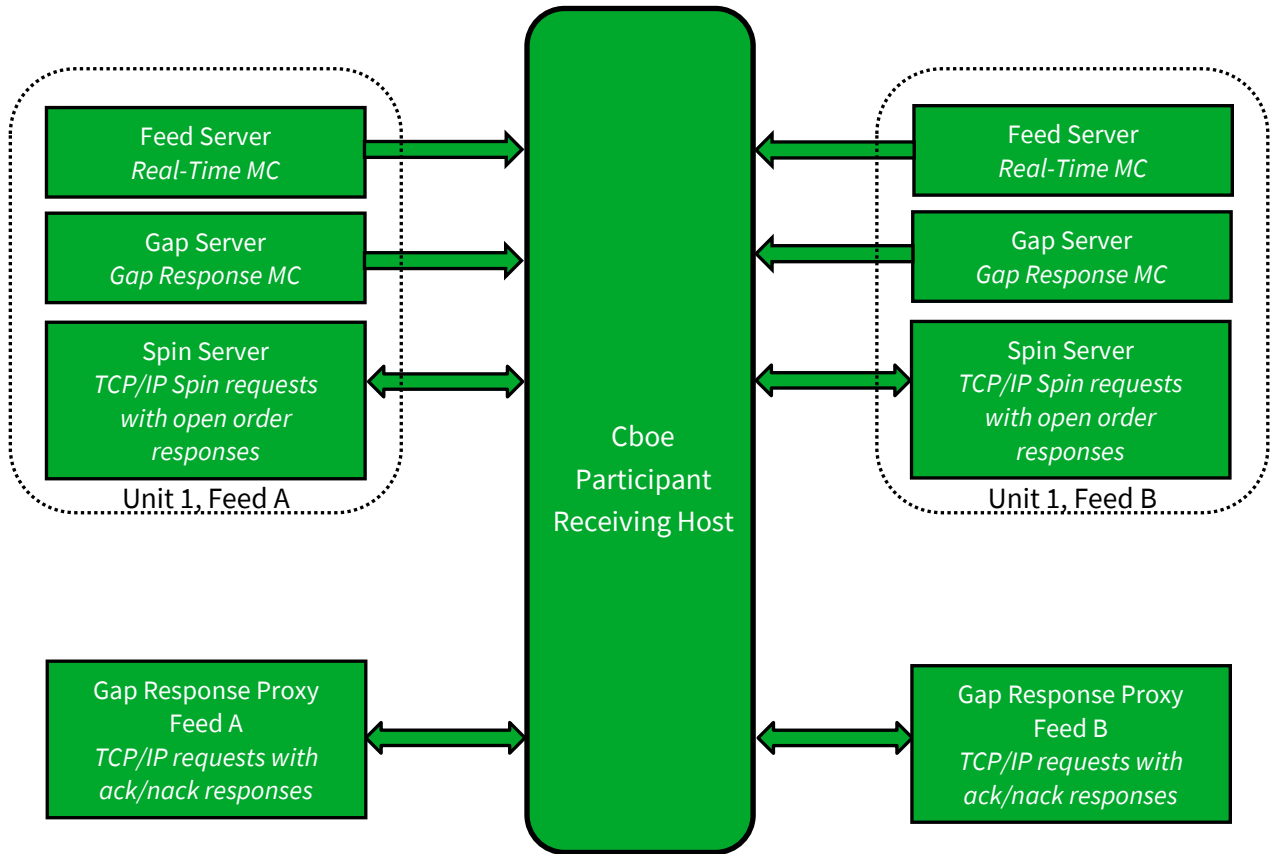
WAN-Shaped feeds are available to customers with a minimum of 100 Mb/s of connectivity to Cboe via cross connect or dedicated circuit.

Customers with sufficient connectivity may choose to take more than one WAN-Shaped feed from the Cboe datacenters. It should be noted that due to proximity, feeds from the secondary datacenter will have additional latency than those co-located with Cboe in the primary datacenter.

Cboe Options FLEX feed real-time events are delivered using a single published multicast address for all symbol ranges.

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

The following diagram is a logical representation of Options FLEX feed message flow between Cboe and a customer feed handler that is listening to the “A” and “B” instances of two units:



1.3 FLEX Specific Symbol Processing

Cboe has implemented a symbol mapping mechanism for both simple and complex instruments on the FLEX feed. This symbol mapping significantly reduces the size of the FLEX feed and allows participants to use the same symbol handling mechanisms for the Cboe operated equity, options, and futures exchanges. Refer to the `FLEX Instrument Definition` and `Complex FLEX Instrument Definition` messages for more information.

The `FLEX Instrument Definition` or `Complex FLEX Instrument Definition` messages are sequenced messages that are sent when an instrument is created. If a user misses a mapping message, then that message can be recovered through either the GRP or SPIN Server.

1.4 Gap Request Proxy and Message Retransmission

Establishing a TCP connection to a Gap Request Proxy (“GRP”) port allows for delivery of missed sequenced data. This GRP port is specific to FLEX feed and is NOT shared with any other Cboe Options market data feed. Participants who do not wish to request missed messages do not need to connect to a GRP port for any reason or listen to the multicast addresses reserved for message retransmission. Participants choosing to request missed data will need to connect to their assigned GRP port, log in, and request gap ranges as necessary. All gap requests will be responded to with a `Gap Response` message. A `Gap Response Status` code of ‘A’accepted signals that the replayed messages will be delivered via the appropriate gap response multicast address. Any other `Gap Response Status` code will indicate the reason that the request cannot be serviced.

Gap requests are limited in message count, frequency, and age by the GRP. Gap requests will only be serviced if they are within a defined sequence range of the current multicast sequence number for the requested unit. Participants will receive a total daily allowance of gap requested messages. In addition, each participant is given renewable one second and one minute gap request limits.

If more than one gap request is received for a particular unit/sequence/count combination within a short timeframe, all requests will receive a successful `Gap Response` message from the GRP, but only a single replayed message will be sent on the gap response multicast address.

If overlapping gap requests are received within a short period of time, the gap server will only send the union of the sequence ranges across grouped gap requests. Participants will receive gap responses for their requested unit/sequence/count, but receivers should be prepared for the **gap responses to be delivered via multicast in non-contiguous blocks**.

Gap acknowledgements or rejects will be delivered to users for every gap request received by the GRP. Users should be prepared to see replayed multicast data before or after the receipt of the gap response acknowledgement from the GRP.

1.5 Spin Servers

A Spin Server is available for each unit. The server allows participants to connect via TCP and receive a spin of all symbol definitions, including both FLEX Instrument Definition and Complex FLEX Instrument Definition messages, and Trading Status messages. By using the spin, a participant can get the current instrument definitions and trading status messages quickly in the middle of the trading session without worry of gap request limits. The Spin Server for each unit is assigned its own address and/or TCP port.

Upon successful login and periodically thereafter, a Spin Image Available message is sent which contains a sequence number indicating the most recent message. Using a Spin Request message, a participant may request a spin for symbol definitions and trading status messages to a sequence number noted within one of the *last ten* Spin Image Available messages distributed. If the Spin Request submitted does not present a sequence number that matches one of the last ten Spin Image Available messages distributed, the spin will return orders up to the next closest sequence number reported through a Spin Image Available message that is greater than the sequence number requested.

In the case a participant sends a sequence number in a Spin Request that is higher than the sequence number reported by the most recent Spin Image Available message, the next spin image to be generated will be returned when it is available. If the requested sequence number is still higher at that time, an “O” (Out of Range) error will be generated.

Customers can also use the Spin Server to request a spin of all Symbol Mapping messages by sending an Instrument Definition Request. The Spin Server can only process one spin at a time. Customers will need to wait for a Spin Finished or Instrument Definition Finished message before submitting another request.

