CBOE Application Programming Interface

CBOE API Version 3.2 - Release Notes

Provides an overview of upcoming changes in the next production release of the CMi

CBOE PROPRIETARY INFORMATION

12 August 2005

Document #[API-00]

Front Matter

Disclaimer

Support and Questions Regarding This Document

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

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

Table of Contents

Overview

This document highlights changes for the new release of the CMi API, Version 3.2. This release details upcoming IDL constant changes to support Hybrid opening enhancements, single acronym and the new auction type, hybrid automated liaison, as well as documentation changes. No simulator changes are required. Your feedback or questions regarding this document should be sent to api@cboe.com.

CMi V3.2 Highlights

Hybrid Opening Enhancements

The CMi expected opening price (EOP) constants have been enhanced to provide descriptive opening messages to be used by Market Makers to send in valid/tradable quotes at the open. In addition to the current opening process, these messages will be sent prior to the opening. The messages will be sent at pre-determined intervals, currently planned for every 30 seconds. Three new EOP type messages have been defined.

- const cmiMarketData::ExpectedOpeningPriceType **PRICE_NOT_IN_BOTR_RANGE = 10**;

  This EOP message type indicates that the potential opening price is not in the valid Best of the Rest (BOTR) range based on the current BOTR price, current potential opening quantity and the system configuration. The EOP message will provide users with the potential opening price and opening quantity.

- const cmiMarketData::ExpectedOpeningPriceType **NEED_MORE_BUYERS = 3**;

  This EOP message type indicates that the current market is imbalanced and will need more buyers' contracts (n) at the opening price. The EOP message will provide users with the potential opening price (bid price) and the imbalanced quantity (n).

- const cmiMarketData::ExpectedOpeningPriceType **NEED_MORE_SELLERS = 2**;

  This EOP message type indicates that the current market is imbalanced and will need more sellers' contracts (m) at the opening price. The EOP message will provide users with the potential opening price (ask price) and the imbalanced quantity (m).

The Hybrid openings have changed to calculate the opening range based on the midpoint of the Best quote (or I order) bid and the best quote (or I order) ask, minus/plus half the spread width for the price range. (Ex. Best bid 1.00, best ask 1.10 Opening Spread width .25 for non-LEAP option will calculate an opening range of .925 - 1.175 which will round to .90 - 1.20)

Example Price Not in BOTR range:

Opening quote range: 1.00 - 1.10 20 x 20

Order to sell 10 at the market; BOTR Range = 1.05 - 1.10

Message #10 sent to users will include the CBOE calculated opening price of 1.00 and the size of 10.

Example Need more buyers:

Opening quote range: 1.00 - 1.10 20 x 20

Order to sell 50 at the market

Message #3 sent to users will include the Bid price of 1.00 and the size of 30.

Example Need more sellers:

Opening quote range: 1.00 - 1.10 20 x 20

Order to buy 50 at the market

Message #2 sent to users will include the Ask price of 1.10 and the size of 30.

Hybrid Automated Liaison Auction Type

The auction type: AUCTION_HAL is introduced in this release. HAL stands for: Hybrid Automated Liaison. This auction type allows users to participate in auctions for orders that are:

1. **NBBO Reject**: An incoming order is marketable but CBOE is not at the NBBO.
2. **Tweener Lock**: An incoming order that is between the market at CBOE but is marketable against an away market.
3. **Tweener**: An incoming order that is between the market at CBOE and does not lock or cross an away market.

Special Considerations for the HAL Auction Type

- A Request for Price (RFP) flash will be used to place the order in an auction state. The RFP flash timer is configurable.
- If a decision to flash the order is taken then CBOEdirect will end any quote locks, quote triggers and any current non-flash auctions will be allocated prior to the start of order flashing.
- CBOEdirect will send RFP at the NBBO price (or order price for a Tweener) to all the market makers quoting in this product or class depending upon the settings.
- RFP will be of type HAL and have the NBBO price as the starting price and the size of the order.
- Response to the flash auction cannot be cancelled.
- Response to the flash auction will result in a Quote Trigger for a pre-configured time.

