# NEO Exchange

# NITCH Market Data Specification

Private & Confidential



**Date:** 19 February 2019
**Version:** 1.08a - Final

**TABLE OF CONTENTS**

# 1 Referenced Documents

## 1.1 Referenced Documents

| Related Documents Title | Version |
|---|---|
| NEO Exchange - Connectivity Guide | 1.05 |

## 1.2 Document Version History

| Version | Comments / Revision Type | Date |
|---|---|---|
| 1.00 | Initial document publication | Jul 16 '14 |
| 1.01 | Document revisions and modifications<br>- Renaming Protocol Specification to NITCH<br>- S8.6.2 – Add'n of Security Subtype Definitions<br>- S10-S11 – Appendix / reference data information addition<br>- Added description information on handling the Auction Bulk Message<br>- Added additional information about All Books Book Type definition and use<br>- Added Appendix II to support conversion of Order and Trade Identifiers | Sept 16 '14 |
| 1.02 | Updates to:<br>- S7.1.2.7 – Update to Market Data Group Identifier from "A" to "a"<br>- S8.6.3 - Added new session change reason for Auction Execution<br>- S8.6.4, S8.6.8 – Change to value for SettlementType<br>- S9.1.3– updates to values to Trading Status to include Post-Close session<br>- S4.1.1.1 and S4.1.1.2 – Updates to the Closed Session transition<br>- S7.1.1.5 & S7.1.2.13 – Updates to connection termination behaviour for Replay and Recovery channels | Nov 4 '14 |
| 1.03 | Updates to: | Mar 13 '15 |

| | | |
|---|---|---|
| | - S4.1.1.1 & S4.1.1.2 – Add'l descriptions on expected handling during each trading session and corrections<br>- S5.1 – Clarification on Auction Individual volume<br>- S7.1 – Add'l description on recipient failure handling<br>- S7.1.2.7, S7.1.2.8, S7.1.2.9, S7.1.2.10, S7.1.2.11, S7.1.2.12 – Correcting status values<br>- S8.6.2 – Add'n of SecuritySubType value<br>- S9.1.3 – Add'l descriptions on expected handling during each Trading Status | |
| 1.03a | Updates to:<br>- S5.1 – Spec point cleanup | Mar 20 '15 |
| 1.04 | Updates to:<br>- S4.2.4 – Spec point cleanup<br>- S5.1 – Spec point cleanup<br>- S9.1.3 – Add'n of Trading Status value | Nov 13 '15 |
| 1.05 | Updates to:<br>- S8.6.11 – Add'n of Derivative Cross Type | Jun 14 '17 |
| 1.06 | - S4.1.6.1 – Spec point cleanup<br>- S9.1.2 – Added the Segment Table<br>- S8.6.2 – Added the Segment Table<br>- Rebranded document | Nov 2 '17 |
| 1.07 | Updates to:<br>- S4.1.1.1 – Add'l description for handling of NEO Connect instruments<br>- S4.2 – Add'l description for handling of NEO Connect instruments<br>- S5.2 – Add'l description for stats dissemination for NEO Connect instruments<br>- S8.6.2 – Add'n of new Segment and Security Type<br>- S9.1.2 – Add'n of Segment value<br>- S9.1.3 – Add'n of Trading Status value | Jan 15 '18 |
| 1.08 | Updates:<br>- S3 – Added reference to NEO-D Book<br>- S4.1.1.1 – Added reference to NEO-D Book in Pre-Open session | Jun 7 '18 |

| | | |
|---|---|---|
| | - S4.1.1.2 – Added reference to NEO-D Book in Pre-Open<br>- S8.6.2 – Added NEO-D Book to allowed book types<br>- S8.6.2 – Added FOTS to Segment<br>- S9.1.1 – Added NEO-D Book to Book Types<br>- S9.1.2 – Added FOTS to Segments<br>- S9.1.3 – Added reference to NEO-D Book in Trading Status for Pre-Open | |
| 1.08a | Updates:<br>- S2.1 – Spec point cleanup | Jun 25 '18 |

## 1.3 Terms & Definitions

| Term | Definition |
|---|---|
| **DR** | Disaster Recovery Site |
| **PDC** | Primary Data Centre |
| **SDC** | Secondary Data Centre |
| **NEO** | NEO Exchange |
| **NEO-L** | Trading Book |
| **NEO-N** | Trading Book |
| **NEO-D** | Trading Book |

# 2 Overview

The market data feed is a stream of fixed width binary messages that provide real time information for each instrument supported by NEO. The following set of information will be available within NEO  market data solution:

Full depth incremental order book service with instrument status updates

1. Trades reported by NEO.
2. A set of statistics; some calculated by the exchange
3. Instrument reference data.
4. Announcements by NEO.

## 2.1 Production Hours of Operation

The feed will operate from *06:00:00* to *18:45:00* (ET) each trading day.

## 2.2 Supported Venues

The feed will disseminate data for the following venue.

1. NEO

## 2.3 Support

For any questions or general enquiries regarding this document, please contact NEO Exchange Operations.

**E-mail:** exchangeoperations@neostockexchange.com

**Phone:** 416-933-5950

# 3 System Architecture

Supporting the distribution of NEO public market data, all information will be disseminated from a single real time multicast channel. This feed will include all NEO-L, NEO-N, NEO-D, NEO Connect and Crossing trading books data on the same channel. The NEO-L will be represented as a Market by Order book and the NEO-N as a Market by Price book. The NEO Connect data and NEO-D and Crossing trading books will not disseminate order events, only trades.

Two TCP based recovery channels are available supporting the distribution channel; Replay and Recovery. For small gaps, a client may connect to the replay channel in order to gap fill recent order book data loss. If large gaps are experienced, a client should use the recovery channel.

## 3.1 Real time channel

The real-time channel is the primary means of disseminating market data. Real time updates to instruments and all market data supported by the feed are available on this multicast channel.

A basic set of information regarding active instruments in the market data group will be published at the start of the trading day.

This channel will publish full depth order book information on a real time incremental basis. Details of each trade are published on the order book channel while a set of statistics will also be disseminated.

In addition to the above, announcements published by NEO will be disseminated on this channel.

Each application message includes a sequence number which is incremented by one for every message disseminated on the Real time channel. The sequence number is reset to 1 at the start of each valid trading day.

The server will send a Heartbeat message to exercise the communication line during periods of inactivity. A Heartbeat will be sent every <2> seconds when the Real time channel is inactive.

Recipients have access to two identically sequenced real time feeds (Feed A and Feed B) from both the PDC and SDC. Recipients may process both feeds within a single site

and arbitrate between them to minimise the probability of a data loss. Arbitration of the real time feed between PDC and SDC are not supported.

## 3.2 Replay Channel

The TCP replay channel permits recipients to request the retransmission of a limited number of messages already published on the real time feed. This channel may be used by recipients to recover from a small data loss.

The replay channel supports the retransmission of the last *<100,000>* messages published on each real time channel. The channel does not support the retransmission of messages published on previous trading days or those published on the recovery channel.

## 3.3 Recovery Channel

The TCP snapshot recovery channel may be used by recipients to recover from a large-scale data loss. The following sets of data is available on this channel:

1. Recovery of the order book for any active instrument in the market data group.
2. Reference data of instruments.
3. All the trades or a set of trades reported during the day.
4. Recovery of the statistics.
5. All the announcements published during the day.
6. Current Status of an instrument.

# 4 Service Description

## 4.1 Overview of a Trading Day

### 4.1.1 Trading on the Order Book

A typical trading day at NEO will consist of several scheduled sessions. The transition of these sessions will be communicated via Instrument Status messages, with the relevant Trading status and the Book Type. The following are some of the key trading sessions and their respective message flows.

#### 4.1.1.1 NEO Exchange listed instruments

| Session | Description |
|---|---|
| Market Open | The market data feed begins. Recipients should aim to join the feed at this time.<br>A System Event message will be published as the first application message for the day with the Event Code "O".<br>Instrument Directory messages will then be published for each instrument. |
| Pre-Trading | An Instrument Status message will be broadcast with the status 'y' to commence this session.<br>Clients may receive Order book related messages during this session as a result of GTC / GTD orders being re-entered into the market. |
| Pre-Open | An Instrument Status message will be broadcasted with the status 'o' to commence this session.<br>For NEO Connect instruments, this session will be used for AM auctions to execute orders entered on T+1 basis. Trades will be executed at the start of the Pre-Open session and be reported as a set of Trade messages.<br>Clients will receive NEO-D related messages during this session. |
| Pre-Open | The trading day begins. Details of order books will be published.<br>An Instrument Status message will be broadcasted with a status 'a' to commence this session.<br>Indicative opening information will be disseminated via the Statistics Update message. |

| | |
|---|---|
| | Trades are executed at the end of the Pre-Open session. The trades executed in the opening auction will, for each instrument, be reported as a set of Trade messages.<br><br>This session is only applicable for the NEO-L. |
| Post-Open | Continuous trading begins.<br><br>An Instrument Status message will be broadcasted with a status 'T' to commence the session.<br><br>Order book updates and trades for each instrument will be disseminated real time.<br><br>Order book updates will be not be disseminated for the NEO Connect instruments. |
| MOC Imbalance | Indicative closing information will be disseminated via the Statistics Update message.<br><br>The trades executed in the closing auction will be reported as individual trade prints via the Trade message with trade type set to 2.<br><br>This session is only applicable for NEO-L. |
| Pre Close | An Instrument Status message will be broadcast with the status 'r' to commence this session.<br><br>This session is only applicable for NEO Connect. |
| Closing Price Publication | The closing price of each instrument will be published via the Statistics Update message. |
| Extended Trading | Continuous trading at the closing price begins.<br><br>An Instrument Status message will be broadcasted with a status 'u' to commence the session.<br><br>Order book updates, trades and statistics updates will be disseminated during this session.<br><br>This session is only applicable for NEO-L. |
| Closed | At the start of this session an Instrument Status message will be broadcast for each tradable instrument with the Trading Status "b". |
| Market Close | The end of the trading day.<br><br>An Instrument Status message will be broadcast for each tradable instrument with the Trading Status "c".<br><br>Clients may also receive Trade messages during this session for the trades which are cancelled. |
| End of Day | The market data feed will stop at this time. |

| | |
|---|---|
| | A System Event message will be published with the Event Code "C". This will be the last application message for the day. |

#### 4.1.1.2 Non-NEO listed instruments

| Session | Description |
|---|---|
| Market Open | The market data feed begins. Recipients should aim to join the feed at this time.<br>A System Event message will be published as the first application message for the day with the Event Code "O".<br>Instrument Directory messages will then be published for each instrument. |
| Pre-Trading | An Instrument Status message will be broadcast with the status 'y' to commence this session.<br>Clients may receive Order book related messages during this session. |
| Pre-Open | An Instrument Status message will be broadcasted with the status 'o' to commence this session.<br>Clients will only receive NEO-D related messages during this session.<br>This session is only applicable for NEO-D. |
| Post-Open | Continuous trading begins.<br>An Instrument Status message will be broadcasted with a status 'T' to commence the session.<br>Order book updates and trades for each instrument will be disseminated real time. |
| Closing Price Publication | The closing price of each instrument will be published via the Statistics Update message. |
| Closed | At the start of this session an Instrument Status message will be broadcast for each tradable instrument with the Trading Status "b". |
| Market Close | The end of the trading day.<br>An Instrument Status message will be broadcast for each tradable instrument with the Trading Status "c".<br>Clients may also receive Trade messages during this session for the trades which are cancelled. |
| End of Day | The market data feed will stop at this time. |

| | |
|---|---|
| | A System Event message will be published with the Event Code "C". This will be the last application message for the day. |

### 4.1.2 Trading Halt

An Instrument Status message with the relevant Halt Trading Status will be published if an instrument or an order book is temporarily halted. If trading is later resumed, an Instrument Status message, with the appropriate Trading Status (e.g. Post-Open) and Book Type, will be published.

