



Cboe Titanium U.S. Equities Summary Depth Feed Specification

Version 1.0.7

January 15, 2025

This content is owned or licensed by Cboe Global Markets, Inc. or its affiliates ("Cboe") and protected by copyright under U.S. and international copyright laws. Other than for internal business purposes, you may not copy, reproduce, distribute, publish, display, perform, modify, create derivative works, transmit, or in any way exploit the content, sell or offer it for sale, use the content to construct any kind of database, or alter or remove any copyright or other notice from copies of the content.

Contents

| | |
|--|----|
| Overview | 3 |
| Cboe Summary Depth Server (TCP) | 4 |
| Cboe Summary Depth Feed Server (UDP) | 5 |
| Cboe Summary Depth Gap Request Proxy and Message Retransmission via Gap Server | 6 |
| Cboe Summary Depth Disaster Recovery | 7 |
| Protocol | 8 |
| Message Format | 8 |
| Data Types | 9 |
| Message Framing | 10 |
| Sequenced Unit Header Message Fields | 11 |
| Heartbeat Messages | 12 |
| Cboe Summary Depth Server Session Messages (TCP) | 13 |
| Cboe Summary Depth Login Message Fields | 13 |
| Login Response Message Fields | 14 |
| Replay Complete Message Fields | 15 |
| Cboe Summary Depth Gap Request Proxy Session Messages (TCP) | 16 |
| GRP Login Message Fields | 16 |
| Login Response Message Fields | 17 |
| Gap Request Message Fields | 18 |
| Gap Response Message Fields | 19 |
| Cboe Summary Depth Update Messages (UDP & TCP) | 20 |
| Clear Quote Message Fields | 20 |
| Cboe Market Status Message Fields | 21 |
| ADAP Message Fields | 22 |
| Retail Price Improvement (RPI) Message Fields | 24 |
| Trade Message Fields | 25 |
| Trade Break Message Fields | 26 |
| Trading Status Message Fields | 27 |
| Multicast Configuration | 28 |
| US Equities Production Environment Configuration | 28 |
| Limitations/Configurations | 28 |
| Cboe Summary Depth Multicast Routing Parameters | 29 |
| Cboe Summary Depth Multicast Addresses | 30 |
| US Equities Certification Environment Configuration | 31 |
| Cboe Summary Depth Certification Multicast Routing Parameters | 31 |

| | |
|---|----|
| Cboe Summary Depth Certification Multicast Addresses..... | 32 |
| References..... | 33 |
| Symbology | 33 |
| Support..... | 34 |
| Revision History | 35 |

Overview

Note that this specification will be the standard Cboe Titanium U.S. Equities Summary Depth Feed specification to be used for Cboe BYX Exchange, BZX Exchange, EDGA Exchange, and EDGX Exchange platforms.

The Cboe Summary Depth Feed delivers quote, trade and Aggregated Depth At Price (ADAP) information for the respective Cboe book via TCP/IP and UDP using the binary Cboe Summary Depth protocol. The feed consists of **Clear Quote**, **Market Status**, **ADAP**, **RPI**, **Trade**, **Trade Break**, and **Trading Status** messages.

The TCP/IP delivered feed can be used as a standalone product or to augment the UDP feed for recovery and start up purposes. The TCP/IP feed is available from the Cboe Summary Depth Server and sends a replay of missed trades and refreshes the current state of the book followed by real-time updates to the book after a connection is established.

The UDP delivered feed is sourced from the Cboe Summary Depth Feed Server (FS). Users may also connect to the Cboe Summary Depth Gap Request Proxy for retransmission of missed packets on the UDP feed by the Cboe Summary Depth Gap Server (GS).

While the TCP/IP and UDP delivered feeds offer equivalent real-time updates with matching sequence numbers, the consumer should assume message framing will be different between the transmission protocols.

It is important to note that the Cboe Summary Depth feed is an aggregated feed. Updates for a symbol are sent as capacity for the feed allows. The image for a symbol will be current at the time of delivery, but multiple updates may be combined into a single update. The interval between updates is dependent on the market conditions and the capacity configuration of the particular Cboe Summary Depth feed.

