CBOE Application Programming Interface

CBOE API Version 5.1 - Release Notes

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

CBOE PROPRIETARY INFORMATION

23 July 2008

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 upcoming changes in the new release of the CMi API, Version 5.1. This release supports new IDL constants and documentation changes. The sections below detail the changes in this release. Your feedback or questions regarding this document should be sent to api@cboe.com.

CMi V5.1 Highlights

The CBOE will begin a rollout of a change to the internal routing of orders. This software load will begin after July expiration. The external behavioral changes are:

(1) Orders that open positions for restricted series will no longer be accepted and routed to a booth for handling. They will be rejected immediately. The CMi cancel reason code is "const exceptions::ErrorCode ORDER_REJECTED_ON_RSS = 1909".

(2) When a series becomes restricted, any resting orders that would open a position will be canceled by the system. The cancel reports will be delivered to the originating user on login with a cancel report Activity Reason of " const cmiUtil::ActivityReason CANCEL_ON_RSS = 24.

(3) Currently cancel requests for partial quantity of a complex order cause the order to be canceled in its entirety. This behavior will remain if the order is resting on PAR or a booth. However, if the order is booked, the partial cancel request will be applied and any remaining quantity will remain in the market.

IDL Interfaces

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

```
module cmiErrorCodes
{
    interface DataValidationCodes {
        const exceptions::ErrorCode DUPLICATE_ID = 1000;
        const exceptions::ErrorCode INVALID_TIME = 1020;
        const exceptions::ErrorCode INCOMPLETE_QUOTE = 1030;
        const exceptions::ErrorCode INVALID_QUANTITY = 1040;
        const exceptions::ErrorCode INVALID_STRATEGY = 1060;
        const exceptions::ErrorCode INVALID_SPREAD = 1070;
        const exceptions::ErrorCode INVALID_USER = 1080;
        const exceptions::ErrorCode INVALID_PRODUCT = 1090;
        const exceptions::ErrorCode INVALID_PRODUCT_CLASS = 1091;
        const exceptions::ErrorCode INVALID_REPORTING_CLASS = 1092;
        const exceptions::ErrorCode INVALID_PRODUCT_DESCRIPTION = 1093;
        const exceptions::ErrorCode INVALID_OPRA_MONTH_CODE = 1094;
```

```
const exceptions::ErrorCode INVALID_PRICE_ADJUSTMENT = 1095;
const exceptions::ErrorCode INVALID_SESSION = 1100;
const exceptions::ErrorCode INVALID_STATE = 1110;
const exceptions::ErrorCode PREFERENCE_PATH_MISMATCH = 1120;
const exceptions::ErrorCode INVALID_ORDER_ID = 1130;
const exceptions::ErrorCode NO_WORKING_ORDER = 1135;
const exceptions::ErrorCode LISTENER_ALREADY_REGISTERED = 1140;
const exceptions::ErrorCode INVALID_SIDE = 1150;
const exceptions::ErrorCode INVALID_PRICE = 1160;
const exceptions::ErrorCode INVALID_UPDATE_ATTEMPT = 1170;
const exceptions::ErrorCode INVALID_ORIGINATOR = 1180;
const exceptions::ErrorCode INVALID_ACCOUNT = 1200;
const exceptions::ErrorCode INVALID_EXECUTING_GIVEUP_FIRM = 1210;
const exceptions::ErrorCode INVALID_CONTINGENCY_TYPE = 1220;
const exceptions::ErrorCode INVALID_TIME_IN_FORCE = 1230;
const exceptions::ErrorCode INVALID_POSITION_EFFECT = 1240;
const exceptions::ErrorCode INVALID_ORIGIN_TYPE = 1250;
const exceptions::ErrorCode INVALID_COVERAGE = 1260;