An Instrument Status message, with the relevant Halt Trading Status, will be published if an instrument is halted across multiple days. Clients will receive an Instrument Status message with the relevant Halt Trading Status if it is still in halted state at start of day. It will be published for each Book Type at the start of the first scheduled session (i.e. Pre-Trading).

Instrument Status messages, with the current active status of the instrument and the order books will be published if the halt is lifted during trading hours.

### 4.1.3 No Matching Halt

An Instrument Status message with the Trading Status "l" (lowercase of "L") will be published if on-book trading for an instrument is no matching halt during the day. An Instrument Status message, with the appropriate Trading Status (e.g. re-opening auction call) will be published if trading is later resumed.

### 4.1.4 Re-Opening Auction Call

An Instrument Status message, with a Trading Status "e", will be published if an instrument is moved to the Re-Opening Auction Call during the day. An Instrument Status message, with the appropriate status (e.g. Post-Open) will be published if trading is later resumed.

### 4.1.5 Instrument Suspension

An instrument may be suspended during or outside trading hours. The Instrument Status message will be published, with a Status of "2", if an instrument is suspended during trading hours.

If, at the start of a trading day, an instrument is still in a suspended state, an Instrument Status message will be published, with a Status of "2". If the suspension is lifted during

the trading day recipients will receive an Instrument Status message, with the current active trading status of the instrument and the order book.

### 4.1.6 Intra-Day Trading Session Updates

#### 4.1.6.1 Extension of a Pre-Open Session

An Instrument Status message will be broadcast with the value "3" or "4" in the field Session Change Reason to indicate an extension of a Pre-Open session. This message will also include the new time at which the auction will take place.

#### 4.1.6.2 Adjustment by Market Operations

NEO may extend or shorten a particular trading session. In such a case, an Instrument Status message will be broadcast with the value "1" or "2" in the field Session Change Reason. The message will indicate the new time at which the session will end.

### 4.1.7 New Instruments

New instruments may be created during the trading day. In such a case, the server will publish an Instrument Directory message to notify recipients of the details of the new instrument. Separate Instrument Status messages will also be published to indicate the current trading status of the instrument and the order book.

## 4.2 Full Depth Incremental Service

This is a market data feed that provides the order depth for the entire order book. This includes two types of feeds, one per order, and the other per price point. It provides adequate displayable information (E.g. side, price and displayed quantity) of each active order.

Details of all active orders will be sent at the start of the market open session in which the order book is published. Thereafter, the order book updates will be sent incrementally in real time.

This service also includes Trades, Statistics, Instrument Status, Instrument Reference Data and System Event data for the relevant instruments as detailed below.

In this service, Order book related market data will be disseminated for two visible book types: NEO-L and NEO-N, and this can be differentiated via the field 'Order Book Type' defined in each of these messages. The service for NEO-L will be market by order (MBO) feeds and market data related to the NEO-N will be represented as a market by price

(MBP) feed. For NEO Connect instruments, MBO and MBP services will be not be disseminated.

### 4.2.1 Order Book Management – MBO

In this service, all order book updates will be published using Add Order Incremental, Delete Order and Modify Order messages.

#### 4.2.1.1 Adding an Order

An Add Order Incremental message will be sent each time a new visible order is added to the order book.

These messages contain a unique identifier (across instruments and days) that can be referenced for all future updates (delete and modify) of the order.

The recipients of this message would be able to identify the type of the order (Market or Limit) from the message explicitly. This would enable clients to identify market orders from limit orders during an Auction.

#### 4.2.1.2 Deleting an Order

A Delete Order message will be sent to notify recipients if a displayed order is fully executed ("filled"), cancelled, or expired. The OrderID will be included in the message to locate and remove the original order from the order book.

#### 4.2.1.3 Modifying an Order

A Modify Order message will be sent if the displayed quantity of an order or its price is changed or if an order loses time priority. The OrderID will be included in the message to locate and update the original order in the book.

#### 4.2.1.4 Executions

A Trade message will be sent whenever an order is executed. Every execution that affects the displayable properties of an order will also result in a Delete message (to remove the respective order from the book if the order is fully executed) and/or a Modify message (to update the displayable properties of the order at multiple executions).

For example, Iceberg order execution with multiple replenishments may result in several Trade messages to report the executions of both the hidden and displayable portions of it, along with Delete and Add Order Incremental messages to reflect the respective displayable changes on the order book.

### 4.2.2 Order Book Management – MBP

In this service, all order book updates will be published using Add Order Incremental MBP, Delete Order MBP and Modify Order MBP messages.

#### 4.2.2.1 Adding an Order

An Add Order Incremental MBP message will be sent each time a new visible price point is added to the order book.

#### 4.2.2.2 Deleting an Order

A Delete Order MBP message will be sent to notify recipients if a displayed quantity of the specified price point is cancelled, fully executed or expired. Recipients should use the field 'Previous Price' to identify the price point to be removed.

#### 4.2.2.3 Modifying an Order

A Modify Order MBP message will be sent if the displayed quantity of a price point or its price is changed. Recipients should use the field 'Previous Price' to identify the price point to be modified. It should be modified with price and quantity values specified in 'New Price' and 'New Quantity' fields respectively.

#### 4.2.2.4 Executions

A Trade message will be sent whenever an order is executed. Every execution that affects the displayable properties of an order will also result in a Delete Order MBP message (to remove the respective price point from the book if quantity is fully executed) and/or a Modify Order MBP message (to update the displayable properties of the order at multiple executions).

For example, iceberg order execution with multiple replenishments may result in several Trade messages to report the executions of both the hidden and displayable portions of it, along with Delete Order MBP and Add Order Incremental MBP / Modify Order MBP messages to reflect the respective displayable changes on the order book.

### 4.2.3 Announcement Service

This service redistributes announcements published by the source market via the Announcements (0x75) message on the same channel used for order management.

### 4.2.4 Indicative Auction Information

The market data feed provides recipients with the indicative price for each auction (i.e. opening, closing, etc.) via the Statistics Update message. The indicative auction price is disseminated along with the imbalance quantity and the side of the imbalance at this price.

A Statistics Update message will be sent with auction information when the indicative auction price is updated or the imbalance quantity and/or side changes.

If an indicative auction price does not exist (i.e. the order book is not locked or crossed) or if there is no imbalance at the indicative price, the Imbalance Direction field of the message will be “O” or “N” respectively.

# 5 Common Data Items

This section describes data items that are disseminated on services identified in the Service Description section.

## 5.1 Trades

All trade related data will be disseminated using the Trade (0x50) message. Recipients will be able to identify Regular, Auction individual and Auction bulk trades separately by referring to the field 'Trade type' specified in the message. The fields 'Cross type' and 'Book type' can be used by the clients to identify Cross trades. Trade cancels (breaks) and Trade corrections (amends) are also disseminated using the Trade (0x50) message. Designated trade types '9' and '11' will be used to indicate a trade break and a correction respectively. These messages will contain an ID (Trade ID) and book type combination that identifies Trade messages reported earlier.

**Note that for the Auction Individual Trade, the volume is reported as Buy Volume and Sell Volume. For the purpose of reporting volume traded in an auction using the Auction Individual Message, the volume should be divided by two.**

Recipients of this message should not update the last sale price for certain trades, such as odd lot trades, SST trades and cross trades with Bypass, Basis, VWAP and Non-NEO cross types. This will be indicated in the 1st bit of the flag field specified in the message.

In addition, trades executed against a bypass order will also be indicated from the 2nd bit of the flags field. The 3rd bit of the flags field will indicate whether the resulting Trade message is due to a manual trade correction.

**Note that the unique identifier for a trade is unique across instruments across days.**

## 5.2 Statistics

A set of statistics will be disseminated via Statistics (0x77) and Statistics Update (0x6a) messages on the real time channel. In addition, Statistics Recovery (0x77) message will

be used for recovery purposes. The different statistics have been grouped into the two real time messages based on their update frequency.

Statistics that are likely to get updated with every trade are included in the Statistics message. All the statistics included in this message will be published as separate fields with every message.

Statistics that are likely to get updated with only some trades have been included in the Statistics Update message. This message contains a STATISTICS TYPE field which can indicate one of many possible statistics and a set of common fields that can contain the relevant values for the said statistics.

Statistics Update message will be used to indicate when the the NAV is updated during the day prior to the PM or AM auction.

Statistics Recovery message contains all the different statistics that can be disseminated via both the real time messages as separate fields. This will be available on the recovery channel only to recover the latest recovery.

## 5.3 Instrument Status

NEO will disseminate data relating the trading status of instruments via the Instrument Status (0x48) message. This will be disseminated whenever an update from a source execution venue is sent to indicate that the trading status of an instrument has changed along with the reason for a session change (if applicable).

This message will also be sent to identify the shortening or extending of a trading session. In such a case, the new end time will be disseminated along with the appropriate reason.

## 5.4 Instrument Reference Data

NEO will use the message Instrument Directory (0x70) to represent all instruments. These messages will contain the instrument reference data and will be available on both real time and in the recovery channel on a request basis.

Instrument Directory messages will be published on the real time channel at the start of the trading day for all the instruments serviced by a particular channel. It will also be disseminated during the trading day if a new instrument is added or if a reference data attribute is updated.

When Instrument Directory messages are disseminated on the real time channel, an Instrument Status message will also be published for each instrument if the status of the instrument is not 'Active'. (The absence of an Instrument Status message implies that the instrument is in an 'Active' state).

## 5.5 System Events

System level data from NEO (such as start of day and end of day) will be redistributed via the System Event (0x53) message.

# 6 Connectivity

## 6.1 Transmission Standards

### 6.1.1 Multicast Channels

The Real time channel utilises UDP over IP version 4 (IPv4) Ethernet standards. UDP header information is as defined in the IETF RFC 791 (IPv4) and RFC 768 (UDP) transmission standards. Each UDP packet will contain just one Unit Header.

### 6.1.2 Unicast Channels

The Replay and Recovery channels utilise TCP over IP version 4 (IPv4) Ethernet standards. TCP header information is as defined in the IETF RFC 793 standard and IPv4 is as defined in the RFC 791 standard.

