CBOE®

CHICAGO BOARD OPTIONS EXCHANGE

CBOE Application Programming Interface

CBOE API Version 2.52 - Production Update Release Notes

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

CBOE PROPRIETARY INFORMATION

28 April 2004

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

CBOE API Version 2.52 - Production Update Release Notes

CBOE Proprietary Information

Overview

Current Production System

This document highlights changes to the current production CMi API, Version 2.5. Firms wishing to connect to the current CMi API production system should use the Version 2.52 documents. IDL changes in this release pertain only to constants and error code updates in the current production code base, CMi V2.5; therefore, a new simulator will not be released. For Phase 1 of CMi certification in the current production system, the simulator in Version 2.5 SDK should be submitted. A recompile of the current production version, CMi V2.5, is necessary in order to capture the new constants and error codes detailed in the CMi V2.52 documentation.

Next Version with new Functionality

CBOE is currently working on new versions of the CMi and FIX APIs to support SToC (Stock Trading on CBOEdirect) and significant functional upgrades for the Hybrid Trading environment, known as the "API Enhancement Project."

The Version 2.62 SDK, currently on the API web site, represents our first release pertaining to SToC. During May 2004, new versions of the APIs will be released that contain most remaining API changes for SToC and will include changes for the API Enhancemnt Project.

Below are descriptions of the current CMi API releases that are available for download on the API website at http://systems.cboe.com/webAPI/

- V2.5 – Current production version, including simulator.
- V2.52 – Current production upgrade of constants and error codes – *this release*
- V2.62 – Stock simulator

IDL and documentation changes for CMi V2.52 are detailed in the sections below. Your feedback or questions regarding this document should be sent to api@cboe.com.

IDL Interfaces

New and modified IDL is reflected in **bold** face.

CMi Constants

```
module cmiConstants
{
        interface ProductStates
          {
             const cmiSession::ProductState CLOSED = 1;
             const cmiSession::ProductState PRE_OPEN = 2;
             const cmiSession::ProductState OPENING_ROTATION = 3;
             const cmiSession::ProductState OPEN = 4;
```

```
        const cmiSession::ProductState HALTED = 5;

        const cmiSession::ProductState FAST_MARKET = 6;

        const cmiSession::ProductState NO_SESSION = 7;

        const cmiSession::ProductState ON_HOLD = 8;

        const cmiSession::ProductState ENDING_HOLD = 9;

        const cmiSession::ProductState SUSPENDED = 10;
    };
```

This code is used in a product state change callback event to indicate that a failover situation has occured affecting the product for which the callback is subscribed and that the product is suspended from trading.

```
interface ClassStates
    {

        const cmiSession::ClassState NOT_IMPLEMENTED = 1;

        const cmiSession::ClassState PRE_OPEN = 2;

        const cmiSession::ClassState OPENING_ROTATION = 3;

        const cmiSession::ClassState OPEN = 4;

        const cmiSession::ClassState HALTED = 5;

        const cmiSession::ClassState FAST_MARKET = 6;

        const cmiSession::ClassState NO_SESSION = 7;

        const cmiSession::ClassState ON_HOLD = 8;

        const cmiSession::ClassState ENDING_HOLD = 9;

        const cmiSession::ClassState CLOSED = 10;

        const cmiSession::ClassState SUSPENDED = 11;
    };
```

This code is used in a class state change callback event to indicate that a failover situation has occurred affecting the class for which the callback is subscribed and that the class is suspended from trading.

```
interface ActivityReasons
    {

        const cmiUtil::ActivityReason NOTHING_DONE = 1;

        const cmiUtil::ActivityReason USER = 2;

        const cmiUtil::ActivityReason SYSTEM = 3;

        const cmiUtil::ActivityReason LOST_CONNECTION = 4;

        const cmiUtil::ActivityReason INSUFFICIENT_QUANTITY = 5;

        const cmiUtil::ActivityReason SPECIAL_ADJUSTMENT = 6;
```

