CBOE Application Programming Interface

CBOE API Version 5.2 - Release Notes

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

CBOE PROPRIETARY INFORMATION

03 October 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 5.2. 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 V5.2 Highlights

The highlights discussed below will be available in the October 2008 CBOEdirect software release.

COB and Auction Enhancements

The extensions field in the AuctionStruct, used by CBOE to solicit participation in an auction, will be enhanced to provide additional information about the auction.

```
module cmiOrder
    struct AuctionStruct
      {
        cmiSession::TradingSessionName sessionName;
        cmiProduct::ClassKey classKey;
        cmiProduct::ProductType productType;
        cmiProduct::ProductKey productKey;
        cmiUtil::CboeIdStruct auctionId;
        cmiOrder::AuctionType auctionType;
        cmiOrder::AuctionState auctionState;
        cmiUtil::Side side;
        long auctionQuantity;
        cmiUtil::PriceStruct startingPrice;
        cmiOrder::ContingencyType auctionedOrderContingencyType;
        cmiUtil::TimeStruct entryTime;
        string extensions;
      };
```

The extensions field may not always be present, and may contain multiple values concerning the auction:

- COA messages may contain NBBO bid and NBBO ask in the auction message.
- Auction messages may be modified to provide user/firm information of the auctioned order.

- User will be able to choose, via firm routing parameter, which of the following information is displayed in the auction message.
    - Include Executing Firm on the auction message: Y/N
    - Include Correspondent Firm on the auction message: Y/N
    - Include CMTA Firm on the auction message: Y/N

Order Contingency Types

CBSX flash enhancements will be available through new order contingency types.

(1) const cmiOrder::ContingencyType NBBO_FLASH_THEN_CANCEL = 25

Orders marked NBBO_FLASH_THEN_CANCEL will be flashed through the flash process if they cannot be executed immediately at the NBBO. Any part of the order that is not filled will be cancelled after the flash.

(2) const cmiOrder::ContingencyType DO_NOT_ROUTE = 26

Orders marked DO_NOT_ROUTE will trade immediately or book as long as they do not violate the NBBO. If a DO_NOT_ROUTE order cannot be booked or traded without violating the NBBO, it will be cancelled. This contingency type will be available for any origin type, including "I" orders.

(3) const cmiOrder::NBBO_FLASH_RESPONSE = 27

When a marketable order is received that cannot trade against the CBSX quote due to NBBO restrictions, the order will be flashed at the NBBO. When the auction message appears, users responding to the flash will respond with the new flash response contingency.

New Stock Exchange

CBOEdirect will support a new stock exchange, the "Better Alternative Trading System (BATS)." BATS has been added to the cmiConstants.idl as, const cmiUser::Exchange BATS = "BATS"

Transaction Sequence Number Change for Strategy Fill/Cancel Reports

A recent change was made to the API that impacts the Transaction Sequence Number field, transSequenceNumber, in strategy fill reports and strategy order cancel reports. With this change, the transSequenceNumber field is duplicated in fill/cancel reports of the strategy itself and fill/cancel reports of each leg. Previously, the transSequenceNumber was different in the strategy fill/cancel report and each leg fill/cancel report.

If the API user is using the transSequenceNumber only to filter out duplicate fill reports, each leg fill report could be duplicated. CBOE suggests the API user utilize a combination of orderID, tradeID and transSequenceNumber to uniquely identify each fill report.