## 6.2 UserIDs

The UserID of each client wishing to connect to the Replay and Recovery channels must be requested and subsequently provisioned by NEO Exchange Operations before communications can begin. Each UserID will be assigned a password on registration.

A UserID may, at any particular time, only be logged in to one TCP channel.

## 6.3 Production IP Addresses and Ports

The IP addresses and ports of the real time, replay and recovery channels for each market data group are contained within the NEO Exchange Connectivity Guide.

## 6.4 Bandwidth

The recommended bandwidth for the real time, replay and recovery channels are contained within the NEO Exchange Connectivity Guide.

# 7 Recovery

## 7.1 Recipient Failures

Recipients have access to two identically sequenced real time feeds (Feed A and Feed B) from both PDC and SDC. Recipients may process both feeds from a single site and arbitrate between them to minimize the probability of a data loss.

If a gap in sequence numbers is detected on the real time channel, the recipient should assume that some or all of the order books maintained on its systems are incorrect and initiate one of the recovery processes outlined below.

Please note that connecting to replay and recovery channels is restricted via a series of permission codes that are assigned to each UserID.

Client initiated administrative NITCH messages submitted to the replay and recovery channels must contain a NITCH Unit header, as described in section 8.3. The message count and message block length specified in the unit header of a client initiated administrative message must be valid; however, the sequence number and market data group fields will not be interpreted by the system. If the system is unable to process a client initiated administrative message due to a missing or invalid unit header, no data will be returned to the client and the TCP/IP connection will eventually be terminated due to exceeding the configurable idle timeout.

### 7.1.1 Replay Channel

The TCP replay channel should be used by recipients to replay from a small-scale data loss. It permits clients to request the retransmission of a limited number of messages already published on the real time channel. The channel supports the retransmission of the last *<100,000>* messages published on the real time channel.

Each UserID may login to the Replay channel up to *<10>* times each day. The total number of Replay Requests that a client may send for a particular market data group is also limited to *<10>* each day. Recipients may request NEO Exchange Operations to reset its login and request counts. This feature is intended to help manage an emergency situation and should not be relied upon as a normal practice.

If a client submits multiple requests on the Replay channel, they will be processed serially (i.e. one at a time). Active requests of multiple clients will be served on a round

robin basis (time sliced per client). Clients are unable to cancel outstanding Replay Requests.

#### 7.1.1.1 Establishing a Connection

The client should use the relevant IP address and port (as outlined in Section 6.3) to establish a TCP/IP session with the Replay channel. The client should initiate a session by sending the Login Request message. The client should identify itself by specifying its UserID and the respective password. The server will validate the UserID, Password and IP address of the client. The IP address validation can be configured to ignore, if such validation is not required for certain clients.

Once the client is authenticated, the server will respond with a Login Response message with the Status "A".

The client must wait for the server's Login Response before sending additional messages. Messages received from the client before the exchange of logons will be ignored.

If a logon attempt fails because of an invalid UserID, Password or IP address, the server will send a Login Response where Status = "f" and then break the TCP/IP connection with the client.

If a logon attempt fails because of a locked UserID or if logins are not currently permitted, the server will send a Login Response where Status = "a" and then break the TCP/IP connection with the client.

If a client has already logged into the Replay channel the maximum allowed number of times during the current day, the server will reject any new logon attempt with a Login Response and then break the TCP/IP connection. The Status of such a Login Response message will be "b".

The replay channel supports only a limited number of concurrent logins across all clients. This is limited to *<500>* logins across all clients. Once the number of active logins has reached this limit, the server will reject login requests from additional clients with a Login Response and then break the TCP/IP connection. The Status of such a Login Response message will be "d".

A log on attempt to the replay channel made by an already logged in client or by a client awaiting a Login Response to a prior Login Request will receive a Login Response Message with Status = e [Failed (Other)].

If the system is not ready to provide replay data due to an internal reason, the server will reject login requests via a Login Response with Status = c [Service Unavailable] and then break the TCP/IP connection.

Other messages sent prior to the login being established (i.e. before a replay response accepting the request is sent) will be ignored by the system.

If a Login Request is not received within *<5>* seconds of the establishment of a TCP/IP connection or a Replay Request is not received within *<5>* seconds of a successful logon, the server will break the TCP/IP connection with the client.

#### 7.1.1.2 Heartbeats

The server will not send heartbeats on the replay channel during periods of inactivity.

#### 7.1.1.3 Requesting Missed Messages

Once connected to the replay channel, clients may use the Replay Request message to request the retransmission of missed messages. The request should include the sequence number of the first message in the range to be retransmitted along with the number of messages to be retransmitted. The client may also specify a Request ID for each Replay Request. This ID will be echoed back to the client in the Replay Response and Replay and Recovery Complete messages.

#### 7.1.1.4 Response to a Replay Request

The server will respond to the Replay Request with a Replay Response message to indicate whether the retransmission request is successful or not. A Status other than "A" will indicate that the request has been rejected.

The following are the other possible Status values and an explanation of when they will be sent:

| Status | Description |
|---|---|
| O | Request out of range. The retransmission request includes one or more messages that are not in the server's cache. (Requesting for more messages than what's currently available in the cache will also result in this.) Note that the entire request is rejected. |
| D | The UserID has completed the number of requests allowed for a day. |

| | |
|---|---|
| U | No data in cache. |
| c | The number of open replay requests from the UserID has reached the maximum allowed number. (The maximum allowed number is set to *<10>*) |
| e | The system is unable to service the replay request for any other reason. |

In the case of a successful request, the server will retransmit the requested messages immediately after the Replay Response. If a Request ID was specified in the Replay Request message, it will be included in the Replay Response message. The sequence numbers of the retransmitted messages will be the same as when they were first disseminated on the Real time channel. The framing of the replayed messages inside of Unit Headers may differ between the original transmission and the retransmission. Once all the messages have been transmitted, the server will send a Replay and Recovery Complete message that includes the client specified Request ID to indicate that the message replay has successfully completed. The field Trading Status will be empty in this message.

#### 7.1.1.5 Termination of the Connection

After a configurable idle time period following the Replay and Recovery Complete message for the last served Replay Request has been sent the server will break the TCP/IP connection with the client (i.e. there is no requirement for an explicit logout request).

### 7.1.2 Recovery Channel

The TCP Recovery channel should be used by recipients to recover from a large-scale data loss (i.e. late joiner or major outage). It permits clients to recover the following types of data:

1. Order book recoveries
2. All trades reported within the trading date
3. Recovery of statistics
4. Current instrument status
5. Full set of instrument reference data

6. All announcements published within the trading day

Each UserID may login to the Recovery channel a limited number of times each day. The total number of Recovery Request messages that a client may submit for a particular market data group is also limited for a given day. Recipients may request NEO Exchange Operations to reset its login and request counts. This feature is intended to help manage an emergency situation and should not be relied upon as a normal practice.

If a client submits multiple requests on the recovery channel, they will be processed serially (i.e. one at a time). Active requests of multiple clients will be served on a round robin basis. Clients are unable to cancel outstanding Recovery Requests.

#### 7.1.2.1 Establishing a Connection

The client should use the relevant IP address and port (as outlined in Section 6.3) to establish a TCP/IP session with the Recovery channel. The client should initiate a session by sending the Login Request message. The client should identify itself by specifying its UserID in the Username field and the Password. The server will validate the UserID, Password and IP address of the client.

Once the client is authenticated, the server will respond with a Login Response message with the Status "A".

The client must wait for the server's Login Response before sending additional messages. Messages received from the client before the exchange of logons will be ignored.

If a logon attempt fails because of an invalid UserID, Password or IP address, the server will send a Login Response where Status = "f" and then break the TCP/IP connection with the client.

If a logon attempt fails because of a locked UserID or if logins are not currently permitted, the server will send a Login Response where Status = "a" and then break the TCP/IP connection with the client.

If a client has already logged into the Recovery channel the maximum allowed number of times during the current day, the server will reject any new logon attempt with a Login Response and then break the TCP/IP connection. The Status of such a Login Response message will be "b".

The Recovery channel supports only a limited number of concurrent logins across all clients. This is limited to *<500>* logins across all clients. Once the number of active

logins has reached this limit, the server will reject login requests from additional clients with a Login Response and then break the TCP/IP connection. The Status of such a Login Response message will be "d".

A log on attempt to the Recovery channel made by an already logged in client or by a client awaiting a Login Response to a prior Login Request will receive a Login Response Message with Status = e [Failed (Other)].

If the system is not ready to provide recovery data due to an internal reason, the server will reject login requests via a Login Response with Status = c [Service Unavailable] and then break the TCP/IP connection.

Other messages sent prior to the login being established (i.e. before a Recovery Response accepting the request is sent) will be ignored by the system.

If a Login Request is not received within *<5>* seconds of the establishment of a TCP/IP connection or a Recovery Request is not received within *<5>* seconds of a successful logon, the server will break the TCP/IP connection with the client.

#### 7.1.2.2 Heartbeats

The server will not send heartbeats on the recovery channel during periods of inactivity.

#### 7.1.2.3 Requesting Recovery Data

Once connected to the recovery channel, clients may use the Recovery Request message to receive the desired set of data by specifying a combination of Recovery Type, Request Level (refer section below for details), Order Book Type and Source venue. The client may specify a Request ID for each Recovery Request message. This will be echoed back in the Recovery Response and Replay and Recovery Complete messages.

#### 7.1.2.4 Recovery Levels

When making recovery requests, the clients can specify a level which determines the data set that is returned. The different levels that can be specified are:

- **Instrument**: With this, a valid Symbol has to be specified. If not, the request will be rejected with status 'a'. Only the data related to the specified Symbol will be sent.

- **Group (Segment)**: With this, a valid Venue Group ID (segments: e.g. TSX, TSXV, etc.) and a source venue has to be specified. If not specified or if no such Venue Group ID is valid for the specified source market, the request will be rejected with status 'a'. Data related to all instruments assigned to the specified group will be sent.

**Channel**: Data related to all the instruments serviced by the channel will be sent. If a client specifies an Instrument ID and/or Venue Group ID with this level, they will be ignored and the request will be considered as for the entire channel.

#### 7.1.2.5 Response to a Recovery Request

The server will transmit a Recovery Response to indicate whether a Recovery Request is accepted or rejected. A Status other than "A" will indicate that the request is rejected.

The following are the other possible Status values and an explanation of when they will be sent:

| Value | Status | Description |
|---|---|---|
| O | Out of Range | Request out of range. It's possible to specify a starting sequence number with certain Recovery Types. This status is used if the specified sequence is not available with the recovery service. |
| a | Group or Symbol Not Specified or Invalid | Symbol not specified for an instrument level request or Venue Group ID not specified for a group level request. Also returned if an invalid Symbol or group ID is specified. |
| b | Request Limit Reached | The UserID has completed the number of requests allowed for a day. |
| c | Concurrent Limit Reached | The number of open recovery requests from the UserID has reached the maximum allowed number. |
| d | Invalid Recovery Type or Request Level | Invalid Recovery Type or Request Level specified. |
| e | Failed (Other) | The system is unable to service the recovery request for any other reason. |

The following sections detail the expected behaviour with successful requests for different Recovery Types.