Single Acronym

Single acronym functionality allows multiple user IDs to share the same acronym. Firms will be allowed to use multiple user ids to quote all their classes and multiple user ids/login ids to share the same acronym and profile. The back office process will be streamlined by coupling trades that are associated with the same acronym

This functionality adds a new restriction to quote entry and changes the behavior of the quote risk monitor (QRM).

Quote Entry Restrictions

With single acronym, two users sharing the same acronym cannot quote the same class. For example,

1. User ID1 and user ID2 share the same acronym
2. User ID1 is quoting class, IBM. User ID2 tries to quote IBM but the quote is rejected. User ID2 will receive the error message **ErrorCode OTHER_USER_FOR_ACR_QUOTING_CLASS = 4150**
3. If user ID1 logs off and all quotes are cancelled, user ID2 will be able to quote IBM.
4. If user ID1 cancels all his/her quotes, without logging off, user ID2 will be able to quote IBM.

QRM

QRM will be setup at the acronym level. Therefore, multiple users that share the same acronym and exchange will be sharing the same QRM values. For example,

- User ID1 and user ID2 are sharing acronym ABC
- User ID1 is quoting IBM
- User ID2 changes QRM on his/her CBOE*direct* trading system for IBM
- User ID1's QRM values will change to reflect the changes made by User ID2

IDL Interfaces

New and modified IDL is reflected in **bold** face. Removed IDL is reflected in ~~**bold strikethrough**~~.

cmiConstants.idl

```
interface ExpectedOpeningPriceTypes
  {
    const cmiMarketData::ExpectedOpeningPriceType OPENING_PRICE = 1;
    const cmiMarketData::ExpectedOpeningPriceType MORE_BUYERS = 2; //deprecated
    const cmiMarketData::ExpectedOpeningPriceType MORE_SELLERS = 3; //deprecated
    const cmiMarketData::ExpectedOpeningPriceType NEED_MORE_SELLERS = 2;
    const cmiMarketData::ExpectedOpeningPriceType NEED_MORE_BUYERS = 3;
    const cmiMarketData::ExpectedOpeningPriceType NO_OPENING_TRADE = 4;
    const cmiMarketData::ExpectedOpeningPriceType MULTIPLE_OPENING_PRICES = 5;
    const cmiMarketData::ExpectedOpeningPriceType NEED_QUOTE_TO_OPEN = 6;
    const cmiMarketData::ExpectedOpeningPriceType PRICE_NOT_IN_QUOTE_RANGE = 7;
    const cmiMarketData::ExpectedOpeningPriceType NEED_DPM_QUOTE_TO_OPEN = 8;
    const cmiMarketData::ExpectedOpeningPriceType DPM_QUOTE_INVALID = 9;
    const cmiMarketData::ExpectedOpeningPriceType PRICE_NOT_IN_BOTR_RANGE = 10;
    const cmiMarketData::ExpectedOpeningPriceType NEED_MORE_BUYERS = 11;
    const cmiMarketData::ExpectedOpeningPriceType NEED_MORE_SELLERS = 12;
  };
```

This interface provides the Expected Opening Price Types that are supported. Expected Opening Price Types: MORE_BUYERS=2 and MORE_SELLERS=3 have been deprecated. The ExpectedOpeningPriceType NO_OPENING_TRADE=4 will only be sent if the previous message was something other than value 4.

```
interface AuctionTypes
  {
    const cmiOrder::AuctionType AUCTION_INTERNALIZATION =1;
     const cmiOrder::AuctionType AUCTION_STRATEGY =2;
     const cmiOrder::AuctionType AUCTION_REGULAR_SINGLE =3;  /* not currently used */
     const cmiOrder::AuctionType AUCTION_HAL = 4; /* not currently used*/
     const cmiOrder::AuctionType AUCTION_UNSPECIFIED = 0;    /* not currently used */
  };
```

This interface provides the types of auction codes that are supported.