The Cboe Summary Depth Feed is available with 5 ADAP levels and requires bandwidth of up to 250Mb.

Cboe Summary Depth Server (TCP)

The client connects to an assigned host and port using a TCP/IP socket.

Upon connection, the Member must send a **Cboe Summary Depth Login** message. In the **Cboe Summary Depth Login** message, the *Next Sequence* field allows Members to specify the next sequence number they expect to receive. If a Member logs in after trading begins or after connection loss, the *Next Sequence* field can be used to tell the server to replay any **Trade** and **Trade Break** messages that have occurred since the last received **Trade** or **Trade Break** message.

If the *Next Sequence* field is set to 1 (one), then all **Trade** and **Trade Break** messages from the beginning of the day will be replayed after the server sends a successful **Login Response** message. Once the server has replayed any **Trade** and **Trade Break** messages, it will deliver relevant **Market Status** message information, **RPI** message information, **Trading Status** messages, and a spin of **ADAP** messages from its cache for each active symbol. Then the server will send the Member a **Replay Complete** message followed by the live stream of **ADAP**, **RPI**, **Trade**, **Trade Break**, **Market Status**, and **Trading Status** messages.

If the *Next Sequence* field is set to 0 (zero), then no **Trade** message or **Trade Break** messages will be replayed after the server sends a successful **Login Response** message. However, the Member may still receive a spin of **Market Status**, **RPI**, **Trading Status** and **ADAP** messages followed by a **Replay Complete** message before receiving the live stream of **ADAP**, **RPI**, **Trade**, **Trade Break**, **Market Status**, and **Trading Status** messages.

If a Member's process cannot keep up with the Cboe Summary Depth feed's rate of transmission, the connection will be closed by the server. The client should then reconnect and login with the appropriate *Next Sequence* number to receive any missed trades, and spin of the latest image for all symbols.

Cboe Summary Depth Feed Server (UDP)

The UDP delivered Cboe Summary Depth Feed is sourced by the Cboe Summary Depth Feed Server (FS). The FS generates the multicast events for the Cboe Summary Depth feed and performs throttling of events to ensure the bandwidth requirements of the feed are not exceeded.

The FS does not receive messages from Members and no login is necessary.

Multicast addresses and ports for the Cboe Summary Depth feed are listed in the Multicast Configuration section of this document.

Cboe Summary Depth Gap Request Proxy and Message Retransmission via Gap Server

Requesting delivery of missed data is achieved by connecting to a Cboe Summary Depth Gap Request Proxy (Cboe Summary Depth GRP). Members who do not wish to request missed messages do not need to connect to a GRP for any reason or listen to the multicast addresses reserved for message retransmission. Members choosing to request missed data will need to connect to their assigned GRP, log in, and request gap ranges as necessary. All gap requests will be responded to with a **Gap Response** message. A **Gap Response** message *Status* code of Accepted signals that the replayed messages will be delivered via the appropriate gap response multicast address. Any other **Gap Response** message *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. Larger sequence number gaps should be recovered via the Cboe Summary Depth Server over TCP. Members will receive a total daily allowance of gap requested messages. In addition, each Member is given renewable one second and one minute gap request limits.

If more than one gap request is received for a particular 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. Members will receive gap responses for their requested 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.

Cboe Summary Depth Disaster Recovery

Users of Cboe Summary Depth that are interested in disaster recovery must maintain connectivity to the Cboe Disaster Recovery (DR) site in Chicago, IL. To establish new connectivity to the DR site, contact the [Cboe NOC](#).

The Cboe Summary Depth feed that is disseminated from the DR site connects to the Cboe PITCH service sourced in the DR site. It serves the same data as is distributed from the Cboe Primary site in Secaucus, NJ all day long, so Cboe Summary Depth customers can fail-over to or utilize the Cboe Summary Depth feed out of the DR site at any time. However, note that sequence numbers are not guaranteed to be the same between the Primary and DR sites.