#### 7.1.2.6 Order Book Recovery – Instrument Level Request

The server will first transmit a Recovery Response message where the status is "A". This will include the message sequence number of the Real time channel with which the instrument's order book recoveries will be synchronised. The response will also include the total number of messages to follow the Recovery Response. The client should buffer all messages on the Real time channel for the instrument with sequence numbers greater than that specified in the Recovery Response.

The Recovery Response will be followed by a series of Add Order Incremental / Add Order Incremental MBP messages to construct the order book recovery. Each order / Price point in the current recovery will be represented by an Add Order Incremental / Add Order Incremental MBP message in this service. The first Add Order Incremental / Add Order Incremental MBP message transmitted by the server will contain the order with the best bid price followed by the full depth of the book for the buy side; after which the server will transmit the order with the best offer price followed by the full depth of the book for the sell side. If the request was for all book types and if there's more than one order book with active orders for the given instrument, recoveries will be provided for all such order books.

The server will transmit a Replay and Recovery Complete message following each recovery for a particular instrument/order book type combination. The message will include the current Trading Status of the instrument/order book type at the relevant market. Once all the recoveries and their complete messages have been sent, an additional request level Replay and Recovery Complete message is sent to indicate the end of service for the particular request. The client may begin processing the buffered messages for the instrument from the Real time channel once this request level Replay and Recovery Complete is received.

The server will not publish a book level Replay and Recovery Complete message for order books that do not have any orders in them. The Recovery Response will be immediately followed by a request level Replay and Recovery Complete message if there are no published orders for any of the order books of the instrument.

Recovery Response message and all Replay and Recovery Complete messages will include the corresponding Request ID if it was specified in the Recovery Request message.

#### 7.1.2.7 Order Book Recovery – Group or Channel Level Request

The server will first transmit a Recovery Response message where the status is "A". This will include the message sequence number of the Real time channel with which the order book recoveries will be synchronized. The response will also include the total number of messages to follow the Recovery Response. The client should buffer all messages on the Real time channel for the instrument with sequence numbers greater than that specified in the Recovery Response.

If the request is successful, the server will then disseminate recoveries of all order books with active orders for all instruments in the requested venue group or for the entire channel. The Recovery Response will be followed by a series of Add Order Incremental / Add Order Incremental MBP messages to construct the order book recovery. Each order / Price point in the current recovery will be represented by an Add Order Incremental message / Add Order Incremental MBP in this service. The first Add Order Incremental / Add Order Incremental MBP message transmitted by the server will contain the order with the best bid price followed by the full depth of the book for the buy side; thereafter the server will transmit the order with the best offer price followed by the full depth of the book for the sell side. Order book recoveries for the requested instruments will be transmitted serially (i.e. one instrument at a time). If an instrument has active orders in more than one order book, recoveries for all the order book types for that instrument will be disseminated before moving on to the next instrument.

The server will transmit a Replay and Recovery Complete message following each recovery for a particular instrument/order book type combination. The message will include the current Trading Status of the instrument/order book type at the market. Once all the recoveries and their complete messages have been sent, a request level Replay and Recovery Complete message is sent to indicate the end of service for the particular request. The client may begin processing the buffered messages for the instrument from the Real time channel once this request level Replay and Recovery Complete is received.

The server will not publish a book level Replay and Recovery Complete message for order books that do not have any orders in them.

The Recovery Response will be immediately followed by a request level Replay and Recovery Complete message if there are no published orders for any of the order books for any of the instruments.

Recovery Response message and all Replay and Recovery Complete messages will include the corresponding Request ID if it was specified in the Recovery Request message.

#### 7.1.2.8 Instrument Directory Recovery

The server will first transmit a Recovery Response message where the status is "A". The Sequence Number will be 0 and the Count field will include the total number of messages to follow the Recovery Response.

The server will then disseminate either one or a series of Instrument Directory messages depending on the Request Level specified. When responding to Group or Channel level requests, messages will be sent for all instruments disseminated in the real time channel as long as they belong to the group or are assigned to the channel.

The server will transmit a Replay and Recovery Complete message after all the Instrument Directory messages have been disseminated. The Trading Status will not be set in the Recovery Complete message. The Recovery Response will be immediately followed by a request level Replay and Recovery Complete message if the Request Level is 'Group' or 'Channel' and there are no instruments in the group or no instruments are assigned to the channel.

Both Recovery Response and Replay and Recovery Complete messages will include the corresponding Request ID if it was specified in the Recovery Request message.

#### 7.1.2.9 Trades Recovery

The server will first transmit a Recovery Response message where the status is "A". The message will contain the (real time channel's) current sequence number. The Count field will include the total number of messages to follow the Recovery Response.

If the Recovery Request message specified a sequence number, only the trades carrying equal or higher real time channel sequence numbers will be sent in response. It is not possible to filter the recovery response based on the trade message type. Trade messages that represent trade breaks will also be included. The messages will be sent in the same sequence as they were sent on the real time channel.

The server will transmit a request level Replay and Recovery Complete message after all the Trade messages have been disseminated. The Trading Status will not be set in the Recovery Complete message. The Recovery Response will be immediately followed

by the Replay and Recovery Complete message if there are no Trade messages that fit the Recovery Request's specifications.

Both Recovery Response and Replay and Recovery Complete messages will include the corresponding Request ID if it was specified in the Recovery Request message.

#### 7.1.2.10 Statistics Recovery

The server will first transmit a Recovery Response message where the status is "A". The message will contain the sequence number of the message (of any type) that was last sent before the Recovery Request was accepted. The Count field will include the total number of messages to follow the Recovery Response. The client should ideally buffer all messages on the Real time channel for the instrument with sequence numbers greater than that specified in the Recovery Response.

The server will then disseminate either one or a series of Statistics Recovery messages depending on the Request Level specified. This message will contain the current value of all the statistics supported by NEO for each instrument.

The server will transmit a request level Replay and Recovery Complete message after all the Statistics Recovery messages have been disseminated. The Trading Status will not be set in the Recovery Complete message. The Recovery Response will be immediately followed by the Replay and Recovery Complete message if there are no statistics messages that fit the Recovery Request's specifications.

#### 7.1.2.11 Announcements Recovery

The server will first transmit a Recovery Response message where the status is "A". The message will contain the (real time channel's) sequence number of the last announcement to be sent. The Count field will include the total number of messages to follow the Recovery Response.

If the Recovery Request message specified a sequence number, only the Announcement messages carrying equal or higher real time channel sequence numbers will be sent in response. Request Level 'Channel' is implied. The messages will be sent in the same sequence as they were sent on the real time channel.

The server will transmit a request level Replay and Recovery Complete message after all the announcement messages have been disseminated. The Trading Status will not be set in this. The Recovery Response will be immediately followed by the Replay and

Recovery Complete message if there are no announcement messages that fit the Recovery Request's specifications.

Both Recovery Response and Replay and Recovery Complete messages will include the corresponding Request ID if it was specified in the Recovery Request message.

#### 7.1.2.12 Instrument Status Recovery

The server will first transmit a Recovery Response message where the status is "A". The message will contain the sequence number of the message (of any type) that was last sent before the Recovery Request was accepted. The Count field will include the total number of messages to follow the Recovery Response. The client should ideally buffer all messages on the Real time channel for the instrument with sequence numbers greater than that specified in the Recovery Response.

The server will then disseminate either one or a series of Instrument Status messages depending on the Request Level specified. Each message will contain the values contained in the last Instrument Status message sent for the corresponding instrument on the real time channel.

The server will transmit a request level Replay and Recovery Complete message after all the Instrument Status messages have been disseminated. The Trading Status will not be setin the Recovery Complete message. The Recovery Response will be immediately followed by the Replay and Recovery Complete message if there are no Instrument Status messages that fit the Recovery Request's specifications.

#### 7.1.2.13 Termination of the Connection

After a configurable idle time period following the Replay and Recovery Complete message for the last served Recovery Request has been sent, the server will break the TCP/IP connection with the client. I.e. there is no requirement for an explicit logout request.

## 7.2 Failures of NEO Exchange NITCH Feed

### 7.2.1 Recoveries on the Real time channel

In the unlikely event of an outage on the NEO Exchange NITCH Feed, recipients may be required to refresh their order book and statistics displays for one or more instruments.

In such a scenario the server will, on the Real time channel, broadcast the current order book and the latest statistics for each affected instrument. In such an event recipients must discard the contents of their order book and statistics displays for these instruments.

The server will continue with real time message publishing once it is ready to disseminate again.

### 7.2.2 Resetting Sequence Numbers

If the market data feed is, in the unlikely event of an outage, failed over to the backup site or is restarted, the message sequence numbers (line level and instrument level) of the Real time channel will be reset to 1 and the Market Data Group value will change. In such a case, messages sent on the Real time channel prior to the resetting of sequence numbers will not be available for retransmission on the Replay channel.

# 8 Message Formats

This section provides details on the data types, unit header, administrative messages and application messages utilised by the server. For each message, a description of each field is provided along with the applicable data type, offset and length (in bytes).

## 8.1 Data Types

The fields of the messages utilised by the server will support the data types outlined below:

| Data Type | Length | Default | Description |
|---|---|---|---|
| Alpha | Variable | Spaces | These fields use standard ASCII character bytes. They are left justified and padded on the right with spaces. |
| Bit Field | 1 | 0 | A single byte used to hold up to eight 1-bit flags. Each bit will represent a Boolean flag. The 0 bit is the lowest significant bit and the 7 bit is the highest significant bit. |
| Byte | 1 | 0 | A single byte used to hold one ASCII character. |
| Date | 8 | Spaces | Date specified in the YYYYMMDD format using ASCII characters. |
| Time | 6 | Spaces | Time specified in HHMMSS format using ASCII characters in a 24 hour clock format. |
| UDT (Unix Date Time) | 8 | 0 | 64bit unsigned integer where;<br><br>time stamp (in UTC) = (date time per second resolution in Unix time format) * 1,000,000,000 + (nanoseconds component) |
| Price | 8 | 0 | Signed Little-Endian encoded 64bit integer field with eight implied decimal places. Please refer to the Appendix for instructions on handling negative values. |
| Size | 8 | 0 | Little-Endian encoded 64 bit unsigned integer with 8 implied decimal places. |

| | | | |
|---|---|---|---|
| UInt8 | 1 | 0 | 8 bit unsigned integer. |
| UInt16 | 2 | 0 | Little-Endian encoded 16 bit unsigned integer. |
| UInt32 | 4 | 0 | Little-Endian encoded 32 bit unsigned integer. |
| UInt64 | 8 | 0 | Little-Endian encoded 64 bit unsigned integer. |
| Price4 | 8 | 0 | Signed Little-Endian encoded 64bit integer field with 4 implied decimal places. Handling negative values will be similar to that of Price. |
| Size4 | 8 | 0 | Little-Endian encoded 64 bit unsigned integer with 4 implied decimal places. |

Please note that in certain field descriptions the document describes the field as blank in certain circumstances. Blank refers to 'space filled' for data types Alpha; whereas it means '0' (zero) for data types Byte, Price, Size, UInt8, UInt16, UInt32 and UInt64.