CMi Error Codes

```
interface NotAcceptedCodes  {
     const exceptions::ErrorCode UNKNOWN_TYPE = 4000;
     const exceptions::ErrorCode INVALID_STATE = 4010;
     const exceptions::ErrorCode INVALID_REQUEST = 4020;
     const exceptions::ErrorCode QUOTE_RATE_EXCEEDED = 4030;
     const exceptions::ErrorCode RATE_EXCEEDED = 4040;
     const exceptions::ErrorCode SEQUENCE_SIZE_EXCEEDED = 4050;
     const exceptions::ErrorCode QUOTE_BEING_PROCESSED = 4060;
     const exceptions::ErrorCode ORDER_BEING_PROCESSED = 4070;
     const exceptions::ErrorCode EXCHANGE_CLASS_GATE_CLOSED = 4080;
     const exceptions::ErrorCode SERVER_NOT_AVAILABLE = 4090;
     const exceptions::ErrorCode ACTION_VETOED = 4100;
     const exceptions::ErrorCode QUOTE_CONTROL_ID = 4110;
    const exceptions::ErrorCode UNSUPPORTED_INTERNALIZATION = 4120;  /* not used */
     const exceptions::ErrorCode AUCTION_INACTIVE = 4130;  / *not used*/
     const exceptions::ErrorCode AUCTION_ENDED = 4140;
     const exceptions::ErrorCode OTHER_USER_FOR_ACR_QUOTING_CLASS = 4150;
  };
```

This interface provides the Not Accepted error codes that are supported. ErrorCode OTHER_USER_FOR_ACR_QUOTING_CLASS = 4150 is new in this release. If a class is being quoted by a user, and any other user sharing the acronym tries to send a quote for the same class, the quote will be rejected with this error code.

Document Changes

API-01

- No changes

API-02

- Created a new section called: Options and Futures Clearing Information. Included in the document are the two tables below.

CMi Options Order Clearing Information in cmiOrder::OrderEntryStruct

Field Name	Sample	Description
account	ABC or QAB	For market-makers, this typically would be either the joint account (often called q-account) or the market-maker three-letter badge acronym. Passed through to OCC. Required for Market-Maker and DPM roles in all sessions. For Market-Maker and DPM roles, CBOE validates the value of this field on inbound orders against the CBOE Membership system. For Market-Maker and DPMs, user cannot use more than one account per class. Optional for Broker-Dealer and Firm roles. CBOE performs no validation checks on the value of this field for Broker-Dealer and Firm roles. Exact size is 3 and data type is alpha only.
cmta	*Exchange*: CBOE *Firm#*: 123	The CMTA (Clearing Member Trade Agreement) field is used to designate an OCC clearing firm if it is different from the executingOrGiveUpFirm. CBOE performs no validation checks on the CMTA field against the CBOE Membership system. This field is optional for all roles in all sessions. CMTA is comprised of two components: an e*xchangeFirmStruct* which contains the exchange code and CMTA *firmNumber*. If you use CMTA, then you must use submit both of these two components. The e*xchange string* is the exchange on which your order will trade. The exchange portion of the CMTA field is alpha only. The *firmNumber* is the OCC clearing firm where the order will clear. The firmNumber portion of the CMTA field is numeric only. Even though the maximum size for the firmNumber component is 5, CBOE will read the first three numbers of this field to use as the OCC clearing firm. In other words, if the desired CMTA firm at the OCC is "123", do not send "00123", send "123".
correspondentFirm	ABC or ABCD	The correspondent firm field is used by the executing give up firm to differentiate the firm or system sending the order. The 1st three characters of this field are mapped to the optional data field on the CBOE Trade Match (CTM) record. This field has no impact on the clearing of the trade. This field is optional. CBOE performs no validation checks on the correspondentFirm field against the CBOE Membership system. Maximum size is 4 characters and data type is uppercase alpha only.