const exceptions::ErrorCode INVALID_PRODUCT_TYPE = 1270;
const exceptions::ErrorCode INVALID_ORDER_STATE = 1280;
const exceptions::ErrorCode INVALID_ORDER_SOURCE = 1290;
const exceptions::ErrorCode INVALID_BRANCH_SEQUENCE_NUMBER = 1300;
const exceptions::ErrorCode MISSING_LISTENER = 1310;
const exceptions::ErrorCode BUSINESS_DAY_NOT_STARTED = 1320;
const exceptions::ErrorCode INVALID_FIELD_LENGTH = 1330;
const exceptions::ErrorCode INVALID_STRATEGY_LEG = 1340;
const exceptions::ErrorCode DUPLICATE_STRATEGY_LEG = 1350;
const exceptions::ErrorCode INVALID_LEG_CONTINGENCY = 1360;
const exceptions::ErrorCode INVALID_CANCEL_REQUEST = 1370;
const exceptions::ErrorCode INVALID_VERSION = 1380;
const exceptions::ErrorCode INVALID_LOGIN_MODE = 1390;
const exceptions::ErrorCode GMD_LISTENER_ALREADY_REGISTERED = 1400;
const exceptions::ErrorCode INVALID_TRADE_SOURCE = 1410;
const exceptions::ErrorCode INVALID_TRADE_TYPE = 1420;
const exceptions::ErrorCode NO_REMAINING_QUANTITY = 1430;
const exceptions::ErrorCode INVALID_OPENING_REQUIREMENT = 1440;
```

```
const exceptions::ErrorCode INVALID_PROCESS_NAME = 1450;
const exceptions::ErrorCode INVALID_GROUP = 1460;
const exceptions::ErrorCode INVALID_NAME = 1461;
const exceptions::ErrorCode INVALID_THRESHOLD = 1462;
const exceptions::ErrorCode INVALID_OPERATOR = 1463;
const exceptions::ErrorCode INVALID_TRADE_REPORT_HANDLING_INSTRUCTION = 1464;
const exceptions::ErrorCode INVALID_OPERATION_TYPE = 1475;
    // GroupService related section
const exceptions::ErrorCode INVALID_GROUPELEMENT = 1474;
const exceptions::ErrorCode INVALID_GROUPELEMENT_TYPE = 1465;
const exceptions::ErrorCode INVALID_GROUPELEMENT_RELATIONSHIP = 1466;
    const exceptions::ErrorCode GROUPELEMENT_ALREADY_EXISTS = 1467;
    const exceptions::ErrorCode GROUPRELATIONSHIP_ALREADY_EXISTS = 1468;
    const exceptions::ErrorCode INVALID_USERID_REQUESTING_CANCEL = 1469;
    const exceptions::ErrorCode INVALID_WORKSTATION_ID = 1470;
    const exceptions::ErrorCode INVALID_USERID_LIST = 1471;
    const exceptions::ErrorCode ROOT_ALREADY_EXISTS = 1472;
    const exceptions::ErrorCode INVALID_GROUP_TYPE = 1473;

// 1500 series for additions made for linkage support
const exceptions::ErrorCode INVALID_EXCHANGE = 1500;
const exceptions::ErrorCode INVALID_EXTENSIONS = 1510;
const exceptions::ErrorCode INVALID_REJECT_REQUEST = 1520;

const exceptions::ErrorCode INVALID_ID = 1530;
const exceptions::ErrorCode INVALID_POLICIES = 1531;
const exceptions::ErrorCode INVALID_LIMITS = 1532;
const exceptions::ErrorCode INVALID_TYPE = 1533;
const exceptions::ErrorCode INVALID_COUNT_BOUNDARIES = 1534;
const exceptions::ErrorCode INVALID_TIME_BOUNDARIES = 1535;

// 1600 series for additions made to support internalized orders
const exceptions::ErrorCode INVALID_MATCH_TYPE = 1600;
const exceptions::ErrorCode INVALID_AUCTION_STATE = 1610;
const exceptions::ErrorCode INVALID_AUCTION_ID = 1620;
const exceptions::ErrorCode INTERNALIZATION_NOT_ALLOWED = 1630;
```

```
const exceptions::ErrorCode INVALID_AUCTION_TYPE = 1640;
const exceptions::ErrorCode INVALID_OPTIONAL_DATA = 1650;

// 1700 series for additions made to support index hybrid feature
const exceptions::ErrorCode INVALID_CONTINGENCY_BOB_IORDER = 1700;
const exceptions::ErrorCode INVALID_CONTINGENCY_VIX_SETTLEMENT = 1701;

// 1710 series for  Cross Product
const exceptions::ErrorCode UNSUPPORTED_ORIGIN_TYPE = 1711;
const exceptions::ErrorCode UNDERLYING_LEG_NOT_LISTED_INSTOCK = 1712;