## 8.2 Message Overview

### 8.2.1 Administrative Messages

| Name | Message Type ASCII | Hex | Usage |
|---|---|---|---|
| Heartbeat | - | - | Used by the server, on the Real time channel, to exercise the communication line during periods of inactivity. |
| Login Request | (soh) | 0x01 | Used by the client to login to the Replay or Recovery channel. |
| Login Response | (stx) | 0x02 | Used by the server to accept or reject a login request to the Replay or Recovery channel. |

| | | | |
|---|---|---|---|
| Replay Request | (etx) | 0x03 | Used by the client to request a retransmission of messages on the Replay channel. |
| Replay Response | (eot) | 0x04 | Used by the server to respond to a retransmission request on the Replay channel. |
| Recovery Request | □ | 0x81 | Used by the client to request for a recovery of the current order book on the Recovery channel. |
| Recovery Response | , | 0x82 | Used by the server to respond to a recovery request on the Recovery channel. |
| Replay and Recovery Complete | *f* | 0x83 | Used by the server to indicate the successful completion of servicing a message replay or a recovery request. |

### 8.2.2 Application Messages

Applications messages may only be sent by the server.

| Name | Message Type ASCII | Hex | Usage |
|---|---|---|---|
| System Event | S | 0x53 | Sent to indicate the start and end of the day. |
| Instrument Directory | p | 0x70 | Used to disseminate data of instruments on the real time channels. |
| Instrument Status | H | 0x48 | Indicates the current trading status of an instrument. Also used to communicate unscheduled session changes. |
| Add Order Incremental | F | 0x46 | Sent to instruct the recipients to add a new order to the respective displayable book in the MBO service. |
| Delete Order | D | 0x44 | Sent to instruct recipients to remove an order from the book in the MBO service. |

| | | | |
|---|---|---|---|
| Modify Order | U | 0x55 | Sent to instruct recipients to update price and/or size of an order in the book in the MBO service. |
| Add Order Incremental MBP | Q | 0x51 | Sent to instruct the recipients to add a new price point to the respective displayable book in the MBP service. |
| Delete Order MBP | T | 0x54 | Sent to instruct recipients to remove an existing price point from the book in the MBP service. |
| Modify Order MBP | V | 0x56 | Sent to instruct recipients to update price and/or size of an existing price point in the book in the MBP service. |
| Order Book Clear | y | 0x79 | Sent to instruct recipients to remove all orders from the order book for the specified instrument. |
| Trade | P | 0x50 | Sent to represent different types of trades published by markets. |
| Statistics | w | 0x77 | Contains a set of statistics that are updated frequently |
| Statistics Update | j | 0x6a | A type-value message that contains a set of statistics that are not updated frequently. |
| Statistics Recovery | k | 0x6b | A recovery of all statistics that is used for recovery |
| Announcements | u | 0x75 | Used to disseminate announcements |

### 8.2.3 Intelligent Throttling

The system, in order to increase efficiency, will pack multiple messages in a single UDP packet whenever possible. However, this will not apply to messages deemed latency sensitive. These messages will be packed one per UDP packet and be disseminated as and when generated. The following are the messages that are identified as latency sensitive:

- System Event
- Instrument Status
- Trade
- Statistics
- Statistic Update
- Announcements
- Add Order Incremental
- Delete Order
- Modify Order
- Add Order Incremental MBP
- Delete Order MBP
- Modify Order MBP

This functionality will be controlled at gateway level via a configuration where it is possible to enable Intelligent Throttling or to pack all messages in to common packets to optimize bandwidth utilization.

## 8.3 Unit Header

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of the message block including the header and all payload messages. |
| Message Count | 2 | 1 | UInt8 | Number of payload messages that will follow the header. |
| Market Data Group | 3 | 1 | Byte | Identity of the market data group the payload messages relate to. |

| | | | | |
|---|---|---|---|---|
| Sequence Number | 4 | 4 | UInt32 | Sequence number of the first payload message. |
| Payload | 8 | Variable | - | One or more payload messages. |

## 8.4 Administrative Messages (Client – Initiated)

### 8.4.1 Login Request

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x01 Login Request |
| Username | 3 | 8 | Alpha | UserID assigned to the client. |
| Password | 11 | 10 | Alpha | Password assigned to the UserID. Maximum password length is 10 characters. |

### 8.4.2 Replay Request

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x03 Replay Request |
| First Message | 3 | 4 | UInt32 | Sequence number of the first message in range to be retransmitted. |
| Count | 7 | 4 | UInt32 | Number of messages to be resent. |

| | | | | |
|---|---|---|---|---|
| Request ID | 11 | 4 | UInt32 | The value set in this will be echoed back in the corresponding Replay Response. The system will not validate uniqueness of the set value. |

### 8.4.3 Recovery Request

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x81 Recovery Request |
| Request Level | 3 | 1 | UInt8 | Defines the level of the request.<br>**Value** **Meaning**<br>0 Instrument<br>1 Group (Segment)<br>2 Channel |
| Symbol | 4 | 14 | Alpha | Symbol of the instrument if Request Level is 0. Blank if not. |
| Venue Group ID | 18 | 6 | Alpha | Group/segment ID if Request Level is 1. Blank if not. |
| Order Book Type | 24 | 1 | UInt8 | Only considered if the Request Level is 0. If specified, only data related to the specified order book type is provided, If not, data for all available book types for the instrument is provided. |

<table>
<tr><td></td><td></td><td></td><td></td><td>Please refer to the Additional Field Values section of this document for valid values.</td></tr>
<tr><td>Source Venue</td><td>25</td><td>2</td><td>UInt16</td><td>Mandatory field if Request Level is 1. Not considered for other Request Levels.</td></tr>
<tr><td>Recovery Type</td><td>27</td><td>1</td><td>UInt8</td><td>The type of messages to be replayed.<br>Value Meaning<br>0 Instrument Directory<br>1 Order book<br>2 All Trades<br>3 Statistics<br>4 Instrument Status<br>5 Announcements</td></tr>
<tr><td>Sequence Number</td><td>28</td><td>4</td><td>UInt32</td><td>Only valid if Recovery Type = 2 (Trades) or 5 (Announcements). If specified, the trades or announcements reported with an equal or higher sequence will be sent.</td></tr>
<tr><td>Request ID</td><td>32</td><td>4</td><td>UInt32</td><td>The value set in this will be echoed back in the corresponding Recovery Response and Recovery Complete message. The system will not validate uniqueness of the set value.</td></tr>
</table>

## 8.5 Administrative Messages (Server – Initiated)

### 8.5.1 Heartbeat

A Unit Header with a Message Count of zero will be used by the server as a Heartbeat message. Such a message will never increment the sequence number of the Real time channel. However, the next expected sequence number will be included in the Sequence Number to enable recipients to detect gaps on the Real time channel.

### 8.5.2 Login Response

<table>
<tr><th>Field</th><th>Offset</th><th>Length</th><th>Type</th><th>Description</th></tr>
<tr><td>Length</td><td>0</td><td>2</td><td>UInt16</td><td>Length of message including this field.</td></tr>
<tr><td>Message Type</td><td>2</td><td>1</td><td>Byte</td><td>Hex Meaning<br>0x02 Login Response</td></tr>
<tr><td>Status</td><td>3</td><td>1</td><td>Byte</td><td>Status of the login request.<br>Value Meaning<br>A Login Accepted<br>a UserID Inactive/Locked<br>b Login Limit Reached<br>c Service Unavailable<br>d Maximum Connections Limit Reached<br>f Invalid UserID, Password or IP Address<br>e Failed (Other)</td></tr>
</table>

### 8.5.3 Replay Response

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** / **Meaning**<br>0x04 / Replay Response |
| First Message | 3 | 4 | UInt32 | Sequence number of the first message in range to be retransmitted. This will be zero if Status is not "A". |
| Count | 7 | 4 | UInt32 | Number of messages to be resent. This will be zero if Status is not "A". |
| Status | 11 | 1 | Byte | **Value** / **Meaning**<br>A / Request Accepted<br>D / Request Limit Reached<br>O / Out of Range<br>U / Replay Unavailable<br>c / Concurrent Limit Reached<br>e / Failed (Other) |
| Request ID | 12 | 4 | UInt32 | Will include the value set as Request ID in the Replay Request message. |

### 8.5.4 Recovery Response

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |

| | | | | |
|---|---|---|---|---|
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x82 Recovery Response |
| Sequence Number | 3 | 4 | UInt32 | Sequence number on the real time channel with which the recovery or recovery is synchronized when Recovery Type = 1,2, 3, 4 and 5. This will be zero for other Recovery Types or if Status is not "A". |
| Count | 7 | 4 | UInt32 | Number of messages to follow. This will be zero if Status is not "A". |
| Status | 11 | 1 | Byte | **Value** **Meaning**<br>A Request Accepted<br>O Out of Range<br>a Group or Symbol Not Specified or Invalid<br>b Request Limit Reached<br>c Concurrent Limit Reached<br>d Invalid Recovery Type or Request Level<br>e Failed (Other) |
| Request ID | 12 | 4 | UInt32 | Will include the value set as Request ID in the Recovery Request message. |

### 8.5.5 Replay and Recovery Complete

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x83 Replay and Recovery Complete |
| Request ID | 3 | 4 | UInt32 | Will include the value set as Request ID in the Recovery Request message. |
| Trading Status | 7 | 1 | Byte | Current Trading status of the Instrument. Populated only when the message is sent at the end of individual order book recoveries.<br>Please refer to the Additional Field Values section of this document for valid values. |

## 8.6 Application Messages

### 8.6.1 System Event

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x53 System Event |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |

| | | | | |
|---|---|---|---|---|
| Event Code | 11 | 1 | Byte | **Value** **Meaning**<br>C End of Day<br>O Start of Day |
| Source Venue | 12 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |

### 8.6.2 Instrument Directory

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x70 Instrument Directory |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Symbol | 11 | 14 | Alpha | Symbol of the instrument. |
| CUSIP | 25 | 9 | Alpha | CUSIP code of the instrument. |
| Allowed Book Types | 34 | 1 | Bit Field | Defines the order book types that are allowed for the instrument. Each designated bit represents a book type. 0 means not allowed and 1 means allowed. |