Field Name	Sample	Description
executingOrGiveUpFirm	CBOE:123 or 123	This is the CBOE clearing firm that is representing the order in live trading (post trade processing firm). If no CMTA firm is present in the order, then the executingOrGiveUpFirm represents the OCC clearing firm where the order will clear. This field is required for all orders sent to CBOE for all roles in all sessions regardless of whether a CMTA firm is given or not. CBOE performs validation checks of executingOrGiveUpFirm against the CBOE Membership system on options orders routed to the W_MAIN session. Broker-Dealer and Firm roles must choose from a list of pre-approved and pre-configured executingOrGiveUpFirms and the Market-Maker and DPM roles must use default executingOrGiveUpFirm only. executingOrGiveUpFirm is comprised of two components: an *ExchangeFirmStruct* which contains the Exchange code and *firmNumber*. The *Exchange string* is the exchange on which your order will trade. The Exchange portion of the executingOrGiveUpFirm field is alpha only. If there is no CMTA given in the order, then the executingOrGiveUpFirm *firmNumber* will be the OCC clearing firm where the order will clear. If there is a CMTA given in the order, then the firmNumber is the CBOE clearing firm that is representing the order in live trading (post trade processing firm). The firmNumber portion of the CMTA field is numeric only. Even though the maximum size for the firmNumber component is 5, CBOE will only read the first three numbers of this field to use as the executingOrGiveUpFirm. In other words, if the desired executingOrGiveUpFirm firm is “123”, do not send “00123”, send “123”.

Field Name	Sample	Description
optionalData	M:ABC ABC123ABC	**Orders of origin Customer ("C"):** The first four characters are reported to the last four characters of CBOE Trade Match Optional Data field. These four characters are reported to OCC. Do not put "C:" in this field. **Orders of origin In-Crowd Market-maker ("I")** This field is not required. Do not put "I:" in this field. In-Crowd Market-Maker (ICM) options orders of origin 'I', similar to two-sided quotes, take their clearing information from the market-maker profile in the CBOE System Administrator GUI (SAGUI). **Orders of origin Market-maker ("M") and Away Market-maker ("N")** This field is *required* for all options orders of origin "M" or "N" that are sent to the CBOE. This contains data that will be passed on to the CBOE Trade Match system (CTM) and will be part of clearing information sent to the OCC. The data is specific to each member firm. For "M" and "N" orders routed to the CBOE Trading Floor (W_MAIN session), this field should contain the Market Maker Account (Q Account, joint account, or market maker acronym). If a subaccount is also used, then it must be supplied as well. If a firm sends an origin of "N", then this field must begin with the characters "M:", not "N:". **Market-Maker format:** `M:ABC ABC123ABC` **M: in characters 1 and 2 (including the colon)** **ABC in characters 3 – 5**. This value should contain the market maker account, Q-account, or joint account where the trade will clear. This value will be moved to the Market Maker Account field on the Trade Match record. The account portion of optional data in characters 3-5 is required for all "M" and "N" orders or the trade will not clear properly. **(Space) in character 6** **ABC123 in characters 7 – 12** This is the firm's back office subaccount field. Not a required field. If present, it will be moved to characters 6–11 of the Trade Match Optional Data field. (Same as a Market-Maker Terminal trade). If a subaccount is present, it may be alpha-numeric and it may be between 1-6 characters. The subaccount portion of this must be left alligned within these six characters. **ABC in characters 13-15** For orders of origin "I", "M", and "N", these three characters should contain the three-letter market-maker originator acronym. *Please see the "originator" field for more detailed requirements information.*

Field Name	Sample	Description
		Examples **M:ABC ABC123ABC** Market-maker ABC has no q-account or joint account, but has subaccount ABC123. **M:QAB AB1 ABC** Market-maker ABC has q-account (joint account) QAB with subaccount AB1. Note that there are three spaces between AB1 and ABC. **M:QAB ABC** Market-maker ABC has q-account (joint account) QAB with no subaccount. Note that there are seven spaces between QAB and ABC. **M:QAB** Market-maker ABC has q-account (joint account) QAB with no subaccount. **M:ABC** Market-maker ABC has no q-account or joint account, and has no subaccount. **Preferred DPM** Firms that give one DPM priority in participating in a trade use this field. Firm is specified as **P:*firm*;** and can coexist with other data that may be present in this field. *"Firm"* is the CBOE firm acronym as listed in the Order Test Plan. Please note that the colon : and semi-colon ; are both mandatory. **Linkage**: This field is *not* used for Linkage.
originator	ABC	This field would only be used for orders of origin "M", "I", and "N". This field is comprised of two components: an *Exchange* string which contains the Exchange code and *acronym*. The *Exchange* string is the exchange on which the market-maker will clear the trade. The Exchange portion of the originator field is alpha only. The *acronym* is the three-letter acronym ("badge") of the market-maker who originates the order. This field will typically be three characters (occasionally two). **Orders of origin "M"** Orders of origin "M" entered by a broker-dealer role must supply the three-letter acronym ("badge") of the market-maker. If a broker-dealer role submits an options order of origin "M" on behalf of a CBOE market-maker, then the three-letter (all alpha, all caps) MM acronym must go into either the originator field (tag 9465 in FIX) or the originator portion (positions 13-15) of the optional data field (tag 9324 in FIX). The market-maker role does not have to enter the originator field when entering orders of origin "M". However, if a market-maker role wishes to enter the originator field when entering orders of origin "M", then it may enter the originator acronym into either the originator field (tag 9465 in FIX) or the originator portion (positions 13-15) of the optional data field (tag 9324 in FIX).