The Cboe DR site is designed to support Cboe Summary Depth in the following DR scenarios:

1. **Hardware failure in the primary Cboe Summary Depth system.** Customers always have the option to switch to the DR site at will if they don't want to wait for remediation of failed hardware in the Primary datacenter to take place.
2. **Failover of the Cboe platform to the DR site.** The Cboe Summary Depth feed in the Primary site currently only connects up to PITCH within the Primary site. If the respective exchange fails over to the DR site, then the Cboe Summary Depth feed out of Primary site will not be able to disseminate updates from the exchange's PITCH feed in the DR site. In this scenario, customers should take the Cboe Summary Depth feed out of the DR site to regain full coverage.
3. **Loss of the Cboe Summary Depth input feed.** Similarly to scenario #2, if Cboe experiences a PITCH dissemination issue at the Primary site, but PITCH market data is valid at the DR site, then Cboe Summary Depth customers have the option to switch over to the DR Cboe Summary Depth feed to regain full coverage.

Protocol

Cboe users may use the Cboe Summary Depth protocol over TCP/IP and/or multicast to receive the Cboe Summary Depth feed direct from Cboe.

Cboe Summary Depth cannot be used to enter orders. For order entry, refer to the Cboe FIX or BOE specifications.

Message Format

The messages that make up the Cboe Summary Depth protocol are delivered using **Sequenced Unit Header** message header which handles sequencing and delivery integrity. All messages delivered via TCP/IP or multicast use the **Sequenced Unit Header** message 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** message header with associated data.

TCP/IP delivered events from the Cboe Summary Depth Server or GRP may cross frames as the data will be delivered as a stream of data with the TCP/IP stack controlling Ethernet framing.

The Cboe Summary Depth 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. Members 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.

Data Types

The following field types are used within Cboe Summary Depth 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).
- **Binary 4.4 Price** fields are unsigned Little Endian encoded 4 byte binary fields with 4 implied decimal places (denominator = 10,000).
- **Binary 8.4 Price** fields are unsigned 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.

Message Framing

Depth of book update messages will be combined into a 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** message header. Framing will be determined by the server for feed and site. The content of the multicast across feeds will be identical, but **framing will not be consistent across feeds**. Receiving processes that receive and arbitrate multiple feeds cannot use frame level arbitration.

Sequenced Unit Header Message Fields

The **Sequence Unit Header** message header is used for all Summary Depth messages delivered via multicast or TCP/IP.

Sequenced and un-sequenced data may be delivered using the **Sequenced Unit Header** message header. Un-sequenced headers will have a 0 value for the sequence field and potentially for the unit field.

Sequenced messages have implied sequences with the first message having the sequence number contained in the header. Each subsequent message will have an implied sequence one greater than the previous message up to a maximum of count messages. Multiple messages can follow a **Sequenced Unit Header** message header, but a combination of sequenced and un-sequenced messages cannot be sent with one header.

The sequence number for the first message in the next frame can be calculated by adding the *Hdr Count* field to the *Hdr Sequence*. This technique will work for sequenced messages and heartbeats.

Table 1. Sequenced Unit Header

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|------------------------|--------|--------|------------|--|
| Hdr Length | 0 | 2 | Binary | Length of entire block of messages. Includes this header and messages following <i>Hdr Count</i> . |
| Hdr Count | 2 | 1 | Binary | Number of messages to follow this header. |
| Hdr Unit | 3 | 1 | Binary | Unit that applies to messages included in this header (0 for all Cboe Summary Depth messages). |
| Hdr Sequence | 4 | 4 | Binary | Sequence of first message to follow this header. |
| Total Length = 8 bytes | | | | |

Heartbeat Messages

The **Sequenced Unit Header** message header with a *Hdr Count* field set to 0 will be used for heartbeat messages. During trading hours heartbeat messages will be sent from the Cboe Summary Depth Server, GRP and all multicast addresses if no data has been delivered within 1 second. Heartbeat messages never increment the sequence number, but can be used to detect gaps on the real-time multicast channels during low update rate periods.

Heartbeats on the real-time multicast addresses during trading hours will have a *Hdr Sequence* value equal to the sequence of the next sequenced message to be sent. Heartbeats on gap multicast addresses will always have the *Hdr Sequence* field set to 0. All heartbeat messages sent to and from the Cboe Summary Depth Server and GRP are considered un-sequenced and should have *Hdr Sequence* and *Hdr Unit* fields set to 0.