4

```
const cmiUtil::ActivityReason QRM_REMOVED = 7;
const cmiUtil::ActivityReason INSUFFICIENT_QUANTITY_BUY_SIDE  = 8;
const cmiUtil::ActivityReason INSUFFICIENT_QUANTITY_SELL_SIDE  = 9;


// acceptServerFailure event would have following reason
const cmiUtil::ActivityReason FAILOVER= 11;
```

> *This code may be used to indicate that a quote has been canceled due to CBOEdirect server failover (ActivityReasons.FAILOVER)*

```
// The following are used for Linkage
const cmiUtil::ActivityReason BROKER_OPTION = 100;
const cmiUtil::ActivityReason CANCEL_PENDING = 101;
const cmiUtil::ActivityReason CROWD_TRADE = 102;
const cmiUtil::ActivityReason DUPLICATE_ORDER = 103;
const cmiUtil::ActivityReason EXCHANGE_CLOSED = 104;
const cmiUtil::ActivityReason GATE_VIOLATION = 105;
const cmiUtil::ActivityReason INVALID_ACCOUNT = 106;
const cmiUtil::ActivityReason INVALID_AUTOEX_VALUE = 107;
const cmiUtil::ActivityReason INVALID_CMTA = 108;
const cmiUtil::ActivityReason INVALID_FIRM = 109;
const cmiUtil::ActivityReason INVALID_ORIGIN_TYPE = 110;
const cmiUtil::ActivityReason INVALID_POSITION_EFFECT = 111;
const cmiUtil::ActivityReason INVALID_PRICE = 112;
const cmiUtil::ActivityReason INVALID_PRODUCT = 113;
const cmiUtil::ActivityReason INVALID_PRODUCT_TYPE = 114;
const cmiUtil::ActivityReason INVALID_QUANTITY = 115;
const cmiUtil::ActivityReason INVALID_SIDE = 116;
const cmiUtil::ActivityReason INVALID_SUBACCOUNT = 117;
const cmiUtil::ActivityReason INVALID_TIME_IN_FORCE = 118;
const cmiUtil::ActivityReason INVALID_USER = 119;
const cmiUtil::ActivityReason LATE_PRINT = 120;
const cmiUtil::ActivityReason NOT_FIRM = 121;
const cmiUtil::ActivityReason MISSING_EXEC_INFO = 122;
const cmiUtil::ActivityReason NO_MATCHING_ORDER = 123;
const cmiUtil::ActivityReason NON_BLOCK_TRADE = 124;
const cmiUtil::ActivityReason NOT_NBBO = 125;
```

```
const cmiUtil::ActivityReason COMM_DELAYS = 126;
const cmiUtil::ActivityReason ORIGINAL_ORDER_REJECTED = 127;
const cmiUtil::ActivityReason OTHER = 128;
const cmiUtil::ActivityReason PROCESSING_PROBLEMS = 129;
const cmiUtil::ActivityReason PRODUCT_HALTED = 130;
const cmiUtil::ActivityReason PRODUCT_IN_ROTATION = 131;
const cmiUtil::ActivityReason STALE_EXECUTION = 132;
const cmiUtil::ActivityReason STALE_ORDER = 133;
const cmiUtil::ActivityReason ORDER_TOO_LATE = 134;
const cmiUtil::ActivityReason TRADE_BUSTED = 135;
const cmiUtil::ActivityReason TRADE_REJECTED = 136;
const cmiUtil::ActivityReason REJECTED_LINKAGE_TRADE  = 170;
const cmiUtil::ActivityReason SATISFACTION_ORD_REJ_OTHER  = 171;
```

const cmiUtil::ActivityReason PRODUCT_SUSPENDED = 172;