<table>
<tr><td></td><td></td><td></td><td></td><td><table>
<tr><th>Bit</th><th>Name</th></tr>
<tr><td>0</td><td>Reserved</td></tr>
<tr><td>3</td><td>NEO-L</td></tr>
<tr><td>4</td><td>NEO-N</td></tr>
<tr><td>5</td><td>NEO-D</td></tr>
<tr><td>6</td><td>Cross</td></tr>
<tr><td>7</td><td>SST</td></tr>
</table></td></tr>
<tr><td>Source Venue</td><td>35</td><td>2</td><td>UInt16</td><td>Venue from which market data is received for the instrument. This will be always set to 1 for NEO.</td></tr>
<tr><td>Segment</td><td>37</td><td>6</td><td>Alpha</td><td>Segment the instrument is assigned to.<table>
<tr><th>Value</th><th>Meaning</th></tr>
<tr><td>NEO</td><td>NEO Listed securities</td></tr>
<tr><td>TSX</td><td>TSX Listed securities</td></tr>
<tr><td>TSXV</td><td>TSXV Listed secuities</td></tr>
<tr><td>CSE</td><td>CSE Listed securities</td></tr>
<tr><td>NEOC</td><td>NEO Connect securities</td></tr>
<tr><td>FOTS</td><td>Foreign Other Traded Securities</td></tr>
</table></td></tr>
<tr><td>Currency</td><td>43</td><td>3</td><td>Alpha</td><td>The possible values will be the ISO 4217 codes for currency.</td></tr>
</table>

<table>
<tr><td>Lot Size</td><td>46</td><td>8</td><td>Size</td><td>Indicates the minimum quantity/nominal value tradable on the market for a security.</td></tr>
<tr><td>Full Name</td><td>54</td><td>120</td><td>Alpha</td><td>The unique long name assigned to the instrument.</td></tr>
<tr><td>Active Market Maker</td><td>174</td><td>11</td><td>Alpha</td><td>The active market maker who will be carrying our market maker obligations for the instrument.</td></tr>
<tr><td>Liquidity Tier</td><td>185</td><td>12</td><td>Alpha</td><td>The market tier to which the instrument is assigned.</td></tr>
<tr><td>Market Maker NEO-L Size Requirement</td><td>197</td><td>8</td><td>Size</td><td>The market maker size requirement for NEO-L.</td></tr>
<tr><td>Market Maker NEO-N Size Requirement</td><td>205</td><td>8</td><td>Size</td><td>The market maker size requirement for NEO-N.</td></tr>
<tr><td>Security Type</td><td>213</td><td>1</td><td>UInt8</td><td>The security type of the instrument.<br><br>Value — Meaning<br>1 — Equity<br>2 — Debt<br>52 — Platform Traded Funds (PTF)</td></tr>
<tr><td>Security Subtype</td><td>214</td><td>1</td><td>UInt8</td><td>Security sub type of the instrument.</td></tr>
</table>

| | | | | | |
|---|---|---|---|---|---|
| | | | | **Value** | **Meaning** |
| | | | | 0 | Value Not Set |
| | | | | 52 | Common Shares |
| | | | | 53 | Preferred Shares |
| | | | | 54 | Corporate Bonds and Debentures |
| | | | | 55 | Exchange Traded Funds |
| | | | | 56 | Rights |
| | | | | 57 | Warrants |
| | | | | 58 | Units (Trust) |
| | | | | 59 | Index Linked Notes / ETNs |
| | | | | 60 | Installment Receipts |
| | | | | 61 | Convertible Bonds and Warrants |
| | | | | 62 | When Issued & When Distributed |
| | | | | 63 | Shares of Beneficial Interest (SBI) |
| | | | | 64 | Uncapped Capital Protection |
| | | | | 65 | Exchangeable Certificates |
| | | | | 66 | Capped Capital Protection |

| | | | | | |
|---|---|---|---|---|---|
| | | | | 67 | Capital Protection w/ Knock-Out |
| | | | | 68 | Capital Protection w/ Coupon |
| | | | | 69 | Discount Certificates |
| | | | | 70 | Barrier Discount Certificates |
| | | | | 71 | Reverse Convertibles |
| | | | | 72 | Barrier Reverse Convertibles |
| | | | | 73 | Capped Outperformance Certificates |
| | | | | 74 | Capped Bonus Certificates |
| | | | | 75 | Express Certificates |
| | | | | 76 | Tracker Certificates |
| | | | | 77 | Outperformance Certificates |
| | | | | 78 | Bonus Certificates |
| | | | | 79 | Outperformance Bonus Certificates |
| | | | | 80 | Twin-Win Certificates |
| | | | | 81 | Structured Products |

| | | | | |
|---|---|---|---|---|
| Odd Lot Allowed | 215 | 1 | UInt8 | **Value** **Meaning**<br>0 Not allowed<br>1 Allowed |
| NEO-L Previous Close | 216 | 8 | Price | The previous day's closing price of NEO-L. |
| NEO-N Previous Close | 224 | 8 | Price | The previous day's closing price of NEO-N. |

### 8.6.3 Instrument Status

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x48 Instrument Status |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Symbol | 11 | 14 | Alpha | Symbol of the instrument. |
| Source venue | 25 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |
| Trading Status | 27 | 1 | Byte | Please refer to the Additional Field Values section of this document for valid values. |

<table>
<tr><td>Session Change Reason</td><td>28</td><td>1</td><td>UInt8</td><td><table><tr><th>Value</th><th>Meaning</th></tr><tr><td>0</td><td>Scheduled Transition</td></tr><tr><td>1</td><td>Extended by Market Ops</td></tr><tr><td>2</td><td>Shortened by Market Ops</td></tr><tr><td>3</td><td>Market Order Imbalance</td></tr><tr><td>4</td><td>Price Outside Range</td></tr><tr><td>5</td><td>Circuit Breaker Tripped</td></tr><tr><td>9</td><td>Unavailable</td></tr><tr><td>50</td><td>SSCB</td></tr><tr><td>51</td><td>SSCB Extended</td></tr><tr><td>52</td><td>Market Maker Delayed Opening / Re-Opening</td></tr><tr><td>53</td><td>Auction Execution</td></tr></table></td></tr>
<tr><td>New End Time</td><td>29</td><td>6</td><td>Time</td><td>New time the session will end. The field will contain only spaces if Session Change Reason is “0” or “9”. New End Time will be in terms of the local market time (i.e. not UTC).</td></tr>
<tr><td>Order Book Type</td><td>35</td><td>1</td><td>UInt8</td><td>Please refer to the Additional Field Values section of this document for valid values.</td></tr>
</table>

### 8.6.4 Add Order Incremental

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** / **Meaning**<br>0x46 / Add Order Incremental |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Order ID | 11 | 8 | UInt64 | Unique identifier of the order. Base 62 Encoded string for source venue AEQ converted to the Type specified. |
| Side | 19 | 1 | Byte | **Value** / **Meaning**<br>B / Buy Order<br>S / Sell Order |
| Size | 20 | 8 | Size | Displayed Size of the order. |
| Symbol | 28 | 14 | Alpha | Symbol of the instrument. |
| Price | 42 | 8 | Price | Limit price of the order. |
| Source venue | 50 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |
| Order Book Type | 52 | 1 | UInt8 | Please refer to the Additional Field Values section of this document for valid values. |
| Broker | 53 | 3 | Alpha | Identity of trading participant that submitted the order. |

| | | | | |
|---|---|---|---|---|
| Order Type | 56 | 1 | UInt8 | **Value** **Meaning**<br>3 Limit<br>4 Market |
| Settlement Type | 57 | 1 | UInt8 | Settlement terms of the order if non-standard. If standard, this field will be null.<br>**Value** **Meaning**<br>1 Cash<br>2 Next Day<br>6 Future<br>11 Non-Net |
| Settlement Date | 58 | 8 | Date | The date the order would settle. |

### 8.6.5 Add Order Incremental MBP

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x51 Add Order Incremental MBP |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Side | 11 | 1 | Byte | **Value** **Meaning**<br>B Buy Order<br>S Sell Order |
| Size | 12 | 8 | Size | Displayed Size of a price point. |

| | | | | |
|---|---|---|---|---|
| Symbol | 20 | 14 | Alpha | Symbol of the instrument. |
| Price | 34 | 8 | Price | Value of the price point. |
| Source venue | 42 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |
| Order Book Type | 44 | 1 | UInt8 | Please refer to the Additional Field Values section of this document for valid values. |

### 8.6.6 Delete Order

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x44 Delete Order |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Order ID | 11 | 8 | UInt64 | Unique identifier of the order. Base 62 Encoded string for source venue AEQ converted to the Type specified. |
| Symbol | 19 | 14 | Alpha | Symbol of the instrument. |
| Order Book Type | 33 | 1 | UInt8 | Please refer to the Additional Field Values section of this document for valid values. |
| Source venue | 34 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |

### 8.6.7 Delete Order MBP

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x54 Delete Order |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Symbol | 11 | 14 | Alpha | Symbol of the instrument. |
| Side | 25 | 1 | Byte | **Value** **Meaning**<br>B Buy<br>S Sell |
| Order Book Type | 26 | 1 | UInt8 | Please refer to the Additional Field Values section of this document for valid values. |
| Source venue | 27 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |
| Previous Price | 29 | 8 | Price | Price point that was deleted from the book. |

### 8.6.8 Modify Order

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x55 Modify Order |

<table>
<tr><td>Timestamp</td><td>3</td><td>8</td><td>UDT</td><td colspan="3">Time the message was generated</td></tr>
<tr><td>Order ID</td><td>11</td><td>8</td><td>UInt64</td><td colspan="3">Unique identifier of the order. Base 62 Encoded string for source venue AEQ converted to the Type specified.</td></tr>
<tr><td>Symbol</td><td>19</td><td>14</td><td>Alpha</td><td colspan="3">Symbol of the instrument.</td></tr>
<tr><td rowspan="4">Flags</td><td rowspan="4">33</td><td rowspan="4">1</td><td rowspan="4">Bit Field</td><td colspan="3"></td></tr>
<tr><td>Bit</td><td>Name</td><td>Meaning</td></tr>
<tr><td rowspan="2">0</td><td rowspan="2">Priority Flag</td><td>0: Priority Lost</td></tr>
<tr><td>1: Priority Retained</td></tr>
<tr><td>New Quantity</td><td>34</td><td>8</td><td>Size</td><td colspan="3">New displayed quantity of the order.</td></tr>
<tr><td>New Price</td><td>42</td><td>8</td><td>Price</td><td colspan="3">New price of the order.</td></tr>
<tr><td>Order Book Type</td><td>50</td><td>1</td><td>UInt8</td><td colspan="3">Please refer to the Additional Field Values section of this document for valid values.</td></tr>
<tr><td>Source venue</td><td>51</td><td>2</td><td>UInt16</td><td colspan="3">Venue from which market data is received for the instrument. This will be always set to 1 for NEO.</td></tr>
<tr><td>Settlement Type</td><td>53</td><td>1</td><td>UInt8</td><td colspan="3">Settlement terms of the order if non-standard. If standard, this field will be null.</td></tr>
</table>

| | | | | |
|---|---|---|---|---|
| | | | | **Value** **Meaning**<br>1 Cash<br>2 Next Day<br>6 Future<br>11 Non-Net |
| Settlement Date | 54 | 8 | Date | The date the order would settle. |
| Broker | 62 | 3 | Alpha | Identity of trading participant that submitted the order. |