2 Protocol

2.1 Message Format

The messages that make up the FLEX feed protocol are delivered using Cboe `Sequenced Unit Header` which handles sequencing and delivery integrity. All messages delivered via multicast as well will use the `Sequenced Unit Header` for handling message integrity.

All UDP delivered events will be self-contained. Developers can assume that UDP delivered data will not cross frame boundaries and a single Ethernet frame will contain only one `Sequenced Unit Header` with associated data.

This FLEX data feed is comprised of a series of dynamic length sequenced messages. Each message begins with Length and Message Type fields. Cboe reserves the right to add message types and grow the length of any message without notice. Customers should develop their decoders to deal with unknown message types and messages that grow beyond the expected length. Messages will only be grown to add additional data to the end of a message.

2.2 Data Types

The following field types are used within the feed.

- **Alphanumeric** fields are left justified ASCII fields and space padded on the right.
- **Binary** fields are unsigned and sized to “Length” bytes and ordered using Little Endian convention (least significant byte first).
- **Signed Binary** fields are signed and sized to “Length” bytes and ordered using Little Endian convention (least significant byte first).
- **Binary Signed Short Price** fields are signed Little Endian encoded 2 byte binary fields with 2 implied decimal places (denominator = 100). The short price range is -327.68 to +327.67. Prices outside of this range will use the long price.
- **Binary Signed Long Price** fields are signed Little Endian encoded 8 byte binary fields with 4 implied decimal places (denominator = 10,000).
- **Bit Field** fields are fixed width fields with each bit representing a Boolean flag (the 0 bit is the lowest significant bit; the 7 bit is the highest significant bit).
- **Printable ASCII** fields are left justified ASCII fields that are space padded on the right that may include ASCII values in the range of 0x20 – 0x7e.

2.3 Message Framing

Messages will be combined into single UDP frame where possible to decrease message overhead and total bandwidth. The count of messages in a UDP frame will be communicated using the `Sequenced Unit Header`. Framing will be determined by the server for each site. The content of the multicast across feeds (e.g. A/B WAN-Shaped) will be identical, **but framing will not be consistent across feeds**. Receiving processes that receive and arbitrate multiple feeds cannot use frame level arbitration to fill gaps.

2.4 Sequenced Unit Header

The `Sequenced Unit Header` is used for all Cboe Multicast messages.

This feed will deliver sequenced and unsequenced data using the `Sequenced Unit Header`. Unsequenced headers will have a 0 value for the sequence field and potentially for the unit field.

| Sequenced Unit Header | | | | |
|-------------------------------|--------|--------|------------|---------------------------------------------------------------------------------------------------|
| Field | Offset | Length | Value/Type | Description |
| <i>Hdr Length</i> | 0 | 2 | Binary | Length of entire block of messages. Includes this header and <i>Hdr Count</i> messages to follow. |
| <i>Hdr Count</i> | 2 | 1 | Binary | Number of messages to follow this header. |
| <i>Hdr Unit</i> | 3 | 1 | Binary | Unit that applies to messages included in this header. |
| <i>Hdr Sequence</i> | 4 | 4 | Binary | Sequence of first message to follow this header. |
| Total Length = 8 bytes | | | | |

2.5 Heartbeat Messages

The `Sequenced Unit Header` with a count field set to “0” will be used for heartbeat messages. During trading hours heartbeat messages will be sent from all multicast addresses if no data has been delivered within 1 second. Heartbeat messages never increment the sequence number.

Outside of trading hours Cboe sends heartbeat messages on all real-time channels with a sequence of “0” to help users validate multicast connectivity. Heartbeat messages may not be sent from 12:00 a.m. – 1:00 a.m. ET or during maintenance windows.

3 FLEX Feed Messages

3.1 Time

A `Time` message is immediately generated and sent when there is a FLEX event for a given clock second. If there is no FLEX event for a given clock second, then no `Time` message is sent for that second. All subsequent time offset fields for the same unit will use the new `Time` value as the base until another `Time` message is received for the same unit.

| Time | | | | |
|-------------------------------|--------|--------|--------------|----------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0x20 | <code>Time</code> Message |
| <i>Time</i> | 2 | 4 | Binary | Number of whole seconds from midnight Eastern Time |
| Total Length = 6 bytes | | | | |

3.2 FLEX Instrument Definition

A `FLEX Instrument Definition` message represents an instrument that is available to place orders. It is sent as a sequenced message.

| Flex Instrument Definition | | | | |
|----------------------------|--------|--------|-----------------|-------------------------------------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x9C | <code>Flex Instrument Definition</code> Message |
| <i>Time Offset</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp. |
| <i>Feed Symbol</i> | 6 | 6 | Printable ASCII | <i>Symbol</i> right padded with spaces. |
| <i>OSI Root</i> | 12 | 6 | Printable ASCII | OSI Root right padded with spaces. |
| <i>Year</i> | 18 | 2 | Alphanumeric | Two digit year |
| <i>Month</i> | 20 | 2 | Alphanumeric | Two digit month |
| <i>Day</i> | 22 | 2 | Alphanumeric | Two digit day |
| <i>Call Put Indicator</i> | 24 | 1 | Alphanumeric | C = Call P = Put |
| <i>Dollar Strike</i> | 25 | 5 | Alphanumeric | Dollar strike (if not percentage) left padded with zero. If percentage, field is space padded. |
| <i>Decimal Strike</i> | 30 | 3 | Alphanumeric | Decimal strike (if not percentage) left padded with zero. If percentage, field is space padded. |
| <i>Symbol Condition</i> | 33 | 1 | Alphanumeric | N = Normal C = Closing Only |
| <i>Underlying</i> | 34 | 8 | Printable ASCII | Instrument Underlying right padded with spaces. |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| | | | | |
|--------------------------------|----|---|--------------|-------------------------------------------------------------------------------------------------------|
| <i>Exercise Style</i> | 42 | 1 | Alphanumeric | A = American E = European Exercise style is always European for Asian and Cliquet. |
| <i>Settlement Type</i> | 43 | 1 | Alphanumeric | A = AM P = PM S = Asian Q = Cliquet |
| <i>Percentage</i> | 44 | 4 | Binary | Percentage when strike and price is percentage based (implied 4 decimal places). Otherwise zero. |
| <i>Observation Day</i> | 48 | 2 | Alphanumeric | Two digit observation day of month. Valid only for Asian and Cliquet options, otherwise spade padded. |
| <i>Return Cap Percentage</i> | 50 | 4 | Binary | Return cap percentage (implied 2 decimal places). Valid only for Cliquet, otherwise zero. |
| <i>Creation Day</i> | 54 | 2 | Alphanumeric | Two digit creation day of the month. Valid only for Cliquet, otherwise space padded. |
| <i>Bit Fields</i> | 56 | 1 | Bit Field | Bit 0: If set, strike/price in percentage. Bit 1-7: Reserved |
| Total Length = 57 bytes | | | | |

3.3 Complex FLEX Instrument Definition

A Complex FLEX Instrument Definition message represents a complex instrument that is available to place orders. It is sent as a sequenced message.