const exceptions::ErrorCode INVALID_RATIO_FOR_CROSS_PROD = 1713;
const exceptions::ErrorCode INVALID_LEG_STOCK_PROD_STATE = 1714;
const exceptions::ErrorCode INVALID_CLEARING_FIRM = 1715;

//1800 series for manual price reporting
const exceptions::ErrorCode END_OF_SALE = 1800;
const exceptions::ErrorCode NOT_AN_OPENING_ONLY_TRADE = 1801;
const exceptions::ErrorCode NO_TRADE_SO_FAR = 1802;
const exceptions::ErrorCode NOT_AN_ONLY_TRADE = 1803;
const exceptions::ErrorCode ONLY_OPENING_TRADE_SO_FAR = 1804;
const exceptions::ErrorCode EITHER_LAST_SALE_OR_OPENING_TRADE = 1805;
const exceptions::ErrorCode PRICE_NOT_EQUAL_TO_LAST_SALE = 1806;
const exceptions::ErrorCode CANCELED_VOL_NOT_CUMMULATIVE_VOL = 1807;
const exceptions::ErrorCode PRICE_NOT_EQUAL_TO_OPENING_PRICE = 1808;
const exceptions::ErrorCode PRICE_GREATER_THAN_HIGH = 1809;
const exceptions::ErrorCode PRICE_LESS_THAN_LOW = 1810;
const exceptions::ErrorCode VOLUME_GREATER_THAN_CUMUMATIVE_VOL = 1811;

// 1900 series for OHS Release One
const exceptions::ErrorCode INVALID_CONTRA_BROKER = 1900;
const exceptions::ErrorCode INVALID_CONTRA_FIRM = 1901;
const exceptions::ErrorCode INVALID_EXECUTING_BROKER = 1903;
const exceptions::ErrorCode INVALID_EXECUTING_BROKER_FIRM = 1904;
const exceptions::ErrorCode INVALID_CABINET_ON_CXLRE = 1905;
const exceptions::ErrorCode INVALID_NO_CURRENT_MARKET = 1906;
const exceptions::ErrorCode INVALID_UPDATE_PRICE_REPORT = 1907;
```

```
        const exceptions::ErrorCode INVALID_NO_MATCHING_MDH = 1908;
        const exceptions::ErrorCode ORDER_REJECTED_ON_RSS = 1909;
  };

module cmiConstants
{
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;
    const cmiUtil::ActivityReason QRM_REMOVED = 7;
    const cmiUtil::ActivityReason INSUFFICIENT_QUANTITY_BUY_SIDE  = 8;
    const cmiUtil::ActivityReason INSUFFICIENT_QUANTITY_SELL_SIDE  = 9;
    const cmiUtil::ActivityReason QUOTE_UPDATE_CONTROL =10;
    // acceptServerFailure event would have following reason
    const cmiUtil::ActivityReason FAILOVER= 11;
    const cmiUtil::ActivityReason QUOTE_IN_TRIGGER =12;
    const cmiUtil::ActivityReason INVALID_SESSION_ID =13;
    const cmiUtil::ActivityReason SAL_IN_PROGRESS = 14;
    const cmiUtil::ActivityReason CROSS_IN_PROGRESS = 15;
    const cmiUtil::ActivityReason INVALID_NBBO = 16;
    const cmiUtil::ActivityReason NOT_WITHIN_NBBO = 17;
    const cmiUtil::ActivityReason TRADE_THROUGH_CBOE = 18;
    const cmiUtil::ActivityReason INSUFFICIENT_CUSTOMER_ORDER_QUANTITY = 19;
    const cmiUtil::ActivityReason INSUFFICIENT_CROSS_ORDER_SIZE = 20;
    const cmiUtil::ActivityReason INSUFFICIENT_CROSS_ORDER_DOLLAR_AMOUNT = 21;
    const cmiUtil::ActivityReason SELL_SHORT_RULE_VIOLATION = 22;
    // acceptUserActivityTimeout (UIM) event would have the following reason:
    const cmiUtil::ActivityReason NO_USER_ACTIVITY = 23;
    const cmiUtil::ActivityReason CANCEL_ON_RSS = 24;

    // 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 ORDER_TIMEOUT = 141;
    const cmiUtil::ActivityReason REJECTED_LINKAGE_TRADE  = 170;
    const cmiUtil::ActivityReason SATISFACTION_ORD_REJ_OTHER  = 171;
    const cmiUtil::ActivityReason PRODUCT_SUSPENDED = 172;

    // 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;

   // Force Cancel Orders due to fallback
    const cmiUtil::ActivityReason CANCEL_ON_FALLBACK = 800;

    // 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;
  };
```