Field Name	Sample	Description
		Orders of origin "I" Orders of origin "I" entered by a broker-dealer role must supply the three-letter acronym ("badge") of the market-maker. If a broker-dealer role submits an options order of origin "I" on behalf of a CBOE market-maker, then the three-letter (all alpha, all caps) MM acronym must go into the originator field (tag 9465 in FIX). If a broker-dealer role submits an options order of origin "I" on behalf of a CBOE market-maker, then it may also if it wishes put the originator acronym in the originator portion (positions 13-15) of the optional data field (tag 9324 in FIX). The market-maker role does not have to enter the originator field when entering orders of origin "I". However, if a market-maker role wishes to enter the originator field when entering orders of origin "I", then it may enter the originator acronym into either the originator field (tag 9465 in FIX) or the originator portion (positions 13-15) of the optional data field (tag 9324 in FIX). **Orders of origin "N"** Orders of origin "N" entered by a broker-dealer role must supply the three-letter acronym ("badge") of the market-maker. If a broker-dealer role submits an options order of origin "N" on behalf of a non-CBOE market-maker (e.g. a CBOE BD role enters an order on behalf of an AMEX market-maker), then the MM acronym must go into the originator portion (positions 13-15) of the optional data field (tag 9324 in FIX) and not the originator field (tag 9465 in FIX). If a broker-dealer role submits an options order of origin "N" on behalf of a non-CBOE market-maker at another exchange, and enters any value into the originator field (tag 9465 in FIX), then CBOE will reject the order. CBOE does not allow the market-maker role to enter orders of origin "N".
subaccount	AB2, ABC, ABC123, QA12, QAB123	CBOE performs no validation checks on subaccount against the CBOE Membership system. It is optional for all roles. Maximum size is 6 and data type is alphanumeric. For Broker and Firm roles, if subaccount is used then the account field is not required.

CMi Futures Order Clearing Information in cmiOrder::OrderEntryStruct

Field Name	Sample	Description
account	ABC or QAB	For market-makers, this typically would be either the joint account (often called q-account) or the market-maker three-letter badge acronym. Passed through to OCC. Required for Market-Maker and DPM roles in all sessions. For Market-Maker and DPM roles, CBOE validates the value of this field on inbound orders against the CBOE Membership system. For Market-Maker and DPMs, user cannot use more than one account per class. If desired, this field can be the same value as subaccount. Optional for Broker-Dealer and Firm roles. CBOE performs no validation checks on the value of this field for Broker-Dealer and Firm roles. Exact size is 3 and data type is alpha only.