The Complex FLEX Instrument Definition message will contain two or more repeating groups of leg definitions. There is a limit of 100 leg definitions. For complex instruments that contain over 17 legs, the complex instrument will be split across multiple messages as indicated by the Message Count.

| Complex FLEX Instrument Definition | | | | |
|------------------------------------|--------|--------|-----------------|---------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x9B | Complex FLEX Instrument Definition Message |
| <i>Time offset</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp. |
| <i>Complex Instrument ID</i> | 6 | 6 | Printable ASCII | Complex Instrument ID right padded with spaces. |
| <i>Underlying</i> | 12 | 8 | Printable ASCII | Complex Instrument Underlying right padded with spaces. |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| | | | | |
|-----------------------------------------------------------------------------------------------------------------------------|------------------------|---|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Complex Instrument Type</i> | 20 | 4 | Alphanumeric | 4 character field; each field describes a characteristic. Character 1: Complex Option Type X = All legs are FLEX options Characters 2-4: Reserved |
| <i>Leg Count</i> | 24 | 1 | Binary | The number of legs in the complex instrument. The maximum number of legs is 100. |
| <i>Message Count</i> | 25 | 1 | Binary | Number of messages required to define the complex instrument. |
| <i>Message Number</i> | 26 | 1 | Binary | Message number of the message count, e.g. 1 of 5. |
| <i>Message Leg Count</i> | 27 | 1 | Binary | The number of legs in this message. |
| The following fields repeat <i>Message Leg Count</i> times. The <i>Leg Index</i> is from 0 to <i>Message Leg Count</i> - 1. | | | | |
| <i>Leg Symbol</i> | 28 + 13 * Leg Index | 8 | Printable ASCII | FLEX Symbol, right padded with spaces. |
| <i>Leg Ratio</i> | 36 + 13 * Leg Index | 4 | Signed Binary | Leg ratio (positive for buy-side, negative for sell-side). |
| <i>Leg Security Type</i> | 40 + 13 * Leg Index | 1 | Alphanumeric | The instrument type of this leg. X = FLEX option |
| Total Length = 28 + (13 * Leg Count) bytes | | | | |

3.4 Auction Notification

Auction Notification messages are used to disseminate order details of a FLEX Auction. FLEX Auctions will be available for a defined period of time known as the exposure period.

| Auction Notification | | | | |
|---------------------------|--------|--------|-----------------|-------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | Length of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0xAD | Auction Notification Message |
| <i>Time offset</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp. |
| <i>FLEX Instrument ID</i> | 6 | 6 | Printable ASCII | FLEX Instrument ID right padded with spaces. |
| <i>Auction ID</i> | 12 | 8 | Binary | Day specific identifier assigned to this auction. |
| <i>Auction Type</i> | 20 | 1 | Alphanumeric | B = AIM F = FLEX Auction S = Solicitation Auction Mechanism |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| | | | | |
|--------------------------------|----|---|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Side</i> | 21 | 1 | Alphanumeric | B = Buy S = Sell |
| <i>Price</i> | 22 | 8 | Binary Signed Long Price | Auction price. Price will be zero for <i>Auction Type</i> = B and F. Price will be included for <i>Auction Type</i> = S. |
| <i>Quantity</i> | 30 | 4 | Binary | Instrument quantity. |
| <i>Customer Indicator</i> | 34 | 1 | Alphanumeric | N = Non-Customer C = Customer FLEX Auctions (<i>Auction Type</i> = F) will always have a blank <i>Customer Indicator</i> value. AIM auctions will have a blank <i>Customer Indicator</i> for simple options while AIM auctions for complex options series will include a <i>Customer Indicator</i> . |
| <i>ParticipantID</i> | 35 | 4 | Alphanumeric | Executing Broker (optional) of firm attributed to this auction. |
| <i>Auction End Offset</i> | 39 | 4 | Binary | Nanosecond offset from last timestamp. |
| <i>Client ID</i> | 43 | 4 | Alphanumeric | User defined identifier for quote attribution. |
| Total Length = 47 bytes | | | | |

3.5 Auction Cancel

Auction Cancel messages are used to disseminate the cancellation of an earlier Auction Notification message as a result of a user cancellation of the original auction, a user modification request to change the auction price or increase the original auction quantity, or to cancel any remaining auction quantity from the original Auction Notification following an auction termination.

A user request to modify the auction price or to increase the original auction quantity will result in a cancellation of the auction followed by a new Auction Notification message. Auction Cancel messages will not be issued for auction quantity decrements.

| Auction Cancel | | | | |
|--------------------------------|--------|--------|--------------|--------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | Length of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0xAE | Auction Cancel Message |
| <i>Time offset</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp |
| <i>Auction ID</i> | 6 | 8 | Binary | Day specific identifier assigned to this auction |
| Total Length = 14 bytes | | | | |

3.6 Auction Trade

`Auction Trade` messages are used to disseminate executions resulting from an auction.

| Auction Trade | | | | |
|--------------------------------|--------|--------|--------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | Length of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0xAF | <code>Auction Trade</code> Message |
| <i>Time offset</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp |
| <i>Auction ID</i> | 6 | 8 | Binary | Day specific identifier assigned to this auction |
| <i>Execution ID</i> | 14 | 8 | Binary | Day specific identifier assigned to this execution |
| <i>Price</i> | 22 | 8 | Binary Signed Long Price | Trade price. If instrument trades in percentage terms, then this represents the traded percentage with four implied decimals. |
| <i>Quantity</i> | 30 | 4 | Binary | Instrument quantity traded |
| Total Length = 34 bytes | | | | |

3.7 Trade

The `Trade` message provides information about executions of FLEX orders on the Cboe Options trading floor. `Trade` messages are necessary to calculate Cboe execution-based data.

No `Add Order` message is sent for FLEX Options orders, and thus, no order modification messages may be sent when FLEX Options are executed. Instead, a `Trade` message is sent whenever a FLEX Options order is executed in whole or in part. A complete view of all Cboe FLEX executions can be built by combining all `Auction Trade` and `Trade` messages.

| Trade (long) | | | | |
|---------------------------|--------|--------|--------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0x2A | <code>Trade</code> Message (long) |
| <i>Time offset</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp |
| <i>Order ID</i> | 6 | 8 | Binary | <i>Order ID</i> of the executed order. |
| <i>Side Indicator</i> | 14 | 1 | Alphanumeric | Always "B" |
| <i>Quantity</i> | 15 | 4 | Binary | Instrument quantity traded |
| <i>FLEX Instrument ID</i> | 19 | 6 | Printable ASCII | <i>FLEX Instrument ID</i> right padded with spaces. |
| <i>Price</i> | 25 | 8 | Binary Signed Long Price | Trade price. If instrument trades in percentage terms, then this represents the traded percentage with four implied decimals. |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| | | | | |
|--------------------------------|----|---|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Execution ID</i> | 33 | 8 | Binary | Cboe generated day-unique execution identifier of this trade. <i>Execution ID</i> is also referenced in the <code>Trade Break</code> message. |
| <i>Trade Condition</i> | 41 | 1 | Alphanumeric | (Space): Normal Trade L: Late Trade (C1 Only) |
| Total Length = 42 bytes | | | | |

| Trade (short) | | | | |
|--------------------------------|--------|--------|---------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0x2B | Trade Message (short) |
| <i>Time offset</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp |
| <i>Order ID</i> | 6 | 8 | Binary | <i>Order ID</i> of the executed order. |
| <i>Side Indicator</i> | 14 | 1 | Alphanumeric | Always "B" |
| <i>Quantity</i> | 15 | 2 | Binary | Instrument quantity traded |
| <i>FLEX Instrument ID</i> | 17 | 6 | Printable ASCII | <i>FLEX Instrument ID</i> right padded with spaces. |
| <i>Price</i> | 23 | 2 | Binary Signed Short Price | Trade price. If instrument trades in percentage terms, then this represents the traded percentage with four implied decimals. |
| <i>Execution ID</i> | 25 | 8 | Binary | Cboe generated day-unique execution identifier of this trade. <i>Execution ID</i> is also referenced in the <code>Trade Break</code> message. |
| <i>Trade Condition</i> | 33 | 1 | Alphanumeric | (Space): Normal Trade L: Late Trade (C1 Only) |
| Total Length = 34 bytes | | | | |

3.8 Trade Break

The `Trade Break` message is sent whenever a FLEX options execution on Cboe is broken. Trade breaks are rare and only affect applications that rely upon Cboe execution-based data.

| Trade Break | | | | |
|--------------------------------|--------|--------|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0x2C | <code>Trade Break</code> Message |
| <i>Time offset</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp |
| <i>Execution ID</i> | 6 | 8 | Binary | Cboe execution identifier of the execution that was broken. <i>Execution ID</i> refers to previously sent <code>Auction Trade</code> or <code>Trade</code> message. |
| Total Length = 14 bytes | | | | |

3.9 Trading Status

The `Trading Status` message is used to indicate the current trading status of a FLEX options instrument. A `Trading Status` message will be sent whenever a FLEX options instrument trading status changes.