This code may be used to indicate that a quote has been canceled due to Product state transition to ProductStates.SUSPENDED (ActivityReasons.PRODUCT_SUSPENDED).

```
// Currently used for TPF linkage; in future may be used for CBOEdirect
const cmiUtil::ActivityReason UNKNOWN_ORDER = 137;
const cmiUtil::ActivityReason INVALD_EXCHANGE = 138;
const cmiUtil::ActivityReason TRANSACTION_FAILED = 139;
const cmiUtil::ActivityReason NOT_ACCEPTED = 140;
```

```
// Used for linkage when cancel reason is not provided (could be user cancel or cancel remaining)
const cmiUtil::ActivityReason AWAY_EXCHANGE_CANCEL = 199;
```

```
// Linkage Business Message Reject codes
const cmiUtil::ActivityReason LINKAGE_CONDITIONAL_FIELD_MISSING = 900;
const cmiUtil::ActivityReason LINKAGE_EXCHANGE_UNAVAILABLE = 901;
const cmiUtil::ActivityReason LINKAGE_INVALID_MESSAGE = 902;
const cmiUtil::ActivityReason LINKAGE_INVALID_DESTINATION = 903;
const cmiUtil::ActivityReason LINKAGE_INVALID_PRODUCT = 904;
const cmiUtil::ActivityReason LINKAGE_SESSION_REJECT = 905;
```

```
                   };
```

CMi Error Codes

interface CommunicationFailureCodes

{

    const exceptions::ErrorCode TRANSPORT_FAILURE = 2500;

    const exceptions::ErrorCode ROUTING_SESSION_UNAVAILABLE = 2510;

    const exceptions::ErrorCode LOST_CONNECTION = 2520;

    const exceptions::ErrorCode SERVER_NOT_AVAILABLE = 2530;

    };

The Communication Failure Code, SERVER_NOT_AVAILABLE, may arise in response to the following requests when the CBOEdirect server responsible for processing the request is not available.

OrderQuery
- *getOrderById*
- *getOrdersForClass*
- *getOrdersForProduct*
- *getOrdersForSession*
- *getOrdersForType*
- *getPendingAdjustmentOrdersByClass*
- *getPendingAdjustmentOrdersByProduct*
- *queryOrderHistory*

OrderQueryV2
- *getOrderById*
- *getOrdersForClass*
- *getOrdersForProduct*
- *getOrdersForSession*
- *getOrdersForType*
- *getPendingAdjustmentOrdersByClass*
- *getPendingAdjustmentOrdersByProduct*
- *queryOrderHistory*

OrderEntry
- *acceptOrderCancelRequest*

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 SERVER_NOT_AVAILABLE = 4080;

};

This code may arise in response to the following requests when the CBOEdirect server responsible for processing the request is not available:

OrderQuery
- *getOrderById*
- *queryOrderHistory*

OrderQueryV2
- *getOrderById*
- *queryOrderHistory*

OrderEntry
- *acceptOrder*
- *acceptOrderByProductName*
- *acceptCrossingOrder*
- *acceptCancelRequestRequest*
- *acceptCancelReplaceRequest*
- *acceptOrderUpdateRequest*
- *acceptStrategyOrder*
- *acceptStrategyOrderCancelReplaceRequest*
- *acceptStrategyOrderUpdateRequest*

Quote
- *acceptQuote*
- *cancelQuote*
- *acceptQuotesForClass*
- *cancelQuotesByClass*

QuoteV2
- *acceptQuote*
- *cancelQuote*
- *acceptQuotesForClass*
- *cancelQuotesByClass*
- *acceptQuotesForClassV2*

8

CAS Simulator Changes

- No changes

Document Changes

API-01

- No changes

API-02

- Users may create multiple sessions with the same login. In order to create multiple sessions for the same user, all logins must be on the same CAS. A user cannot create multiple user sessions using different CASes. This is regardless of the logon session mode (primary, secondary) or GMD status.

- CBOE strongly encourages users to enter quotes from a user that is logged on with logon session mode Primary.

- If a user logs on with only one Secondary logon session type and there are no Primary users when this Secondary user logs out, CBOE does cancel the user's quotes.