Field Name	Sample	Description
cmta	*Exchange*: CBOE *Firm#*: 123	The CMTA (Clearing Member Trade Agreement) field is used to designate an OCC clearing firm if it is different from the executingOrGiveUpFirm. CBOE performs no validation checks on the CMTA field against the CBOE Membership system. This field is optional for all roles in all sessions. CMTA is comprised of two components: an e*xchangeFirmStruct* which contains the exchange code and CMTA *firmNumber*. If you use CMTA, then you must use submit both of these two components. The e*xchange string* is the exchange on which your order will trade. The exchange portion of the CMTA field is alpha only. The *firmNumber* is the OCC clearing firm where the order will clear. The firmNumber portion of the CMTA field is numeric only. Even though the maximum size for the firmNumber component is 5, CBOE will read the first three numbers of this field to use as the OCC clearing firm. In other words, if the desired CMTA firm at the OCC is "123", do not send "00123", send "123".
correspondentFirm	ABC or ABCD	The correspondent firm field is used by the executing give up firm to differentiate the firm or system sending the order. The 1st three characters of this field are mapped to the optional data field on the CBOE Trade Match (CTM) record. This field has no impact on the clearing of the trade. This field is optional. CBOE performs no validation checks on the correspondentFirm field against the CBOE Membership system. Maximum size is 4 characters and data type is uppercase alpha only.
executingOrGiveUpFirm	*Exchange:* CBOE, CFE, or ONE *Firm:*123	This is the CBOE clearing firm that is representing the order in live trading (post trade processing firm). If no CMTA firm is present in the order, then the executingOrGiveUpFirm represents the OCC clearing firm where the order will clear. This field is required for all orders sent to CBOE for all roles in all sessions regardless of whether a CMTA firm is given or not. CBOE performs validation checks of executingOrGiveUpFirm against the CBOE Membership system on options orders routed to all sessions. Broker-Dealer and Firm roles must choose from a list of pre-approved and pre-configured executingOrGiveUpFirms and the Market-Maker and DPM roles must use default executingOrGiveUpFirm only. This field is comprised of two components: an *ExchangeFirmStruct* which contains the Exchange code and *firmNumber*. The *Exchange string* is the exchange on which your order will trade. The Exchange portion of the executingOrGiveUpFirm field is alpha only. If there is no CMTA given in the order, then the executingOrGiveUpFirm *firmNumber* will be the OCC clearing firm where the order will clear. If there is a CMTA given in the order, then the firmNumber is the CBOE clearing firm that is representing the order in live trading (post trade processing firm). The firmNumber portion of the CMTA field is numeric only. Even though the maximum size for the firmNumber component is 5, CBOE will only read the first three numbers of this field to use as the executingOrGiveUpFirm. In other words, if the desired executingOrGiveUpFirm firm is "123", do not send "00123", send "123".
optionalData	ThisIsBob sOrder12	This field stays with the order for the life of the order. The first 16 bytes go to the OCC. This field is optional for all roles for futures orders. Maximum size is 128 characters and data type is alphanumeric. Do not send "M:" account information like is used for options clearing.

Field Name	Sample	Description
Originator	*Exchange:* CBOE, CFE, or CME *Acronym:* ABC	This field is used for market-maker orders only. It is optional for OneChicago futures but not used for CFE futures. It would contain the three letter market-maker acronym. This field is comprised of two components: an *Exchange* string which contains the Exchange code and *acronym*. The *Exchange* string is the exchange on which the market-maker will clear the trade. The Exchange portion of the originator field is alpha only. The *acronym* is the three-letter acronym ("badge") of the market-maker who originates the order. This field will typically be three characters (occasionally two).
Subaccount	AB2, ABC, ABC123, QA12, QAB123	Subaccount is required for CFE and OneChicago futures. It specifies the account into which the trade will clear. CBOE performs no validation checks on subaccount against the CBOE Membership system. Maximum size is 6 and data type is alphanumeric.

- Changed the fourth paragraph in the section, "Callback Performance Design Issues" to read:

  The solution for this problem is to balance the load across multiple callback objects. Two different ways of accomplishing this might be to use a pool of callback objects or to have a distinct callback object per product class. It should be noted that each user session has a limited number of threads associated with it. If the CAS is trying to service too many callback objects it is possible for some high volume messages to become thread starved. *CBOE suggest that you subscribe to market data by class for recap and current market. Note that book depth must be by product (not class) and ticker should be subscribed by class for futures and options by class or by product for underlying. Also note that CBOE currently does not permit book depth subscriptions for options.* CBOE's recommendation currently is to use a callback object per product class and message type. This would allow the CAS to invoke the callbacks on distinct threads, without the overhead of managing an excessive number of queues or the delay of one subject's messages behind another subject's messages.

- "Special Consideration for CFE Orders" – added the text below to this section

  The only CFE user role that can enter orders of origin "M" is the market-maker role. The broker-dealer role is not allowed to enter "M" orders into the CFE_MAIN session.