A `Trading Status` message will be sent for all FLEX options instruments where the underlying security is Halted, Trading or Quoting.

Starting at 7:30 a.m. ET, Cboe will send a *Trading Status* of “Q” once orders can be accepted for queuing in preparation for the market open. Sometime after 9:30 a.m. ET, Cboe will send a *Trading Status* of “T” as series are open for trading on the Cboe platform. Note *Trading Status* of “Q” can also be explicitly disseminated during a Regulatory Halt Quoting Period.

A `Trading Status` message will also be sent:

- for a Regulatory Halt “Q”oting Period in any series where the underlying has experienced a Regulatory Halt as well as the “T”rading resumption for the same series.
- for instruments that are in a “Q”oting period for auctions.

The *Trading Status* field will be used to represent the status of the RTH Session (9:30 a.m. ET – 4:15 p.m. ET) and the *GTH Trading Status* field will be used to represent the status of the GTH Session (3 a.m. ET – 9:30 a.m. ET).

| Trading Status | | | | |
|--------------------------------|--------|--------|-----------------|---------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | Length of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0x31 | <code>Trading Status</code> message |
| <i>Time offset</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp |
| <i>Symbol</i> | 6 | 6 | Printable ASCII | <i>Symbol</i> right padded with spaces. |
| <i>Reserved</i> | 12 | 2 | Reserved | Reserved |
| <i>Trading Status</i> | 14 | 1 | Alpha | H = Halted Q = Quote-Only T = Trading |
| <i>Reserved</i> | 15 | 1 | Reserved | Reserved |
| <i>GTH Trading Status</i> | 16 | 1 | Alpha | H = Halted Q = Quote-Only T = Trading |
| <i>Reserved</i> | 17 | 1 | Reserved | Reserved |
| Total Length = 18 bytes | | | | |

3.10 End of Session

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

The `End of Session` message is sent for each unit when the unit shuts down. No more messages will be delivered for this unit, but heartbeats from the unit may be received.

| End of Session | | | | |
|-------------------------------|--------|--------|--------------|----------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0x2D | End of Session Message |
| <i>Timestamp</i> | 2 | 4 | Binary | Nanosecond offset from last unit timestamp |
| Total Length = 6 bytes | | | | |

Gap Request Proxy Messages

The following messages are used for initializing a TCP/IP connection to the Gap Request Proxy (“GRP”) and to request message retransmissions. Customers only need to implement the following messages if gap requests will be made. The following messages will not be delivered using multicast.

3.11 Login

The `Login` message is the first message sent to the GRP by a user’s process after the connection to the GRP is established. Failure to login before sending any other message type will result in the connection being dropped by the GRP.

| Login | | | | |
|--------------------------------|--------|--------|--------------|-----------------------------------------------------|
| Field | Offset | Length | Value/Type | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x01 | Login Message |
| <i>SessionSubId</i> | 2 | 4 | Alphanumeric | <i>SessionSubId</i> supplied by Cboe. |
| <i>Username</i> | 6 | 4 | Alphanumeric | <i>Username</i> supplied by Cboe. |
| <i>Filler</i> | 10 | 2 | Alphanumeric | (space filled) |
| <i>Password</i> | 12 | 10 | Alphanumeric | <i>Password</i> supplied by Cboe. |
| Total Length = 22 bytes | | | | |

3.12 Login Response

The `Login Response` message is sent by the GRP to a user’s process in response to a `Login` message. The status field is used to reflect an accepted login or the reason the session was not accepted. If login fails, the connection will be dropped after the `Login Response` message is sent.

| Login Response | | | | |
|---------------------|--------|--------|--------------|-----------------------------------------------------|
| Field | Offset | Length | Value/Type | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x02 | Login Response Message |
| <i>Status</i> | 2 | 1 | Alphanumeric | Accepted or reason for reject. |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| Total Length = 3 bytes | |
|--------------------------------------|--------------------------------------------|
| Login Response - Status Codes | |
| 'A' | Login Accepted |
| 'N' | Not authorized (Invalid Username/Password) |
| 'B' | Session in use |
| 'S' | Invalid Session |

3.13 Gap Request

The `Gap Request` message is used by a user's process to request retransmission of a sequenced message (or messages) by one of Cboe's gap servers.

| Gap Request | | | | |
|-------------------------------|--------|--------|------------|--------------------------------------------------------------|
| Field | Offset | Length | Value/Type | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x03 | <code>Gap Request</code> Message |
| <i>Unit</i> | 2 | 1 | Binary | <i>Unit</i> that the gap is requested for. |
| <i>Sequence</i> | 3 | 4 | Binary | <i>Sequence</i> of first message (lowest sequence in range). |
| <i>Count</i> | 7 | 2 | Binary | <i>Count</i> of messages requested. |
| Total Length = 9 bytes | | | | |

3.14 Gap Response

The `Gap Response` message is sent by the GRP in response to a `Gap Request` message. The *Unit* and *Sequence* fields will match the values supplied in the `Gap Request` message. A `Gap Response` message, with a Status of Accepted or reason for failure, will be sent for each `Gap Request` message received by the GRP.

| Gap Response | | | | |
|--------------------------------|----------------------------------------------------|--------|--------------|-----------------------------------------------------|
| Field | Offset | Length | Value/Type | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x04 | <code>Gap Response</code> Message |
| <i>Unit</i> | 2 | 1 | Binary | <i>Unit</i> the gap was requested for. |
| <i>Sequence</i> | 3 | 4 | Binary | <i>Sequence</i> of first message in request. |
| <i>Count</i> | 7 | 2 | Binary | <i>Count</i> of messages requested. |
| <i>Status</i> | 9 | 1 | Alphanumeric | Accepted or reason for reject*. |
| Total Length = 10 bytes | | | | |
| Gap Response - Status Codes | | | | |
| 'A' | Accepted | | | |
| 'O' | Out of range (ahead of sequence or too far behind) | | | |
| 'D' | Daily gap request allocation exhausted | | | |
| 'M' | Minute gap request allocation exhausted | | | |
| 'S' | Second gap request allocation exhausted | | | |
| 'C' | Count request limit for one gap request exceeded | | | |
| 'I' | Invalid Unit specified in request | | | |
| 'U' | Unit is currently unavailable | | | |

* - All non-'A' status codes should be interpreted as a reject.

4 Spin Messages

4.1 Login

The `Login` message is the first message sent to the Spin Server by a user's process after the connection to the Spin Server is established. Failure to login before sending any other message type will result in the connection being dropped by the Spin Server.

The format of the `Login` message for the Spin Server is identical to that of the GRP described previously in [Section 4.1](#).

4.2 Login Response

The `Login Response` message is sent by the Spin Server to a user's process in response to a `Login` message. The status field is used to reflect an accepted login or the reason the session was not accepted. If login fails, the connection will be dropped after the `Login Response` message is sent.

The format of the `Login Response` message for the Spin Server is identical to that of the GRP described previously in [Section 4.2](#).

