



CBOE Application Programming Interface

CBOE API Version 6.1 - Release Notes

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

CBOE PROPRIETARY INFORMATION

14 August 2009

Document #[API-00]

Front Matter

Disclaimer

Copyright © 1999-2009 by the Chicago Board Options Exchange (CBOE), as an unpublished work. The information contained in this document constitutes confidential and/or trade secret information belonging to CBOE. This document is made available to CBOE members, member firms and other appropriate parties to enable them to develop software applications using the CBOE Market Interface (CMi), and its use is subject to the terms and conditions of a Software License Agreement that governs its use. This document is provided “AS IS” with all faults and without warranty of any kind, either express or implied.

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

FRONT MATTER	I
DISCLAIMER	I
SUPPORT AND QUESTIONS REGARDING THIS DOCUMENT	I
TABLE OF CONTENTS	2
OVERVIEW	3
CMI API V6.1 HIGHLIGHTS	3
CBSX NEW ORDER CONTINGENCY TYPES	3
IDL INTERFACES	3
DOCUMENT CHANGES.....	5
API-01	5
API-02	5
API-03	5
API-04	5
API-05	5
API-06	5
API-07	5
API-08	5
CAS-01	5
CAS-02	5
SIMULATOR	5
TEST PLAN CHANGES	5

Overview

This document highlights upcoming changes in the new release of the CMi API, Version 6.1. Version 6.1 supports two new CMi constants and overall 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 API V6.1 Highlights

CBSX New Order Contingency Types

The September 2009 release of CBOEdirect will support two new order contingency types: (1) ContingencyType BID_PEG_CROSS = 29 and (2) ContingencyType OFFER_PEG_CROSS = 30. Order providers will be able to specify on inbound paired orders whether the paired orders should cross at the NBBO bid price, the NBBO offer price, or 'x' better than the NBBO bid or offer price (where 'x' can be designated as a value in pennies including up to 4 decimal places to indicate sub-penny values). These contingency types apply only to equity orders and the CBSX trading session. The order price field on these orders will be set as 'Market.'

The contingency information sent on the paired order set will have the following two characteristics:

1. An indication that the paired order set should cross at the NBBO Bid or the NBBO Offer ('Bid peg' vs. 'Offer peg').
2. A numeric value stored in the contingency price field representing how many pennies (including up to 4 decimal places to indicate sub-penny values) better than the NBBO Bid or the NBBO Offer at which this paired order set should trade.

If the specified increment should cause the price to go outside the NBBO (worse than the NBBO), the system should only increment to the other side of the NBBO. For example:

If the NBBO is 0.82-0.83 and the sender specifies on the paired order set that the contingency is a 'bid peg' with a contingency price of 2, the expected crossing price would be 0.84, but because it violates the NBBO, the system should cross the orders at 0.83 (assuming that there is no book being traded through).

IDL Interfaces

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

Module cmConstants

```
interface ContingencyTypes
{
    const cmOrder::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
(ISO)

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;
const cmiOrder::ContingencyType INTERMARKET_SWEEP_BOOK = 28; // Intermarket
Sweep Book (ISB)

const cmiOrder::ContingencyType BID_PEG_CROSS = 29;
const cmiOrder::ContingencyType OFFER_PEG_CROSS = 30;
};

```

Document Changes

API-01

- No changes

API-02

- Updated the contingency mapping table to include BID_PEG_CROSS and OFFER_PEG_CROSS

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