- Added CFE Quote Throttle limits.

API-03

- Order and Quote status and status update reasons were updated.

- Order Entry Struct - Position Effect: ... However, this value must be set to a value of "N" (neither) for all orders of origin "M" or "I" in the W_MAIN session.

- Enhanced the description for Branch and Branch Sequence Number.

API-04

- Order and Quote status and status update reasons were updated.

- Order Entry Struct - Position Effect: ... However, this value must be set to a value of "N" (neither) for all orders of origin "M" or "I" in the W_MAIN session.

- Enhanced the description for Branch and Branch Sequence Number.

API-05

- No changes

API-06

- No changes

API-07

- No changes

CAS-01

- No changes

CAS-02

- No changes

Test Plans

2, CMi Phase 2 Test Plan

- No changes

3a, Security Definition Test Plan

- Added a step to test the error condition that occurs when a user attempts to download products from a session for which that user is not an authorized participant.

- Added a new sub-section "Important W_MAIN Product State Considerations" within the "Trading Sessions and Product Downloads" section.

3b, Market Data Test Plan

- Added new testing steps for EOP type PRICE_NOT_IN_QUOTE_RANGE.

- CBOE publishes the underlying index market data for index classes (such as OEX, S&P500, Dow Jones, NDX etc.). CBOE does not publish the market data for the futures contract that mirrors the index. For example, CBOE does publish the underlying market data for the S&P 500 *index*, but CBOE does not publish the market data for the S&P 500 *futures* contract.

- Added minimum tick information for CFE.

3c, Quote Test Plan - Hybrid-ONE-CFE (includes 3j, Hybrid Sections and 3L, CFE Supplemental Tests)

- Test plan now requests that the user identifies the contra-party acronym and contra-party clearing firm in fill reports.

- Electronic quote busts are not supported in W_MAIN (Hybrid or non-Hybrid) for market-makers or DPMs. CBOE does, however, manually bust trades in W_MAIN. Therefore, the firm's application should somehow be able to reflect the effects of trade busts.

- CBOE does not consider quote or mass quote cancels when calculating quote thresholds.

- Added CFE quote threshold amounts (same as ONE_MAIN).

- Hybrid Mass Quote messages may contain up to 10 quotes.

10

- For each Hybrid user, the Quote limits are a total of 100 Quote Messages (either single quotes or mass quote) per <u>one (1)</u> second period and 1000 total quotes per <u>five (5)</u> second period. In other words, if a user submits 100 (Mass) quote messages in one second, each containing 10 quotes, then that user will also have reached the 1000 quotes per 5 second limit and be blocked from submitting any additonal quotes until a total of 5 seconds have passed.

- CFE and OneChicago Mass Quote messages may contain up to 4 quotes.

- For each CFE and OneChicago user, the Quote limits are a total 50 Quote Messages (either single quote or mass quote) per <u>one (1)</u> second period, and 1000 total quotes per user per <u>five (5)</u> second period.

- Reintroduced the Hybrid Quote Trigger section and added two error condition steps to send a quote cancel and update request during a Hybrid Quote Trigger.

- Added a step to test a shorter Quote Lock time period of 4 seconds after the quote is entered.

- QRM is now required for all Market-Makers and DPMs. If the firm decides to not code to the QRM in CMi or FIX, the CBOE Administrator will create QRM settings in the CBOEdirect Administrator terminal for the user to test.

3d, RTH Order Test Plan

- No changes

3e, RTH-ONE-CFE Order Test Plan (includes 3k, CFE Supplemental Tests)

- In the production environment, "DAY" orders persist for the day that the order is entered and the following day. In the test environment, "DAY" orders will persist throughout the entire week until the system is recycled (usually every Wednesday).

- In the production environment, "GTC" orders will persist through the end of the day following the completion of the order (e.g. filled or canceled). In the test environment, "GTC" orders will persist throughout the entire week until the system is recycled (usually every Wednesday).