Outside of trading hours Cboe sends heartbeat messages on all real-time and gap 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.

Cboe expects heartbeat messages to be sent to the Cboe Summary Depth Server and GRP on live connections no less than every 5 seconds. Failure to receive 2 consecutive heartbeat messages will result in the termination of the client connection.

Cboe Summary Depth Server Session Messages (TCP)

The following messages are used for initializing a TCP/IP connection to the Cboe Summary Depth Server. Members only need to implement the following messages if a TCP/IP connection to the Cboe Summary Depth Server is desired. The following messages will not be delivered using multicast.

See [Cboe Summary Depth Update Messages \(UDP & TCP\)](#) on page 20 for a description of book and market related messages that are available from the Cboe Summary Depth Server.

Cboe Summary Depth Login Message Fields

The Cboe **Summary Depth Login** message is the first message sent to the server by a member's process after its connection to the server is established. Failure to login before sending any other message type will result in the connection being dropped by the server.

Table 2. Cboe Summary Depth Login

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|-------------------------|--------|--------|--------------|---|
| Length | 0 | 1 | Binary | Length of this message including this field. |
| Message Type | 1 | 1 | 0xA0 | Cboe Summary Depth Login message |
| SessionSubId | 2 | 4 | Alphanumeric | <i>SessionSubId</i> supplied by Cboe. |
| Username | 6 | 4 | Alphanumeric | <i>Username</i> supplied by Cboe |
| Filler | 10 | 2 | Alphanumeric | (space filled) |
| Password | 12 | 10 | Alphanumeric | <i>Password</i> supplied by Cboe |
| Next Sequence | 22 | 4 | Binary | Sequence number of the next sequenced message expected by the user. |
| Total Length = 26 bytes | | | | |

Login Response Message Fields

The **Login Response** message is sent by the server to a member'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.

Table 3. Login Response

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|-------------------------------|--|--------|--------------|---|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0x02 | Login Response message |
| Status | 2 | 1 | Alphanumeric | Accepted or reason for reject |
| Total Length = 3 bytes | | | | |
| Login Response - Status Codes | | | | |
| 'A' | Login Accepted | | | |
| 'N' | Not authorized (Invalid Username/Password) | | | |
| 'B' | Session in use | | | |
| 'S' | Invalid Session | | | |
| 'Q' | Next Sequence is ahead of sequence | | | |

Replay Complete Message Fields

The **Replay Complete** message is sent to indicate that messages related to refreshing the state of the Cboe book have been delivered. After receipt of the **Replay Complete** message, message updates will be sent on the session as needed until the client disconnects.

Market Status, **RPI**, **Trading Status**, and **ADAP** messages will be sent as needed to replay the current state of the Cboe book.

During the replay phase of the connection all messages with the exception of **Trade** and **Trade Break** messages will be un-sequenced.

Table 4. Replay Complete

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|------------------------|--------|--------|------------|---|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0xA1 | Replay Complete message |
| Sequence | 2 | 4 | Binary | Sequence number that reflects that last update on the feed. |
| Total Length = 6 bytes | | | | |

Cboe Summary Depth Gap Request Proxy Session Messages (TCP)

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

GRP Login Message Fields

The **GRP 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.

Table 5. GRP Login

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|-------------------------|--------|--------|--------------|---|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0x01 | GRP Login message |
| SessionSubId | 2 | 4 | Alphanumeric | <i>SessionSubId</i> supplied by Cboe. |
| Username | 6 | 4 | Alphanumeric | <i>Username</i> supplied by Cboe |
| Filler | 10 | 2 | Alphanumeric | (space filled) |
| Password | 12 | 10 | Alphanumeric | <i>Password</i> supplied by Cboe |
| Total Length = 22 bytes | | | | |

Login Response Message Fields

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.