### 8.6.9 Modify Order MBP

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x56 Modify Order MBP |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Symbol | 11 | 14 | Alpha | Symbol of the instrument. |
| Side | 25 | 1 | Byte | **Value** **Meaning**<br>B Buy<br>S Sell |
| Order Book Type | 26 | 1 | UInt8 | Please refer to the Additional Field Values section of this document for valid values. |
| Source venue | 27 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |

| | | | | |
|---|---|---|---|---|
| New Quantity | 29 | 8 | Size | New displayed quantity of the order. |
| New Price | 37 | 8 | Price | New price point after the modification. |
| Previous Price | 45 | 8 | Price | Previous price point before the modification. |

### 8.6.10 Order Book Clear

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x79 Order Book Clear |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Symbol | 11 | 14 | Alpha | Symbol of the instrument. |
| Order Book Type | 25 | 1 | UInt8 | Please refer to the Additional Field Values section of this document for valid values. |
| Source venue | 26 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |

### 8.6.11 Trade

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |

| | | | | |
|---|---|---|---|---|
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x50 Trade |
| Timestamp | 3 | 8 | UDT | Message dissemination time. |
| Source venue | 11 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |
| Executed Size | 13 | 8 | Size | Size executed. |
| Symbol | 21 | 14 | Alpha | Symbol of the instrument. |
| Price | 35 | 8 | Price | Executed price. |
| Trade ID | 43 | 8 | UInt64 | Unique identifier of the trade. Base 36 or 62 encoded as per venue format. |
| Trade Type | 51 | 1 | UInt8 | **Value** **Meaning**<br>0 Post-Open Trade<br>1 Auction Trade – Bulk<br>2 Auction Trade – Individual<br>9 Trade Cancellation<br>11 Trade Correction |
| Auction Type | 52 | 1 | Byte | The value in this field is only relevant when Trade Type is 1.<br>**Value** **Meaning**<br>C Closing Auction<br>O Opening Auction<br>A Re-opening |
| Buy Attribution | 53 | 3 | Alpha | Broker ID of the buyer of the trade. |

<table>
<tr><td>Sell Attribution</td><td>56</td><td>3</td><td>Alpha</td><td>Broker ID of the seller of the trade.</td></tr>
<tr><td>Order Book Type</td><td>59</td><td>1</td><td>UInt8</td><td>Please refer to the Additional Field Values section of this document for valid values.</td></tr>
<tr><td>Cross Type</td><td>60</td><td>1</td><td>UInt8</td><td>The type of the Cross/BTF Order.<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>5</td><td>Internal Cross</td></tr><tr><td>11</td><td>Basis Cross</td></tr><tr><td>12</td><td>Contingent Cross</td></tr><tr><td>14</td><td>VWAP Cross</td></tr><tr><td>15</td><td>National Cross</td></tr><tr><td>16</td><td>Bypass Cross</td></tr><tr><td>17</td><td>Non-NEO Cross</td></tr><tr><td>18</td><td>Derivative Cross</td></tr></table></td></tr>
<tr><td>Flags</td><td>61</td><td>1</td><td>Bit Field</td><td><table><tr><th>Bit</th><th>Name</th><th>Meaning</th></tr><tr><td>0</td><td>LSP updated</td><td>0: No<br>1: Yes</td></tr><tr><td>1</td><td>Bypass trade</td><td>0: No<br>1: Yes</td></tr><tr><td>2</td><td>Manual Trade Correction</td><td>0: No<br>1: Yes</td></tr></table></td></tr>
<tr><td>Order ID</td><td>62</td><td>8</td><td>UInt64</td><td>The unique identifier of the resting order in the book which got executed. Base 62 Encoded string for source</td></tr>
</table>

| | | | | |
|---|---|---|---|---|
| | | | | venue AEQ converted to the Type specified. This will not be populated for hidden executions, NEO-N trades, Trade Cancellations, and Trade Corrections. |

### 8.6.12 Statistics

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x77 Statistics |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Symbol | 11 | 14 | Alpha | Symbol of the instrument. |
| Source venue | 25 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |
| Volume | 27 | 8 | Size4 | Cumulative volume of all trades for the trading day. |
| VWAP | 35 | 8 | Price4 | Volume weighted average price for the day for all trades. |
| Number of Trades | 43 | 4 | UInt32 | Count of all trades for the day. |
| Turnover | 47 | 8 | Price4 | Turnover of all trades for the day. |

| | | | | |
|---|---|---|---|---|
| Order Book Type | 55 | 1 | UInt8 | Please refer to the Additional Field Values section of this document for valid values. |

### 8.6.13 Statistics Update

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x6a Statistics Update |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Symbol | 11 | 14 | Alpha | Symbol of the instrument. |
| Source venue | 25 | 2 | UInt16 | Venue from which market data is received for the instrument. This will be always set to 1 for NEO. |
| Statistic Type | 27 | 2 | UInt16 | The statistic that is disseminated with this message instance for the various NEO trading books only. |

<table>
<tr><td></td><td></td><td></td><td></td><td><table><tr><th>Value</th><th>Meaning</th></tr><tr><td>1</td><td>Indicative Auction Uncrossing Data</td></tr><tr><td>2</td><td>Official Opening Price</td></tr><tr><td>3</td><td>Official Closing Price</td></tr><tr><td>6</td><td>Trade High</td></tr><tr><td>7</td><td>Trade Low</td></tr><tr><td>8</td><td>52 Wk Trade High All Trades</td></tr><tr><td>9</td><td>52 Wk Trade Low All Trades</td></tr><tr><td>16</td><td>Previous Close</td></tr></table></td></tr>
<tr><td>Stat Price</td><td>29</td><td>8</td><td>Price4</td><td>The value of Price type statistics. Note that If the Opening or Closing Price is cleared manually by the venue, “-1” will be stamped on this field to reflect that the Opening/Closing price was unset.</td></tr>
<tr><td>Auction Type</td><td>37</td><td>1</td><td>Byte</td><td>Populated if the Statistic Type is 1.<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>C</td><td>Closing Auction</td></tr><tr><td>O</td><td>Opening Auction</td></tr><tr><td>A</td><td>Re-opening</td></tr></table></td></tr>
</table>

| | | | | |
|---|---|---|---|---|
| Imbalance Quantity | 38 | 8 | Size | Quantity that is eligible to be matched at the indicative price but will not be matched. |
| Imbalance Direction | 46 | 1 | Byte | **Value** / **Meaning**<br>B – Buy Imbalance<br>N – No Imbalance<br>O – Insufficient Orders for Auction<br>S – Sell Imbalance |
| Opening/ Closing Price Indicator | 47 | 1 | Byte | Populated if the Statistic Type is 2 or 3.<br>Please refer to the Additional Field Values section of this document for valid values. |
| Order Book Type | 48 | 1 | UInt8 | Please refer to the Additional Field Values section of this document for valid values. |

### 8.6.14 Statistics Snapshot

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** / **Meaning**<br>0x6b – Statistics Recovery |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Symbol | 11 | 14 | Alpha | Symbol of the instrument. |
| Source venue | 25 | 2 | UInt16 | Venue from which market data is received for the |

| | | | | |
|---|---|---|---|---|
| | | | | instrument. This will be always set to 1 for NEO. |
| Volume | 27 | 8 | Size4 | Cumulative volume of all trades for the trading day. |
| VWAP | 35 | 8 | Price4 | Volume weighted average price for the day for all trades. |
| Number of Trades | 43 | 4 | UInt32 | Count of all trades for the day. |
| Turnover | 47 | 8 | Price4 | Turnover of all trades for the day. |
| Official Opening Price | 55 | 8 | Price4 | Opening Price for the instrument. Note that If the Opening price is cleared manually by the venue, "-1" will be stamped on this field to reflect that the Opening price was unset. |
| Official Closing Price | 63 | 8 | Price4 | Closing Price for the instrument. Note that If the Closing Price is cleared manually by the venue, "-1" will be stamped on this field to reflect that the Closing price was unset. |
| Trade High | 71 | 8 | Price4 | Day's high price of all trades. |
| Trade Low | 79 | 8 | Price4 | Day's low price of all trades. |
| 52 Wk Trade High | 87 | 8 | Price4 | 52 Week high price of all trades. |
| 52 Wk Trade Low | 95 | 8 | Price4 | 52 Week low price of all trades. |

<table>
<tr><td>Opening Price Indicator</td><td>103</td><td>1</td><td>Byte</td><td>Please refer to Description in Statistics Update message for valid values.</td></tr>
<tr><td>Closing Price Indicator</td><td>104</td><td>1</td><td>Byte</td><td>Please refer to Description in Statistics Update message for valid values.</td></tr>
<tr><td>IAU Price</td><td>105</td><td>8</td><td>Price4</td><td>Contains the last reported Indicative Auction Crossing Price.</td></tr>
<tr><td>Imbalance Quantity</td><td>113</td><td>8</td><td>Size</td><td>Quantity that was eligible to be matched at the indicative price but was not to be matched at the last indicative price.</td></tr>
<tr><td>Imbalance Direction</td><td>121</td><td>1</td><td>Byte</td><td><table><tr><th>Value</th><th>Meaning</th></tr><tr><td>B</td><td>Buy Imbalance</td></tr><tr><td>N</td><td>No Imbalance</td></tr><tr><td>O</td><td>Insufficient Orders for Auction</td></tr><tr><td>S</td><td>Sell Imbalance</td></tr></table></td></tr>
<tr><td>Auction Type</td><td>122</td><td>1</td><td>Byte</td><td><table><tr><th>Value</th><th>Meaning</th></tr><tr><td>C</td><td>Closing Auction</td></tr><tr><td>O</td><td>Opening Auction</td></tr><tr><td>A</td><td>Re-opening</td></tr></table></td></tr>
<tr><td>Order Book Type</td><td>123</td><td>1</td><td>UInt8</td><td>Please refer to the Additional Field Values section of this document for valid values.</td></tr>
<tr><td>Previous Close</td><td>124</td><td>8</td><td>Price4</td><td>Previous day’s closing price.</td></tr>
</table>

### 8.6.15 Announcements

| Field | Offset | Length | Type | Description |
|---|---|---|---|---|
| Length | 0 | 2 | UInt16 | Length of message including this field. |
| Message Type | 2 | 1 | Byte | **Hex** **Meaning**<br>0x75 Announcements |
| Timestamp | 3 | 8 | UDT | Time the message was generated. |
| Urgency | 11 | 1 | Byte | **Value** **Meaning**<br>0 Regular<br>1 High Priority<br>2 Low Priority |
| Headline | 12 | 50 | Alpha | Headline or subject of the announcement. |
| Text | 62 | 120 | Alpha | Text of the announcement. |
| Symbols | 182 | 30 | Alpha | Pipe separated list of symbols of the securities for which the announcement has been sent. |
| Underlying/Market | 212 | 30 | Alpha | Pipe separated list of symbols of the underlying securities/markets for which the announcement has been sent. |

# 9 Additional Field Values

### 9.1.1 Book Type

| Value | Meaning |
|---|---|
| 0 | All Books |
| 3 | NEO-L |
| 4 | NEO-N |
| 5 | NEO-D |
| 6 | Cross |
| 7 | SST |

N.B. A value of "0", All Books, is only valid for recovery requests. A consolidated statistics feed will not be published using the value of "0", All Books.