- Test plan now requests that the user identifies the contra-party acronym and contra-party clearing firm in fill reports.

- Enhanced the bust section and added a step to cancel a reinstated order.

- Added all order origins for CBOE, CFE, and ONE.

- CBOE does cancel the user's "I" orders when the user logs out. (CBOE formerly did not cancel them).

- Note: when a firm enters an order into the W_MAIN session (Hybrid or non-Hybrid) and the order rejects, it will reject to the firm's booth on the trading floor. The CMi or FIX 4.2 application **will not** receive an error message.

- Enhanced the description of the Quote Lock timer for additional traders that join a trade after the initial Quote Lock takes place.

- Added a testing step to enter the Originator field. Originator is required for all Broker-Dealer users entering orders of origin "M" or "I" in all sessions. The value of the originator field

CONFIDENTIAL

must be an actual, valid MM badge in membership and the CBOEdirect system. The Originator exchange must be the exchange that the user is a member of (CBOE, CFE, ONE, etc.)

- Electronic order busts or order bust reinstates are not supported in W_MAIN (Hybrid or non-Hybrid) for all order origins. CBOE does, however, manually bust trades in W_MAIN. Therefore, the firm's application should somehow be able to reflect the effects of trade busts

- Changed the CFE minimum tick from 0.01 to 0.10.

- Added a step at the beginning of the test plan to do a straight cancel on an order that has not been modified yet.

- Added an error condition step to send an ICM order cancel request during a Hybrid Quote Trigger.

- Added a step to test a shorter Quote Lock time period of 4 seconds after an ICM order is entered.

3f, Clearing Firm, Duplicate Message Test Plan

- The Firm and Firm Display roles are currently not implemented. Nevertheless, Market-Makers, DPMs, and Broker-Dealers must complete the "Market-Maker and Broker-Dealer Possible Resends" section of this test plan.

3g, Strategy Quote Test Plan - ONE-CFE

- CBOE does not support quote strategy busts electronically in any session. In W_MAIN, however, CBOE can manually bust a strategy trade. Therefore, you should allow your traders to manually change their positions on the fly. In purely electronic sessions that do not interact with the trading floor, CBOE does not support strategy busts in any fashion, manually or electronically.

- Changed the CFE minimum tick from 0.01 to 0.10.

- Diagonal strategies have to have a ratio of 1:1.

- The sides of the option and stock legs for buy_writes do not have to be opposite sides

- Pricing for buy_writes is calculated as if the stock leg were a round lot, whether it is an odd-lot or not.

3h, RTH Strategy Order Test Plan

- No changes

3i, RTH-ONE-CFE Strategy Order Test Plan

- Added a section "Important Considerations for W_MAIN Strategy Orders".

- Changed the CFE minimum tick from 0.01 to 0.10.

- Diagonal strategies have to have a ratio of 1:1.

- The sides of the option and stock legs for buy_writes do not have to be opposite sides.

- Pricing for buy_writes is calculated as if the stock leg were a round-lot, whether it is an odd-lot or not.

- Added a section for FIX cancel replaces for enhanced certification functionality.

3m, Stock Trading On CBOEdirect (STOC) Order Test Plan

- In the production environment, "DAY" orders will persist for the day that the order is entered and the following day. In the test environment, "DAY" orders will persist throughout the entire week until the system is recycled (usually every Wednesday).

- In the production environment, "GTC" orders will persist through the end of the day following the completion of the order (e.g. filled or canceled). In the test environment, "GTC" orders will persist throughout the entire week until the system is recycled (usually every Wednesday).

- User identifies the contra-party acronym and contra-party clearing firm.

3n, Stock Trading On CBOEdirect (STOC) Quote Test Plan

- User identifies the contra-party acronym and contra-party clearing firm.

3o, Stock Trading On CBOEdirect (STOC) DPM Administrative Test Plan

- No changes

Phase 4 Test Plan

- No changes