Table 6. Login Response

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|-------------------------------|--|--------|--------------|---|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0x02 | Login Response message |
| Status | 2 | 1 | Alphanumeric | Accepted or reason for reject |
| Total Length = 3 bytes | | | | |
| Login Response - Status Codes | | | | |
| 'A' | Login Accepted | | | |
| 'N' | Not authorized (Invalid Username/Password) | | | |
| 'B' | Session in use | | | |
| 'S' | Invalid Session | | | |

Gap Request Message Fields

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.

The **Gap Request** message for Cboe Summary Depth is identical to the Multicast PITCH **Gap Request** message. The *Unit* field should be set to 0 since the Cboe Summary Depth feed is not unitized.

Table 7. Gap Request

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|------------------------|--------|--------|------------|--|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0x03 | Gap Request message |
| Unit | 2 | 1 | Binary | <i>Unit</i> that the gap is requested for (0 for Cboe Summary Depth implementation). |
| Sequence | 3 | 4 | Binary | <i>Sequence</i> of first message (lowest sequence in range). |
| Count | 7 | 2 | Binary | <i>Count</i> of messages requested |
| Total Length = 9 bytes | | | | |

Gap Response Message Fields

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.

The **Gap Response** message for Cboe Summary Depth is identical to the Multicast Pitch **Gap Response** message. The *Unit* field should be set 0 to since the Cboe Summary Depth feed is not unitized.

Table 8. Gap Response

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|-----------------------------|--|--------|--------------|--|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0x04 | Gap Response message |
| Unit | 2 | 1 | Binary | <i>Unit</i> the gap was requested for (0 for Cboe Summary Depth implementation). |
| Sequence | 3 | 4 | Binary | <i>Sequence</i> of first message in request. |
| Count | 7 | 2 | Binary | <i>Count</i> of messages requested |
| Status | 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.

Cboe Summary Depth Update Messages (UDP & TCP)

The messages described in this section are delivered from the Cboe Summary Depth Server (TCP), Cboe Summary Depth Feed Server (UDP), and the Cboe Summary Depth Gap Server (UDP gap responses).

Clear Quote Message Fields

The **Clear Quote** message instructs feed recipients to clear all quotes and ADAP information for the specified symbol. This message does not affect the executed volume of the symbol.

Table 9. Clear Quote

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|--------------------------|--------|--------|--------------|--|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0xA2 | Clear Quote message |
| Last Update Timestamp | 2 | 8 | Binary | Timestamp of the last matching engine message that updated the quote information for this symbol in the Cboe Summary Depth server's cache. Encoded as the number of nanoseconds since midnight. |
| Symbol | 10 | 8 | Alphanumeric | Symbol right padded with spaces. |
| Reserved | 18 | 1 | Alphanumeric | Reserved |
| Total Length = 19 bytes | | | | |

Cboe Market Status Message Fields

The **Cboe Market Status** message is disseminated to reflect a change in the status of the market. The market should be assumed to be Normal unless otherwise indicated by a **Cboe Market Status** message.

The Incomplete market status is used to indicate that the feed has not delivered updates for all of the market center’s symbols and that the feed is transitioning to Normal. If the market center transitions from Normal to Excluded the feed will deliver updates of symbol quote/ADAP information to properly reflect the state of the book. At the start of the market center’s transition from Excluded to Normal a **Cboe Market Status** message will be sent with Incomplete for the market center’s status. Symbol quote/ADAP information will then be sent for all applicable symbols. Once the market center’s symbol information has been disseminated a **Cboe Market Status** message will be delivered with a Normal market status.

Table 10. Cboe Market Status

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|-------------------------|--------|--------|--------------|---|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0xA6 | Cboe Market Status message |
| Timestamp | 2 | 8 | Binary | Timestamp of when the Market Status changed, encoded as the number of nanoseconds since midnight. |
| Reserved | 10 | 1 | Alphanumeric | Reserved |
| Market Status | 11 | 1 | Alphanumeric | N = Normal E = Excluded from ADAP updates I = Incomplete |
| Session Indicator | 12 | 1 | Alphanumeric | R = Regular trading session P = Pre- or post-market session |
| Total Length = 13 bytes | | | | |

ADAP Message Fields

Each **ADAP** message delivers one or more updates for a Symbol's ADAP book. Each **ADAP** message contains one or more **ADAP Blocks** messages. A receiving process should interpret each **ADAP Block** message as a replacement for any previously delivered **ADAP Blocks** messages at that price level.