4.3 Spin Image Available

The `Spin Image Available` message is sent once per second and indicates what sequence number a spin is available through.

| Spin Image Available | | | | |
|-------------------------------|--------|--------|--------------|------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x80 | Spin Image Available Message |
| <i>Sequence</i> | 2 | 4 | Binary | Spin is available which is current through this sequence number. |
| Total Length = 6 bytes | | | | |

4.4 Spin Request

The `Spin Request` message is used by a user's process to request transmission of a spin of the unit's order book. Refer to [Section 1.6](#) for more complete details regarding *Sequence* specification as well as buffering requirements.

| Spin Request | | | | |
|-------------------------------|--------|--------|--------------|--------------------------------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x81 | Spin Request Message |
| <i>Sequence</i> | 2 | 4 | Binary | Sequence number from a <code>Spin Image Available</code> message received by the customer. |
| Total Length = 6 bytes | | | | |

4.5 Spin Response

The `Spin Response` message is sent in response to a user's `Spin Request` message indicating whether a spin will be sent.

| Spin Response | | | | |
|-------------------------------------|------------------------------------------------------------------------------------------------------|--------|--------------|--------------------------------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x82 | <code>Spin Response</code> Message |
| <i>Sequence</i> | 2 | 4 | Binary | Sequence number from a <code>Spin Image</code> Available message received by the customer. |
| <i>Order Count</i> | 6 | 4 | Binary | Always zero. |
| <i>Status</i> | 10 | 1 | Alphanumeric | Accepted or reason for reject*. |
| Total Length = 11 bytes | | | | |
| Spin Response - Status Codes | | | | |
| 'A' | Accepted | | | |
| 'O' | Out of Range (<i>Sequence</i> requested is greater than <i>Sequence</i> available by the next spin) | | | |
| 'S' | Spin already in progress (only one spin can be running at a time). | | | |

* - All non-'A' status codes should be interpreted as a reject.

4.6 Spin Finished

The `Spin Finished` message is sent to indicate that all messages for the spin requested have been sent. A `Spin Finished` message is only sent if a `Spin Request` was not rejected. Upon receipt of a `Spin Finished` message, any buffered multicast messages should be applied to the customer's copy of the book to make it current.

| Spin Finished | | | | |
|-------------------------------|--------|--------|--------------|-------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| <i>Message Type</i> | 1 | 1 | 0x83 | <code>Spin Finished</code> Message |
| <i>Sequence</i> | 2 | 4 | Binary | Sequence number from the <code>Spin Request</code> message. |
| Total Length = 6 bytes | | | | |

4.7 Instrument Definition Request

The `Instrument Definition Request` message is used by a user's process to request transmission of this unit's Symbol Mappings. Refer to Section 1.5 for more complete details regarding *Sequence* specification as well as buffering requirements.

| Instrument Definition Request | | | | |
|-------------------------------|--------|--------|--------------|----------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| | | | | |
|-------------------------------|---|---|--------|------------------------------------------------------------|
| <i>Message Type</i> | 1 | 1 | 0x84 | Instrument Definition Request Message |
| <i>Sequence</i> | 2 | 4 | Binary | Must be 0. Only the current Symbol Mappings are available. |
| Total Length = 6 bytes | | | | |

4.8 Instrument Definition Response

The Instrument Definition Response message is sent in response to a user's Instrument Definition Request message indicating whether a spin will be sent.

| Instrument Definition Response | | | | |
|-----------------------------------------------|-------------------------------------------------------------------|--------|--------------|-------------------------------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0x85 | Instrument Definition Response Message |
| <i>Sequence</i> | 2 | 4 | Binary | Will always be 0. |
| <i>Instrument Count</i> | 6 | 4 | Binary | Number of Symbol Mapping messages which will be contained in this spin. |
| <i>Status</i> | 10 | 1 | Alphanumeric | Accepted or reason for reject |
| Total Length = 11 bytes | | | | |
| Instrument Definition Response - Status Codes | | | | |
| 'A' | Accepted | | | |
| 'O' | Out of Range (<i>Sequence</i> must be 0) | | | |
| 'S' | Spin already in progress (only one spin can be running at a time) | | | |

* - All non-'A' status codes should be interpreted as a reject.

4.9 Instrument Definition Finished

The Instrument Definition Finished message is sent to indicate that all Symbol Mapping messages for this unit have been sent. An Instrument Definition Finished message is only sent if an Instrument Definition Request was not rejected.

| Instrument Definition Finished | | | | |
|--------------------------------|--------|--------|--------------|----------------------------------------------------|
| Field Name | Offset | Length | Type/(Value) | Description |
| <i>Length</i> | 0 | 1 | Binary | <i>Length</i> of this message including this field |
| <i>Message Type</i> | 1 | 1 | 0x86 | Instrument Definition Finished Message |
| Total Length = 2 bytes | | | | |

5 Message Types

5.1 Gap Request Proxy Messages

| | |
|------|----------------|
| 0x01 | Login |
| 0x02 | Login Response |
| 0x03 | Gap Request |
| 0x04 | Gap Response |

5.2 Spin Server Messages

| | |
|------|--------------------------------|
| 0x01 | Login |
| 0x02 | Login Response |
| 0x80 | Spin Image Available |
| 0x81 | Spin Request |
| 0x82 | Spin Response |
| 0x83 | Spin Finished |
| 0x84 | Instrument Definition Request |
| 0x85 | Instrument Definition Response |
| 0x86 | Instrument Definition Finished |

5.3 FLEX Feed Messages

| | |
|------|------------------------------------|
| 0x20 | Time |
| 0x9C | FLEX Instrument Definition |
| 0x9B | Complex FLEX Instrument Definition |
| 0xAD | Auction Notification |
| 0xAE | Auction Cancel |
| 0xAF | Auction Trade |
| 0x2A | Trade (long) |
| 0x2B | Trade (short) |
| 0x2C | Trade Break |
| 0x99 | Trading Status |
| 0x2D | End of Session |

6 Example Messages

Each of the following message types must be wrapped by a sequenced or unsequenced unit header as described in [Section 2.4](#). Note that in the following examples, each byte is represented by two hexadecimal digits.

6.1 Login Message

| | | |
|--------------|-------------------------------|----------|
| Length | 16 | 22 bytes |
| Type | 01 | Login |
| SessionSubId | 30 30 30 31 | "0001" |
| Username | 46 49 52 4D | "FIRM" |
| Filler | 20 20 | " " |
| Password | 41 42 43 44 30 30 20 20 20 20 | "ABCD00" |

6.2 Login Response Message

| | | |
|--------|----|----------------|
| Length | 03 | 3 bytes |
| Type | 02 | Login Response |
| Status | 41 | Login accepted |

6.3 Gap Request Message

| | | |
|----------|-------------|---------------------|
| Length | 09 | 9 bytes |
| Type | 03 | Gap Request |
| Unit | 01 | Unit 1 |
| Sequence | 3B 10 00 00 | First message: 4155 |
| Count | 32 00 | 50 messages |

6.4 Gap Response Message

| | | |
|----------|-------------|---------------------|
| Length | 08 | 8 bytes |
| Type | 04 | Gap Response |
| Unit | 01 | Unit 1 |
| Sequence | 3B 10 00 00 | First message: 4155 |
| Status | 41 | Accepted |

6.5 Spin Image Available Message

| | | |
|----------|-------------|----------------------|
| Length | 06 | 6 bytes |
| Type | 80 | Spin Image Available |
| Sequence | 3B 10 00 00 | Sequence: 4155 |

6.6 Spin Request Message

| | | |
|----------|-------------|----------------|
| Length | 06 | 6 bytes |
| Type | 81 | Spin Request |
| Sequence | 3B 10 00 00 | Sequence: 4155 |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