- Created a new section for Preferred DPM

  Firms that wish to give one DPM priority in participating in a trade use optional data. The firm would send "P:firm;" and can coexist with other data that may be present in this field. "firm" is the CBOE firm acronym that will be supplied by CBOE. Please note that the message must include the colon : and semi-colon ;

- Updated the QRM section to reflect multiple user IDs sharing the same QRM values.

- Added a new section for Overlay Mode that includes the example below.

  Let's say there are 10 products for a given class.

1) The firm gets an update from CBOE for one of those products.
2) Before the firm finishes processing that one update, the other nine tick.
3) The next call the firm gets from us will be the remaining nine ticks on the remaining nine products all at once.

So, one of the features of overlay is to "group" up Current Market into chunks. The chunks will be small if the firm processes them fast and potentially larger as the firm's processing becomes slower.

The second feature of overlay is to only give the very latest Current Market.

1) The firm gets an update from CBOE for product X
2) While the firm is processing that update, Product X ticks eight more times.
3) CBOE would then send only the eighth of those updates for Product X to the firm. CBOE would not send the first seven messages in step #2 above.

The only messages that will ever be dropped in overlay mode are "old" current market messages that are no longer "current" for one particular product. Those old current markets will be overlaid on a product by product basis. Also, there will never be queuing in overlay because of this, thus it is impossible to disconnect because of large queues if the firm is using overlay mode.

If the firm wants to limit market data, it can limit the market data to a maximum of some number of calls per second. So once the firm gets a market data message from the MDCAS, it sleeps for X milliseconds, then returns the call. This has the effect of forcing overlay on the MDCAS side if the option ticks more then once per some number of seconds. This allows the firm's application to avoid "old" current market messages while at the same time "grouping" the current market into larger messages. It also has the effect of setting a "worst case scenario" for application throughput which steadies out the maximum CPU the application will try to use. If the firm needs larger chunks, this is the only way they can ensure it will be achievable. The firm may want to implement a strategy similar to this.

API-03

- Added new constant descriptions based on this IDL release
- Added new clearing descriptions to the cmiOrder::OrderEntryStruct
- Changed the description of AuctionTypes.AUCTION_REGULAR_SINGLE to read:

  AuctionTypes.AUCTION_REGULAR_SINGLE Initial Value = 3
  Used for auctioning normal orders (single-leg orders that are not internalized) NOT currently supported.

API-04

- Added new constant descriptions based on this IDL release
- Added new clearing descriptions to the cmiOrder::OrderEntryStruct
- Changed the description of AuctionTypes.AUCTION_REGULAR_SINGLE to read:

AuctionTypes.AUCTION_REGULAR_SINGLE Initial Value = 3
Used for auctioning normal orders (single-leg orders that are not internalized) NOT currently supported.

API-05

- No changes

API-06

- No changes

API-07

- Added a section to describe the new “Cert” environment.

  CBOE has a new environment available for firms to use for CMi development and testing. The new environment is called the "Certification" or "Cert" environment. This environment will be the exact same CMi version as production, which will make it easy for your firm to test production issues if they arise. In addition, if your firm is coding to CBOE's APIs to go live in production, your firm may wish to use this new Cert environment to ensure API version compatibility when going live. This environment has all sessions (W_MAIN, CFE_MAIN, ONE_MAIN, UNDERLYING, etc.), similar to the current test environment.

- Modified Phase 5 Production test to read:
  CBOE Futures Exchange (CFE)
  Test your application in the production environment with the CBOE Production Help Desk prior to going live. No test plans are needed.
  CH
  BT

  OneChicago
  Test your application in the production environment with the CBOE Production Help Desk prior to going live. No test plans are needed.
  AK
  HOB

CAS-01

- No changes

CAS-02

- No changes

Test Plan Changes

CMi Phase 2 Test Plans

- Added steps CU.200 - CU.260 in the CAS User Session section to test obtaining the reference to the following interfaces: getMarketQueryV2, getOrderQueryV2, getQuoteV2, getMarketQueryV3, getOrderEntryV3, getOrderQueryV3, getQuoteV3.
- Removed step: At the completion of the CMi Phase 2 Test Plan, API Client Relations Staff will check to verify that all messages were delivered properly, and that there are no messages waiting in queue to be delivered to the user.