A quantity of 0 indicates that the price level is either no longer available or the price level is not within the number ADAP levels maintained by the feed. In either case a receiving process should delete a price level with a 0 quantity from its cache.

Table 11. ADAP

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|--------------------------|--------|--------|--------------|---|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field |
| Message Type | 1 | 1 | 0xA7 | ADAP message |
| Last Update Timestamp | 2 | 8 | Binary | Timestamp of the last matching engine message that updated the quote information for this symbol in the Cboe Summary Depth server's cache. Encoded as the number of nanoseconds since midnight. |
| Symbol | 10 | 8 | Alphanumeric | Symbol right padded with spaces. |
| Flags | 18 | 1 | Bit Field | Bit 0: Clear ADAP 0 : Keep any ADAP information for this symbol. 1 : Delete any ADAP information for this symbol before applying ADAP Blocks . Bit 1: ADAP Complete 0 : The ADAP view for this symbol is complete. 1 : More ADAP updates for this symbol to follow in another ADAP message. Bit 2 : Short/Long Block(s) 0 : Short Update ADAP Block (s) to follow 1 : Long Update ADAP Block (s) to follow Bits 3-7: Spare |
| Spare | 19 | 1 | | Spare |
| ADAP Blocks | 20 | 1 | Binary | Number of ADAP Blocks messages to follow |
| ADAP Block Size | 21 | 1 | Binary | Size of each ADAP Block message |
| Header Length = 22 bytes | | | | |

Table 12. Short Update ADAP Block

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|----------|--------|--------|--------------|--------------|
| Reserved | 0 | 1 | Alphanumeric | Reserved |
| Side | 1 | 1 | Alphanumeric | B = Buy Side |

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|--|--------|--------|------------------|---|
| | | | | S = Sell Side |
| Price | 2 | 4 | Binary 4.4 Price | Price level to add/update in the ADAP book. |
| Quantity | 6 | 4 | Binary | Quantity of shares at this price level in the ADAP book. A value of zero implies deletion of this ADAP level. |
| Short ADAP Block Length Indicated by ADAP Block Size in Header | | | | |

Table 13. Long Update ADAP Block

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|--|--------|--------|------------------|---|
| Reserved | 0 | 1 | Alphanumeric | Reserved |
| Side | 1 | 1 | Alphanumeric | B = Buy Side S = Sell Side |
| Price | 2 | 8 | Binary 8.4 Price | Price level to add/update in the ADAP book. |
| Quantity | 10 | 8 | Binary | Quantity of shares at this price level in the ADAP book. A value of zero implies deletion of this ADAP level. |
| Long ADAP Block Length Indicated by ADAP Block Size in Header | | | | |
| Total Length = Variable → (Header Length [22 bytes] + ADAP Blocks x ADAP Block Size) | | | | |

Retail Price Improvement (RPI) Message Fields

The **Retail Price Improvement (RPI)** message is a retail liquidity indicator that includes symbol and side, but not price and size. An **RPI** message will be disseminated when there is a retail price improving order present for a symbol on the Cboe Exchange order book OR to indicate a RPI order is no longer available. RPI orders offer price improvement in increments of \$.001 to Retail Member Organizations.

Table 14. RPI

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|--------------------------|--------|--------|--------------|--|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0xA8 | RPI message |
| Timestamp | 2 | 8 | Binary | Timestamp of the matching engine RPI message, encoded as the number of nanoseconds since midnight. |
| Symbol | 10 | 8 | Alphanumeric | Symbol right padded with spaces. |
| Reserved | 18 | 1 | Alphanumeric | Reserved |
| Retail Price Improvement | 19 | 1 | Alphanumeric | B = Buy Side RPI S = Sell Side RPI A = Buy & Sell RPI N = No RPI |
| Total Length = 20 bytes | | | | |

Trade Message Fields

Trade messages are sent when an order is executed in whole or in part on the Cboe exchange. The last-sale eligible status (*Flags* field Bit 1) is derived based on four criteria:

1. The *Session Indicator* (see **Cboe Market Status** message) must be in the regular session.
2. The *Last Quantity* must be at least one round lot.
3. The *Market Status* (see **Cboe Market Status** message) for the executing exchange must be Normal.
4. The *Transaction Time* of the event must be within 10 seconds of the current time.