6.7 Spin Response Message

| | | |
|-------------|-------------|----------------|
| Length | 0B | 11 bytes |
| Type | 82 | Spin Request |
| Sequence | 3B 10 00 00 | Sequence: 4155 |
| Order Count | 00 00 00 00 | 0 orders |
| Status | 41 | Accepted |

6.8 Spin Finished Message

| | | |
|----------|-------------|----------------|
| Length | 06 | 6 bytes |
| Type | 83 | Spin Finished |
| Sequence | 3B 10 00 00 | Sequence: 4155 |

6.9 Instrument Definition Request

| | | |
|----------|-------------|-------------------------------|
| Length | 06 | 6 bytes |
| Type | 84 | Instrument Definition Request |
| Sequence | 00 00 00 00 | Sequence: 0 |

6.10 Instrument Definition Response

| | | |
|------------------|-------------|--------------------------------|
| Length | 0B | 11 bytes |
| Type | 85 | Instrument Definition Response |
| Sequence | 00 00 00 00 | Sequence: 0 |
| Instrument Count | B8 0B 00 00 | 3,000 Instruments |
| Status | 41 | Accepted |

6.11 Instrument Definition Finished

| | | |
|--------|----|--------------------------------|
| Length | 02 | 2 bytes |
| Type | 86 | Instrument Definition Finished |

6.12 Time Message

| | | |
|--------|-------------|--------------------------------------|
| Length | 06 | 6 bytes |
| Type | 20 | Time |
| Time | 98 85 00 00 | 34,200 seconds = 09:30 AM Eastern |

6.13 FLEX Instrument Definition Message

| | | |
|-------------|-------------------|------------------------------------|
| Length | 39 | 57 bytes |
| Type | 9C | Flex Instrument Definition |
| Time offset | 18 D2 06 00 | 447,000 ns since last Time Message |
| Feed Symbol | 46 30 30 30 31 32 | F00012 |
| OSI Root | 31 41 41 50 4C 20 | 1AAPL |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| | | |
|------------------|-------------------------|------------------------|
| Year | 31 38 | 18 |
| Month | 31 31 | 11 |
| Day | 32 32 | 22 |
| Call/Put | 43 | C = Call |
| Dollar Strike | 00 00 32 30 37 | 00207 = \$207 |
| Decimal Strike | 37 35 00 | 750 = \$0.750 |
| Symbol Condition | 4E | N = Normal |
| Underlying | 41 41 50 4C 20 20 20 20 | AAPL |
| Exercise Style | 41 | American |
| Settlement Type | 50 | PM |
| Percentage | 00 00 00 00 | Not percentage pricing |
| Observation Day | 20 20 | No Observation Day |
| Return Cap | 00 00 00 00 | 250 = 2.5% Return cap |
| Creation Day | 20 20 | No Creation Day |
| Bit Fields | 00 | Not percentage pricing |

6.14 Complex FLEX Instrument Definition Message

| | | |
|-------------------------------|-------------------------|-------------------------------------------------|
| Length | 43 | 67 bytes |
| Type | 9B | FLEX Instrument Definition |
| Time offset | 18 D2 06 00 | 447,000 ns since last Time Message |
| CID | 43 30 30 30 31 32 | C00012 |
| Underlying | 5A 56 5A 5A 54 20 20 20 | ZVZZT |
| Complex Instrument Type | 58 20 20 20 | X = All Legs are Flex Options |
| Leg Count | 03 | 3 legs |
| Message Count | 01 | Only one message needed to define instrument |
| Message Number | 01 | Message 1 of 1 |
| Message Leg Count | 03 | 3 legs |
| Leg Symbol | 30 30 30 30 30 31 20 20 | 000001 |
| Leg Ratio | FF FF FF FF | -1 = Sell 1 |
| Leg Security Type | 58 | FLEX Option Leg |
| Leg Symbol | 30 30 30 30 30 32 20 20 | 000002 |
| Leg Ratio | FF FF FF FF | -1 = Sell 1 |
| Leg Security Type | 58 | FLEX Option Leg |
| Leg Symbol | 30 30 30 30 30 33 20 20 | 000003 |
| Leg Ratio | 02 00 00 00 | 2 = Buy 2 |
| Leg Security Type | 58 | FLEX Option Leg |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

6.15 Auction Notification Message

| | | |
|------------------|-------------------------|---------------------------------------|
| Length | 2F | 47 bytes |
| Type | AD | Auction Notification |
| Time offset | 18 D2 06 00 | 447,000 ns since last Time Message |
| CID | 30 30 6d 45 56 4f | 00mEVO |
| Auction ID | 05 40 5B 77 8F 56 1D 0B | 631WC4000005 |
| Auction Type | 53 | S = SAM |
| Side | 42 | B = Buy Side |
| Price | E8 A3 0F 00 00 00 00 00 | \$102.50 |
| Quantity | 64 00 00 00 | 100 |
| Customer | 43 | C = Customer |
| Indicator | | |
| ParticipantID | 45 46 49 44 | EFID |
| Auct. End Offset | 38 73 0E 00 | 947,000 ns since last Time Message |
| ClientID | 41 42 43 44 | ABCD |

6.16 Auction Cancel Message

| | | |
|-------------|-------------------------|---------------------------------------|
| Length | 0E | 14 bytes |
| Type | AE | Auction Cancel |
| Time offset | 18 D2 06 00 | 447,000 ns since last Time Message |
| Auction ID | 05 40 5B 77 8F 56 1D 0B | 631WC4000005 |

6.17 Auction Trade Message

| | | |
|--------------|-------------------------|---------------------------------------|
| Length | 22 | 34 bytes |
| Type | AF | Auction Trade |
| Time offset | 18 D2 06 00 | 447,000 ns since last Time Message |
| Auction ID | 05 40 5B 77 8F 56 1D 0B | 631WC4000005 |
| Execution ID | 34 2B 46 E0 BB 00 00 00 | 0AAP09VEC |
| Price | E8 A3 0F 00 00 00 00 00 | 1.0250% |
| Quantity | 64 00 00 00 | 100 |

6.18 Trade (long) Message

| | | |
|-------------|-------------------------|-------------------------------------------|
| Length | 2A | 42 bytes |
| Type | 2A | Trade (long) |
| Time Offset | 08 5C 44 25 | 625,237,000 ns since Last Time Message |
| Order Id | 05 40 5B 77 8F 56 1D 0B | |
| Side | 42 | Buy |
| Quantity | F8 24 01 00 | 75,000 contracts |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| | | |
|-----------------|-------------------------|------------------|
| Symbol | 33 34 35 33 32 31 | 345321 |
| Price | E8 A3 0F 00 00 00 00 00 | \$102.50 |
| Execution Id | 34 2B 46 E0 BB 00 00 00 | 0AAP09VEC |
| Trade Condition | 20 | <Space> = Normal |

6.19 Trade (short) Message

| | | |
|-----------------|-------------------------|-------------------------------------------|
| Length | 22 | 33 bytes |
| Type | 2B | Trade (short) |
| Time Offset | 08 5C 44 25 | 625,237,000 ns since Last Time Message |
| Order Id | 05 40 5B 77 8F 56 1D 0B | |
| Side | 42 | Buy |
| Quantity | 64 00 | 100 contracts |
| Symbol | 33 34 35 33 32 31 | 345321 |
| Price | 0A 28 | \$102.50 |
| Execution Id | 34 2B 46 E0 BB 00 00 00 | 0AAP09VEC |
| Trade Condition | 20 | <Space> = Normal |

6.20 Trade Break Message

| | | |
|--------------|-------------------------|---------------------------------------|
| Length | 0E | 14 bytes |
| Type | 2C | Trade Break |
| Time offset | 18 D2 06 00 | 447,000 ns since last Time Message |
| Execution Id | 34 2B 46 E0 BB 00 00 00 | 0AAP09VEC |