Document Changes

API-01

- No changes

API-02

- Modified the quote rate limits:

Hybrid (in session W_MAIN)

- 400 Quotes (products) per Mass Quote message
- 133 Mass Quote or Quote message calls per user per 1 second period
- 4000 total quotes (products) per user per 3 second period

CFE_MAIN (CBOE Futures Exchange) and **ONE_MAIN** (OneChicago)

- 12 Quotes (products) per Mass Quote message
- 250 Mass Quote or Quote message calls per user per five (5) second period
- 1000 total quotes (products) per user per five (5) second period

- Updated the "Cancel Functionality" section with the following:

  CancelAllQuotes is a best effort asynchronous request that instructs all CBOEDirect trade servers to cancel all resting quotes for a user. The return of this call does not indicate that all of the users resting quotes have been removed but that the process has been initiated on all trade servers.

  Cancel quotes for a class is a synchronous call that directly removes the quotes for a user for a given class. The return of this call indicates that all quotes for the user have been removed. It is possible that there are in flight execution reports. It is also possible a quote is involved in a trade at the moment of the cancel request. If so, any remaining quantity for that series will be canceled at the end of the trading process.

- Added the following to the Current Rate Limit section:

  The *UserTradingParameters* interface was extended to include a new method, *getUserRateSettings* in the CMi V5 module. This method takes a session name and returns the current values of various rate limits. Each rate limit is returned as a generic struct containing a key and a value, both of which are string types. Currently, this method returns the quote rate limit, quote call limit, order rate limit, order call limit and book depth call limit. This feature will be available in April 2008.

```
interface UserTradingParameters :
cmi::UserTradingParameters
{
cmiUtil::KeyValueStructSequence
getUserRateSettings(in string sessionName)
raises(
exceptions::SystemException,
exceptions::CommunicationException,
exceptions::AuthorizationException,
exceptions::DataValidationException
);
};
```

And example of the return data:
User settings for W_MAIN

quoteConstraints.rateMonitorWindow=133	Calls per second.
quoteConstraints.rateMonitorInterval=1000	Monitor window size in milliseconds for call rate.
quoteConstraints.quoteRateMonitorInterval=3000	Monitor window size in milliseconds for quote rate.

quoteConstraints.quoteRateMonitorWindow=4000	Quotes per monitor interval
quoteConstraints.rateMonitorQuoteSequenceSize=400	Quote block size
orderConstraints.rateMonitorWindow=100	Orders per monitor interval
orderConstraints.rateMonitorInterval=1000	Monitor window size in milliseconds for Order rate.
marketDataConstraints.rateMonitorWindow=1	
marketDataConstraints.rateMonitorInterval=1000	

- Updated the document to include external behavior changes based on this release.
    - Orders that open positions for restricted series will no longer be accepted and routed to a booth for handling. They will be rejected immediately. The CMi cancel reason code is "const exceptions::ErrorCode ORDER_REJECTED_ON_RSS = 1909".
    - When a series becomes restricted, any resting orders that would open a position will be canceled by the system. The cancel reports will be delivered to the originating user on login with a cancel report Activity Reason of " const cmiUtil::ActivityReason CANCEL_ON_RSS = 24.
    - Currently cancel requests for partial quantity of a complex order cause the order to be canceled in its entirety. This behavior will remain if the order is resting on PAR or a booth. However, if the order is booked, the partial cancel request will be applied and any remaining quantity will remain in the market.

API-03

- Added values for the new constants based on this release

API-04

- Added values for the new constants based on this release

API-05

- No changes

API-06

- No changes

API-07

- No changes

API-08

- No changes

CAS-01

- No changes

CAS-02

- No changes

Simulator

- No changes

Test Plan Changes

- No changes