### 9.1.2 Segment

| Value | Meaning |
|---|---|
| NEO | NEO Listed Securities |
| TSX | TSX Listed Securities |
| TSXV | TSXV Listed Secuities |
| CSE | CSE Listed Securities |
| NEOC | NEO Connect Securities |
| FOTS | Foreign Other Traded Securities |

### 9.1.3 Trading Status

Source Venue – AEQ:

| Value | Meaning |
|---|---|
| H | Full Halt<br>During a full halt, new orders cannot be entered and existing orders cannot be amended. Orders can only be cancelled. |
| J | Full Halt – Matching partition Suspended<br>During a full halt, new orders cannot be entered and existing orders cannot be amended. Orders can only be cancelled. |

| | |
|---|---|
| K | Full Halt – System Suspended<br>During a full halt, new orders cannot be entered and existing orders cannot be amended. Orders can only be cancelled. |
| T | Post-Open<br>During post-open, the normal order entry and trading rules of a specific book apply. |
| a | Pre-Open<br>Applicable for NEO-L, orders can be entered, however no trades will occur until the system transitions to post-open. |
| b | Closed<br>During a Closed state, new orders cannot be entered, existing orders cannot be amended or cancelled. |
| c | Market Closed<br>The market is closed for the day. No new messages will be published for the given trading day. |
| d | Closing Auction Call<br>Applicable to NEO-L, orders can be entered, however no trades will occur until the system executes the closing auction. |
| e | No Matching Halt – Auction<br>During a No Matching Halt – Auction, new orders can be entered, existing orders can be amended or cancelled. Trades will only occur at the end of the specified session. Statistics will be published only when a potential auction trade can be executed at the end of the session. |
| I | No Matching Halt<br>During a No Matching Halt, new orders can be entered, existing orders can be amended or cancelled. The symbol will first transition to No Matching Halt – Auction in order to enter a Post-Open state. No statistics will be published in this session. |
| 2 | Suspended<br>A symbol can be suspended for various business reasons. Instrument directory messages will still be published, however the symbol will not be valid for trading on a given day. When a symbol enters a suspended session, all resting orders will be immediately cancelled. |
| y | Pre-Trading |

| | |
|---|---|
| | During Pre-Trading session, no new orders can be entered and existing orders can not be amended or cancelled. |
| u | Extended Trading<br>During Extended Trading, trades can only occur at the Last Sale Price for a NEO listed security within NEO-L. |
| r | Pre Close<br>During Pre Close session, no new orders can be entered and exisiting orders can not be amended or cancelled for any of the PTF securities. |
| o | Pre-Open<br>Applicable to the NEO Connect, orders can not be entered, however Trade messages and Statistics will be published only when an auction trade can be executed at the start of this session.<br><br>Applicable to NEO-D. NEO-D orders can be entered during this session but trading will not commence until Post-Open. |
| 9 | MOC Imbalance |

### 9.1.4 Opening / Closing Price Indicator

This can be configured to support any indicator:

| Value | Meaning |
|---|---|
| A | Auction Trade |
| B | Post-Open Trade |
| C | Mid-Point |
| D | Last Post-Open Trade |
| E | Last Auction |
| F | Manual |
| H | VWAP |
| I | Previous Close |
| T | Theoretical Price |
| U | Best Bid |
| V | Best Offer |
| W | None |

| | |
|---|---|
| X | VWAP of Last n Trades |
| Y | Reference Price |
| Z | Price Unavailable |

### 9.1.5 Gateway ID

NOTE: Instance 1 of each Gateway will be denoted by an Odd number and the instance 2 will be denoted by an even number.

| Value | Process Name |
|---|---|
| 1 | <Specify the list of process names used for the services offered as channels><br>example: TOBGateway:1:1:TOBMarketDataGateway:1 |

### 9.1.6 Source Venue

| Value | Meaning |
|---|---|
| 1 | AEQ (Aequitas NEO Exchange) |

# 10 Appendix I

## 10.1 Conversion of Negative Values in Price Fields

Negative values are never expected to be sent on the NITCH feed in the normal course of events, however one may be sent in an unlikely error situation.

### 10.1.1 Encoding Negative Values in Price Fields

Decimal value = -1

Decimal value with eight implied decimal places = -100000000

Remove sign bit = 100000000

Convert to binary = 00000000 00000000 00000000 00000000 00000101 11110101

11100001 00000000

Add sign bit = 10000000 00000000 00000000 00000000 00000101 11110101

11100001 00000000

Hex value = 80 00 00 00 05 f5 e1 00

Hex value converted to Little endian = 00 e1 f5 05 00 00 00 80

### 10.1.2 Decoding Negative Values in Price Fields

Received bytes in hex = 00 e1 f5 05 00 00 00 80

Change the byte order to big endian = 80 00 00 00 05 f5 e1 00

Convert to binary = 10000000 00000000 00000000 00000000 00000101

11110101 11100001 00000000

Most significant bit is set. Therefore this is a negative value.

Remove the sign bit = 00000000 00000000 00000000 00000000 00000101

11110101 11100001 00000000

Convert to decimal = 100000000

Add sign to decimal = -100000000

Mark eight implied decimal places -1.00000000

# 11 Appendix II: Conversion of Order and Trade Identifiers

## 11.1 Order ID

### 11.1.1 Order ID format (in ASCII)

The composition of the Order IDs assigned by the matching system and received via FIX is given below. This Order ID is specified as ASCII printable characters and will not exceed 12 bytes.

12 bytes

| 0-9, A-Z, a-z |
|---|

Base 62 encoded order id

The FIX Tag (Order ID (37)) can be directly converted to MITCH Order ID by using base 62 decoding.

### 11.1.2 Conversion Logic

The Order IDs assigned by the market data feed are specified as Little-Endian encoded 64 bit unsigned integers. The logic used to convert the Order ID received via FIX into the format used by the market data feed is as follows:

- Remove the left most byte (i.e. O of Order ID)
- Convert the rest of the digits to decimal using the base 62 dictionary
- Convert the decimal values to binary

**Example**

| **OrderID received via FIX**<br>(ASCII base 62 characters) | O06WoCOv0Lwq |
|---|---|

**Step 1:** Remove the left most Byte "O" → 06WoCOv0Lwq

**Step 2:** Convert the rest of the digits to decimal using the base 62 dictionary. Please refer to the base 62 conversion table provided in Section 11.3.

| Order ID (ASCII Character) | Decimal Value | Base 62^x | Value | Multiplied Decimal Value |
|---|---|---|---|---|
| q | 52 | 62^0 | 1 | 52 |
| w | 58 | 62^1 | 62 | 3596 |
| L | 21 | 62^2 | 3,844 | 80724 |
| 0 | 0 | 62^3 | 238,328 | 0 |
| v | 57 | 62^4 | 14,776,336 | 842251152 |
| O | 24 | 62^5 | 916,132,832 | 21987187968 |
| C | 12 | 62^6 | 56,800,235,584 | 681602827008 |
| o | 50 | 62^7 | 3,521,614,606,208 | 176080730310400 |
| W | 32 | 62^8 | 218,340,105,584,896 | 6986883378716672 |
| 6 | 6 | 62^9 | 13,537,086,546,263,552 | 81222519277581312 |
| 0 | 0 | 62^10 | 839,299,365,868,340,224 | 0 |
| **Total** | | | | **88386187818958884** |

**Step 3:** Convert the decimal values to binary.

| | |
|---|---|
| **OrderID on Market Data Feed** | 10011101000000010110001010000000000000000000000000100100 |

## 11.2 Trade ID

### 11.2.1 Trade ID format (in ASCII)

The composition of the Trade IDs received via FIX is given below. This Trade ID is specified as ASCII printable characters.

11 bytes

| 0-9, A-Z, a-z |
|---|

Base 62 encoded Trade ID

The FIX Trade ID (ExecID (17)) can be directly converted to MITCH Trade ID by using base 62 decoding.

### 11.2.2 Conversion Logic

The Trade IDs assigned by the market data feed are specified as Little-Endian encoded 64 bit unsigned integers. The logic used to convert the Trade ID received via FIX into the format used by the market data feed is as follows:

- Remove the left most byte (i.e. T of Trade ID) and the right most byte ( i.e. S or B denoting the side of the trade)
- Convert the rest of the digits to decimal using the base 62 dictionary
- Convert the decimal values to binary

**Example**

| **Trade ID received via FIX** (ASCII base 62 characters) | T1aNhwVdkvB |
|---|---|

**Step 1:** Remove the left most Byte "T" and the right most byte "B"→ 1aNhwVdkv

**Step 2:** Convert the rest of the digits to decimal using the base 62 dictionary. Please refer to the base 62 conversion table provided in Section 11.3.

| **Trade ID (ASCII Character)** | **Decimal Value** | **Base 62^x** | **Value** | **Multiplied Decimal Value** |
|---|---|---|---|---|
| v | 57 | 62^0 | 1 | 57 |
| k | 46 | 62^1 | 62 | 2852 |
| d | 39 | 62^2 | 3,844 | 149916 |
| V | 31 | 62^3 | 238,328 | 7388168 |
| w | 58 | 62^4 | 14,776,336 | 857027488 |
| h | 43 | 62^5 | 916,132,832 | 39393711776 |
| N | 23 | 62^6 | 56,800,235,584 | 1306405418432 |
| a | 36 | 62^7 | 3,521,614,606,208 | 126778125823488 |
| 1 | 1 | 62^8 | 218,340,105,584,896 | 218340105584896 |
| **Total** | | | | **346464895107073** |

**Step 3:** Convert the decimal values to binary.

| | |
|---|---|
| **Trade ID on Market Data Feed** | 10011101100011011101001010000000000000000000000001 |

## 11.3 Base 62 Mapping Table

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 20 | K | 40 | e | 60 | y |
| 1 | 1 | 21 | L | 41 | f | 61 | z |
| 2 | 2 | 22 | M | 42 | g | | |
| 3 | 3 | 23 | N | 43 | h | | |
| 4 | 4 | 24 | O | 44 | i | | |
| 5 | 5 | 25 | P | 45 | j | | |
| 6 | 6 | 26 | Q | 46 | k | | |
| 7 | 7 | 27 | R | 47 | l | | |
| 8 | 8 | 28 | S | 48 | m | | |
| 9 | 9 | 29 | T | 49 | n | | |
| 10 | A | 30 | U | 50 | o | | |
| 11 | B | 31 | V | 51 | p | | |
| 12 | C | 32 | W | 52 | q | | |
| 13 | D | 33 | X | 53 | r | | |
| 14 | E | 34 | Y | 54 | s | | |
| 15 | F | 35 | Z | 55 | t | | |
| 16 | G | 36 | a | 56 | u | | |
| 17 | H | 37 | b | 57 | v | | |
| 18 | I | 38 | c | 58 | w | | |
| 19 | J | 39 | d | 59 | x | | |