Table 15. Trade

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|---------------------------------|--------|--------|------------------|---|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0xA9 | Trade message |
| Transaction Time | 2 | 8 | Binary | The time the trade occurred, encoded as the number of nanoseconds since midnight. |
| Symbol | 10 | 8 | Alphanumeric | Symbol right padded with spaces. |
| Reserved | 18 | 1 | Alphanumeric | Reserved |
| Execution ID | 19 | 8 | Binary | Execution identifier of this trade. <i>Execution ID</i> is also referenced in the Trade Break message. |
| Last Price | 27 | 8 | Binary 8.4 Price | Last trade price. |
| Last Quantity | 35 | 8 | Binary | Last trade quantity. |
| Cboe Cumulative Executed Volume | 43 | 8 | Binary | Cumulative number of shares traded today. |
| Reserved | 51 | 8 | Binary | Reserved |
| Flags | 59 | 1 | Bit Field | Bit 0: Reserved. Bit 1: 0 : Trade is not last-sale eligible 1 : Trade is last-sale eligible Bits 2-7: Reserved. |
| Total Length = 60 bytes | | | | |

Trade Break Message Fields

The **Trade Break** message is sent whenever an execution on the Cboe exchange is broken. Trade breaks are rare and only affect applications that rely upon Cboe execution based data.

Table 16. Trade Break

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|---------------------------------|--------|--------|--------------|---|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0xAA | Trade Break message |
| Transaction Time | 2 | 8 | Binary | The time the trade break occurred, encoded as the number of nanoseconds since midnight. |
| Symbol | 10 | 8 | Alphanumeric | Symbol right padded with spaces. |
| Reserved | 18 | 1 | Alphanumeric | Reserved |
| Execution ID | 19 | 8 | Binary | Execution identifier of trade to be broken. |
| Cboe Cumulative Executed Volume | 27 | 8 | Binary | Cumulative number of shares traded today. |
| Reserved | 35 | 8 | Binary | Reserved |
| Flags | 43 | 1 | Bit Field | Bits 0-7: Reserved. |
| Total Length = 44 bytes | | | | |

Trading Status Message Fields

The **Trading Status** message is used to indicate the current trading status of a security on the Cboe exchange. A **Trading Status** message is whenever a security's trading status changes.

Sequenced **Trading Status** messages are sent upon system start up for all active securities with Trading Status=S (suspended). **Trading Status** messages will continue to be published upon symbol state changes, such as at the beginning of order acceptance.

A **Trading Status** message will be sent:

- for Regulatory H=Halts in any security as well as the T=Tradingresumption for the same security.
- for Cboe Listed securities that are in a Q=Quoting period for auctions.
- to indicate a Reg SHO price test is in effect.

Table 17. Trading Status

| FIELD | OFFSET | LENGTH | VALUE/TYPE | DESCRIPTION |
|-------------------------|--------|--------|--------------|---|
| Length | 0 | 1 | Binary | <i>Length</i> of this message including this field. |
| Message Type | 1 | 1 | 0xAB | Trading Status message |
| Timestamp | 2 | 8 | Binary | Timestamp of the matching engine trading status message, encoded as the number of nanoseconds since midnight. |
| Symbol | 10 | 8 | Alphanumeric | Symbol right padded with spaces. |
| Reserved | 18 | 1 | Alphanumeric | Reserved |
| Halt Status | 19 | 1 | Alphanumeric | A = Accepting Orders for Queuing H = Halted Q = Quote-Only S = Exchange Specific Suspension T = Trading |
| Reg SHO Action | 20 | 1 | Alphanumeric | 0 = No price test in effect 1 = Reg SHO price test restriction in effect |
| Total Length = 21 bytes | | | | |