6.21 Trading Status Message

| | | |
|----------------|-------------------|---------------------------------------|
| Length | 12 | 18 bytes |
| Type | 31 | Trading Status |
| Time Offset | 18 D2 06 00 | 447,000 ns since last Time Message |
| Symbol | 39 39 38 38 37 37 | 998877 |
| Reserved | 20 20 | Reserved |
| Trading Status | 54 | T = Trading |
| Reserved | 20 | Reserved |
| Global Trading | 48 | H = Halted |
| Hours Status | | |
| Reserved | 20 | Reserved |

6.22 End of Session

| | | |
|-------------|-------------|---------------------------------------|
| Length | 06 | 6 bytes |
| Type | 2D | End of Session |
| Time offset | 18 D2 06 00 | 447,000 ns since last Time Message |

7 Multicast Configuration

7.1 Production Environment Configuration

7.1.1 Limitations/Configurations

The following table defines Cboe current configuration for network and gap request limitations. These limitations are session based. Cboe reserves the right to adjust the gap request limitations to improve the effectiveness of the gap request infrastructure.

| Period/Type | Limit/Setting | Notes |
|---------------------|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MTU | 1500 | Cboe will send UDP messages up to 1500 bytes. Members should ensure that their infrastructure is configured accordingly. |
| WAN-Shaped Throttle | 100 Mb/s | The real-time and gap multicast head ends are configured to shape their output to this level to minimize packet loss. |
| Gap Response Delay | 2 ms | The Gap Server will delay resending sequenced messages via multicast for the specified limit in order to satisfy multiple GRP gap requests with one multicast response. |
| Count | 100 | Any single gap request may not be for more than this number of dropped messages. |
| 1 Second | 320 Requests | This is the maximum number of retransmission requests allowed per second for each session. This is renewed every clock second. |
| 1 Minute | 1500 Requests | This is the maximum number of retransmission requests allowed per minute for each session. This is renewed every clock minute. |
| Day | 100,000 Requests | This is the maximum number of retransmission requests allowed per day for each session. |
| Within Range | 1,000,000 Messages | Users' retransmission requests must be within this many messages of the most recent sequence sent by the real-time feed per session. |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

7.1.2 Unit/Symbol Distribution

| Unit | C1 Symbol Range | C1 Exceptions |
|------|-----------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|
| 1 | A – ACNAZ | |
| 2 | ACNB – AMGNZ | |
| 3 | AMGO – BAAZ | Excludes AMZN |
| 4 | BAB – BKNFZ | |
| 5 | BKNG – CASZZ | |
| 6 | CAT – COOZZ | |
| 7 | COP – DEAAZ | |
| 8 | DEAB – EEMAZ | Excludes DJX |
| 9 | EEMB – FBAAZ | |
| 10 | FBAB – GOOFZ | |
| 11 | GOOG – GOOGZ | |
| 12 | GOOH – IFFAZ | |
| 13 | IFFB – IWLZZ | |
| 14 | IWM – IWMAZ | |
| 15 | IWMB – LOVZZ | |
| 16 | LOW – MPCAZ | |
| 17 | MPCB – NFLWZ | Excludes MXEA, MXEF |
| 18 | NFLX – NUEAZ | |
| 19 | NUEB – PEPAZ | Excludes OEX |
| 20 | PEPB – ROKUZ | Excludes RLG, RLV |
| 21 | ROKV – SPZZZ | Excludes RUI, RUT, RUTW, SIXB, SIXC, SIXE, SIXI, SIXR, SIXRE, SIXT, SIXU, SIXV, SIXY, SPX, SPXW, SPY |
| 22 | SQ – TLRYZ | |
| 23 | TLRZ – TSLAZ | |
| 24 | TSLB – UVXXZ | Excludes UKXM |
| 25 | UVXY – VZZZZ | Excludes VIX, VIXW |
| 26 | W – XLEAZ | Excludes XEO |
| 27 | XLEB – ZZZZZ | Excludes XSP |
| 28 | QQQ | |
| 29 | AMZN | |
| 30 | SPY | |
| 31 | DJX, MXEA, MXEF, OEX, RLG, RLV, RUI, RUT, RUTW, SIXB, SIXC, SIXE, SIXI, SIXR, SIXRE, SIXT, SIXU, SIXV, SIXY, XEO, UKXM, XSP | |
| 32 | VIX, VIXW | |
| 33 | SPX | |
| 34 | SPXW | |
| 35 | SPX/SPXW, Cross Product Spreads | |

Note - Cboe reserves the right to add units and/or change symbol distribution with 48 hours of notice and no migration period. Notice will be given that the distribution will change on a certain date. Care should be taken to support mappings in these tables via software configuration.

7.1.3 Cboe Options FLEX Multicast Routing Parameters

| Data center | Rendezvous Point |
|----------------------------------|------------------|
| NY5 Primary Data Center A feed | 74.115.128.183 |
| NY5 Primary Data Center B feed | 74.115.128.184 |
| CH4 Secondary Data Center E feed | 174.136.181.249 |

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

7.1.4 Cboe Options FLEX Multicast Addresses

The following tables describe the multicast address distribution across production multicast Cboe FLEX feeds.

| Primary Datacenter | | WAN-Shaped [CAF] 170.137.114.80 /28 | | WAN-Shaped [CBF] 170.137.115.80 /28 | |
|--------------------|---------|----------------------------------------|--------------|----------------------------------------|-----------------|
| Unit | IP Port | Real-time MC | Gap Resp. MC | Real-time MC | Gap Resp. MC |
| 1 | 30501 | | | | |
| 2 | 30502 | | | | |
| 3 | 30503 | | | | |
| 4 | 30504 | | | | |
| 5 | 30505 | | | | |
| 6 | 30506 | | | | |
| 7 | 30507 | | | | |
| 8 | 30508 | | | | |
| 9 | 30509 | | | | |
| 10 | 30510 | | | | |
| 11 | 30511 | | | | |
| 12 | 30512 | | | | |
| 13 | 30513 | | | | |
| 14 | 30514 | | | | |
| 15 | 30515 | | | | |
| 16 | 30516 | | | | |
| 17 | 30517 | 224.0.74.92 | 224.0.74.93 | 233.182.199.220 | 233.182.199.221 |
| 18 | 30518 | | | | |
| 19 | 30519 | | | | |
| 20 | 30520 | | | | |
| 21 | 30521 | | | | |
| 22 | 30522 | | | | |
| 23 | 30523 | | | | |
| 24 | 30524 | | | | |
| 25 | 30525 | | | | |
| 26 | 30526 | | | | |
| 27 | 30527 | | | | |
| 28 | 30528 | | | | |
| 29 | 30529 | | | | |
| 30 | 30530 | | | | |
| 31 | 30531 | | | | |
| 32 | 30532 | | | | |
| 33 | 30533 | | | | |
| 34 | 30534 | | | | |
| 35 | 30535 | | | | |

Note – Cboe reserves the right to add multicast addresses with prior notice, but no migration period. Notice will be given that the distribution will change on a certain date. Care should be taken to support mappings in these tables via software configuration.