IDL Interfaces

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

```
module cmiConstants
{
interface ExchangeStrings
   {
   const cmiUser::Exchange AMEX    = "AMEX";   //American Stock Exchange
   const cmiUser::Exchange BSE     = "BSE";    //Boston Stock Exchange
   const cmiUser::Exchange CBOE    = "CBOE";   //Chicago Board Options Exchange
   const cmiUser::Exchange CBOT    = "CBOT";   //Chicago Board of Trade
   const cmiUser::Exchange CHX     = "CHX";    //Chicago Stock Exchange
   const cmiUser::Exchange CME     = "CME";    //Chicago Mercantile Exchange
   const cmiUser::Exchange CSE     = "CSE";    //Cincinnati Stock Exchange
   const cmiUser::Exchange ISE     = "ISE";    //International Stock Exchange
   const cmiUser::Exchange LIFFE   = "LIFFE";  //International Financial Futures and Options Exchange
   const cmiUser::Exchange NASD    = "NASD";   //National Association of Securities Dealers
   const cmiUser::Exchange NYME    = "NYME";   //New York Mercantile Exchange
   const cmiUser::Exchange NYSE    = "NYSE";   //New York Stock Exchange
   const cmiUser::Exchange ONE     = "ONE";    //OneChicago Exchange
   const cmiUser::Exchange PHLX    = "PHLX";   //Philadelphia Stock Exchange
   const cmiUser::Exchange PSE     = "PSE";    //Pacific Stock Exchange
   const cmiUser::Exchange NQLX    = "NQLX";   //Nasdaq Liffe Markets
   const cmiUser::Exchange BOX     = "BOX";    // Boston Options Exchange
   const cmiUser::Exchange CFE     = "CFE";    //CBOE Futures Exchange
   const cmiUser::Exchange NSX     = "NSX";    //National Stock Exchange
   const cmiUser::Exchange NASDAQ  = "NASDQ";  //National Association of Securities Dealers
Automated Quotation
  const cmiUser::Exchange BATS    = "BATS";   //Better Alternative Trading System

interface ContingencyTypes
   {
      const cmiOrder::ContingencyType NONE = 1; // no contingency
      const cmiOrder::ContingencyType AON = 2; // All or None
      const cmiOrder::ContingencyType FOK = 3; // Fill or Kill
      const cmiOrder::ContingencyType IOC = 4; // Immediate or Cancel
```

```
const cmiOrder::ContingencyType OPG = 5; // Opening only
const cmiOrder::ContingencyType MIN = 6; // Minimum
const cmiOrder::ContingencyType NOTHELD = 7; // Not held
const cmiOrder::ContingencyType WD = 8; // With discretion
const cmiOrder::ContingencyType MIT = 9; // Market if touched
const cmiOrder::ContingencyType STP = 10; // Stop order
const cmiOrder::ContingencyType STP_LOSS = 11; // Stop loss
const cmiOrder::ContingencyType CLOSE = 12; // On close
 const cmiOrder::ContingencyType STP_LIMIT = 13; // Stop limit
const cmiOrder::ContingencyType AUCTION_RESPONSE = 14; // Auction response order
const cmiOrder::ContingencyType INTERMARKET_SWEEP = 15; // Intermarket sweep
const cmiOrder::ContingencyType RESERVE = 16; // Reserve order
const cmiOrder::ContingencyType MIDPOINT_CROSS    = 17; // Mid Point Cross
const cmiOrder::ContingencyType CROSS             = 18; // Cross
const cmiOrder::ContingencyType TIED_CROSS        = 19; // Tied cross
const cmiOrder::ContingencyType AUTOLINK_CROSS    = 20; // Auto link cross
const cmiOrder::ContingencyType AUTOLINK_CROSS_MATCH = 21; // Auto link cross
const cmiOrder::ContingencyType CROSS_WITHIN      = 22;
const cmiOrder::ContingencyType TIED_CROSS_WITHIN = 23;
 const cmiOrder::ContingencyType STOCK_ODD_LOT_NBBO_ONLY = 24;
 const cmiOrder::ContingencyType NBBO_FLASH_THEN_CANCEL = 25;
const cmiOrder::ContingencyType DO_NOT_ROUTE = 26;
const cmiOrder::ContingencyType NBBO_FLASH_RESPONSE = 27;
};
```

Document Changes

API-01

- No changes

API-02

- Updated the Automated Improvement Mechanism (AIM) section of the document to include the COB and auction enhancements available in this release.
- The Automated Improvement Mechanism (AIM) section of the document contains references to primary order (customer) and match (firm) order. The “customer” and “firm” references have been removed from this section.

API-03

- Added values for the new constants based on this release
    - const cmiUser::Exchange BATS = "BATS";   //Better Alternative Trading System
    - const cmiOrder::ContingencyType NBBO_FLASH_THEN_CANCEL = 25;
    - const cmiOrder::ContingencyType DO_NOT_ROUTE = 26;
    - const cmiOrder::NBBO_FLASH_RESPONSE = 27

API-04

- Added values for the new constants based on this release
    - const cmiUser::Exchange BATS   = "BATS";   //Better Alternative Trading System
    - const cmiOrder::ContingencyType NBBO_FLASH_THEN_CANCEL = 25;
    - const cmiOrder::ContingencyType DO_NOT_ROUTE = 26;
    - const cmiOrder::NBBO_FLASH_RESPONSE = 27

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