Multicast Configuration

US Equities Production Environment Configuration

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/TYPER | LIMIT/SETTING | NOTES |
|--------------------|--------------------|---|
| MTU | 1500 | Cboe will send UDP messages up to 1500 bytes. Members should ensure that their infrastructure is configured accordingly. |
| 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 Summary Depth Multicast Routing Parameters

Table 18. Cboe Summary Depth Multicast Routing Parameters

| DATA CENTER | RENDEZVOUS POINT |
|-----------------------|------------------|
| Primary Data Center | 74.115.128.156 |
| Secondary Data Center | 174.136.181.252 |

Cboe Summary Depth Multicast Addresses

The following tables describe the distribution across production multicast Cboe Summary Depth feeds.

Table 19. Primary Datacenter

| PRIMARY DATACENTER 174.136.169.8/29 | REAL-TIME MC | GAP RESP. MC |
|--|---------------------|---------------------|
| Cboe Summary Depth BZX | 224.0.131.136:32202 | 224.0.131.137:32202 |
| Cboe Summary Depth BYX | 224.0.131.138:32203 | 224.0.131.139:32203 |
| Cboe Summary Depth EDGA | 224.0.131.140:32204 | 224.0.131.141:32204 |
| Cboe Summary Depth EDGX | 224.0.131.142:32205 | 224.0.131.143:32205 |

Table 20. Secondary Datacenter

| SECONDARY DATACENTER 174.136.181.144/28 | REAL-TIME MC | GAP RESP. MC |
|--|-------------------|-------------------|
| Cboe Summary Depth BZX | 233.19.3.40:32202 | 233.19.3.41:32202 |
| Cboe Summary Depth BYX | 233.19.3.42:32203 | 233.19.3.43:32203 |
| Cboe Summary Depth EDGA | 233.19.3.44:32204 | 233.19.3.45:32204 |
| Cboe Summary Depth EDGX | 233.19.3.46:32205 | 233.19.3.47:32205 |

US Equities Certification Environment Configuration

Cboe Summary Depth Certification Multicast Routing Parameters

Table 21. Cboe Summary Depth Certification Multicast Routing Parameters

| DATA CENTER | RENDEZVOUS POINT |
|---------------------------------|------------------|
| Certification Data Center (NY5) | 74.115.128.129 |

Cboe Summary Depth Certification Multicast Addresses

The following tables describe the current unit distribution across certification multicast Cboe Summary Depth feeds.

Table 22. Cboe Summary Depth Certification Multicast Addresses

| PRIMARY DATACENTER 174.136.174.248/29 | REAL-TIME MC | GAP RESP. MC |
|--|--------------------|--------------------|
| Cboe Summary Depth BZX | 224.0.74.200:32200 | 224.0.74.201:32200 |
| Cboe Summary Depth BYX | 224.0.74.202:32200 | 224.0.74.203:32200 |
| Cboe Summary Depth EDGA | 224.0.74.204:32200 | 224.0.74.205:32200 |
| Cboe Summary Depth EDGX | 224.0.74.206:32200 | 224.0.74.207:32200 |

References

Symbology

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

Support

Please direct questions or comments regarding this specification to tradedesk@cboe.com.

Revision History

| DOCUMENT VERSION | DATE | DESCRIPTION |
|------------------|----------|---|
| 1.0.0 | 08/17/16 | Initial version. |
| 1.0.1 | 10/03/16 | Updated to include multicast routing addresses and bandwidth recommendation. |
| 1.0.2 | 10/17/17 | Cboe branding/logo changes. |
| 1.0.3 | 05/03/18 | Corrected support email address. Fixed typo in spelling of "certification" in section 6.2.2. |
| 1.0.4 | 07/26/21 | Updated Cboe Symbology Reference link. |
| 1.0.5 | 09/06/24 | Effective 11/13/24, Trading Status messages will be sent upon system start up rather than at the start of order acceptance. |
| 1.0.6 | 10/09/24 | Effective 11/13/24, clarified initial Trading Status messages will be sent upon system start up in addition to sending at the start of order acceptance. |
| 1.0.7 | 01/15/25 | Updated with Cboe Titanium branding. |