Cboe Options Exchange
FLEX Feed Specification (Version 1.0.9)

| Secondary Datacenter | | WAN-Shaped [CEF] 170.137.124.224/28 | |
|----------------------|---------|----------------------------------------|--------------|
| Unit | IP Port | Real-time MC | Gap Resp. MC |
| 1 | 31501 | 233.19.3.252 | 233.19.3.253 |
| 2 | 31502 | | |
| 3 | 31503 | | |
| 4 | 31504 | | |
| 5 | 31505 | | |
| 6 | 31506 | | |
| 7 | 31507 | | |
| 8 | 31508 | | |
| 9 | 31509 | | |
| 10 | 31510 | | |
| 11 | 31511 | | |
| 12 | 31512 | | |
| 13 | 31513 | | |
| 14 | 31514 | | |
| 15 | 31515 | | |
| 16 | 31516 | | |
| 17 | 31517 | | |
| 18 | 31518 | | |
| 19 | 31519 | | |
| 20 | 31520 | | |
| 21 | 31521 | | |
| 22 | 31522 | | |
| 23 | 31523 | | |
| 24 | 31524 | | |
| 25 | 31525 | | |
| 26 | 31526 | | |
| 27 | 31527 | | |
| 28 | 31528 | | |
| 29 | 31529 | | |
| 30 | 31530 | | |
| 31 | 31531 | | |
| 32 | 31532 | | |
| 33 | 31533 | | |
| 34 | 31534 | | |
| 35 | 31535 | | |

Note – Cboe reserves the right to add multicast addresses with prior notice, but no migration period. Notice will be given that the distribution will change on a certain date. Care should be taken to support mappings in these tables via software configuration.

7.2 Certification Environment Configuration

7.2.1 Unit/Symbol Distribution

| Unit | C1 Symbol Range | C1 Exceptions |
|------|--------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
| 1 | A – ACNAZ | |
| 2 | ACNB – AMGNZ | |
| 3 | AMGO – BAAZ | Excludes AMZN |
| 4 | BAB – BKNFZ | |
| 5 | BKNG – CASZZ | |
| 6 | CAT – COOZZ | |
| 7 | COP – DEAAZ | |
| 8 | DEAB – EEMAZ | Excludes DJX |
| 9 | EEMB – FBAAZ | |
| 10 | FBAB – GOOFZ | |
| 11 | GOOG – GOOGZ | |
| 12 | GOOH – IFFAZ | |
| 13 | IFFB – IWLZZ | |
| 14 | IWM – IWMAZ | |
| 15 | IWMB – LOVZZ | |
| 16 | LOW – MPCAZ | |
| 17 | MPCB – NFLWZ | |
| 18 | NFLX – NUEAZ | |
| 19 | NUEB – PEPAZ | Excludes OEX |
| 20 | PEPB – ROKUZ | |
| 21 | ROKV – SPZZZ | Excludes RUT, RUTW, SIXB, SIXC, SIXE, SIXI, SIXR, SIXRE, SIXT, SIXU, SIXV, SIXY, SPX, SPXW, SPY |
| 22 | SQ – TLRYZ | |
| 23 | TLRZ – TSLAZ | |
| 24 | TSLB – UVXXZ | |
| 25 | UVXY – VZZZZ | Excludes VIX, VIXW |
| 26 | W – XLEAZ | Excludes XEO |
| 27 | XLEB – ZZZZZ | Excludes XSP |
| 28 | QQQ | |
| 29 | AMZN | |
| 30 | SPY | |
| 31 | DJX, OEX, RUT, RUTW, XEO, XSP, SIXB, SIXC, SIXE, SIXI, SIXR, SIXRE, SIXT, SIXU, SIXV, SIXY | |
| 32 | VIX, VIXW | |
| 33 | SPX | |
| 34 | SPXW | |
| 35 | SPX/SPXW, Cross Product Spreads | |

Note – Cboe reserves the right to add units and/or change symbol distribution with 48 hours of notice and no migration period. Notice will be given that the distribution will change on a certain date. Care should be taken to support mappings in these tables via software configuration.

7.2.2 Cboe Options FLEX Certification Multicast Routing Parameters

| Data center | Rendezvous Point |
|---------------------|-----------------------|
| Primary Data Center | 74.115.128.131 |

7.2.3 Cboe Options FLEX Certification Multicast Addresses

The following tables describe the multicast address distribution across production multicast Cboe FLEX feeds.

| Primary Datacenter | | WAN-Shaped 170.137.126.16/28 | |
|--------------------|---------|---------------------------------|----------------|
| Unit | IP Port | Real-time MC | Gap Resp. MC |
| 1 | 32501 | | |
| 2 | 32502 | | |
| 3 | 32503 | | |
| 4 | 32504 | | |
| 5 | 32505 | | |
| 6 | 32506 | | |
| 7 | 32507 | | |
| 8 | 32508 | | |
| 9 | 32509 | | |
| 10 | 32510 | | |
| 11 | 32511 | | |
| 12 | 32512 | | |
| 13 | 32513 | | |
| 14 | 32514 | | |
| 15 | 32515 | | |
| 16 | 32516 | | |
| 17 | 32517 | 233.103.126.18 | 233.103.126.19 |
| 18 | 32518 | | |
| 19 | 32519 | | |
| 20 | 32520 | | |
| 21 | 32521 | | |
| 22 | 32522 | | |
| 23 | 32523 | | |
| 24 | 32524 | | |
| 25 | 32525 | | |
| 26 | 32526 | | |
| 27 | 32527 | | |
| 28 | 32528 | | |
| 29 | 32529 | | |
| 30 | 32530 | | |
| 31 | 32531 | | |
| 32 | 32532 | | |
| 33 | 32533 | | |
| 34 | 32534 | | |
| 35 | 32535 | | |

Note – Cboe reserves the right to add multicast addresses with prior notice, but no migration period. Notice will be given that the distribution will change on a certain date. Care should be taken to support mappings in these tables via software configuration.

8 Connectivity

8.1 Supported Extranet Carriers

Cboe has certified a number of carriers defined in the [Cboe C1 Options Connectivity Manual](#) with respect to redistribution of Cboe Multicast data feeds. For more information on receiving the FLEX feed through any of these providers, reach out to the vendor contact noted in the Extranet Providers section of the Connectivity Manual.

8.2 Bandwidth Recommendation

The WAN-shaped feeds require 100 Mbps of bandwidth. Cboe will use 90% of these respective bandwidths for the FLEX feed to allow customers to use the same physical connection for FIX order entry if desired.

9 References

For more information on Cboe Symbology, please refer to the [Cboe Symbology Reference](#) document.

10 Support

Please e-mail questions or comments regarding this specification to tradedesk@cboe.com.

Revision History

| Document Version | Date | Description |
|------------------|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1.0.0 | 11/16/18 | Initial draft in support of FLEX Options on Cboe Options Exchange. |
| 1.0.1 | 02/14/19 | Added certification environment IP port and unit distribution information. |
| 1.0.2 | 04/02/19 | Added C1 Certification data center rendezvous point IP address. Corrected C1 Symbol Range. |
| 1.0.3 | 04/17/19 | Added Production IP addresses for C1 Options. |
| 1.0.4 | 05/08/19 | Removed <i>Trading Status</i> field value 'S' = Exchange Specific Suspension from <i>Trading Status</i> message, as this was added in error. Corrected C1 Production WAN-Shaped [CAF] and [CBF] source network IP addresses. |
| 1.0.5 | 06/12/19 | Added additional proprietary products to matching unit 31 in C1. Corrected certification and production C1 symbol range for units 9 and 20. Corrected example for <i>FLEX Instrument Definition</i> message. |
| 1.0.6 | 06/21/19 | Added <i>Trade (short)</i> message type and corresponding example message. |
| 1.0.7 | 09/06/19 | Clarified note for <i>Price</i> field in <i>Auction Notification</i> message to indicate that the field value will be zero when <i>Auction Type</i> = B or F. |
| 1.0.8 | 09/25/19 | <i>Customer Indicator</i> will be blank when <i>Auction Type</i> = F. |
| 1.0.9 | 10/11/19 | Clarified description of <i>Time</i> message. Corrected <i>Auction Notification Price</i> field description. |