

# Turquoise Equities

---

## TQ201 - FIX 5.0 Trading Gateway

Issue 2.7

27 September 2013



**London**

Stock Exchange Group



# Contents

---

## 1.0 Introduction – TQ201 Trading Gateway (FIX 5.0) 4

1.1 Purpose	4
1.2 Readership	4
1.3 Document Series	4
1.4 Document History	5
1.5 Enquires	7

---

## 2.0 Service Description 8

2.1 Order Handling	8
2.1.3 Order Status	16
2.2 Liquidity Pools	22
2.3 Symbology Schemes	22
2.4 Market Operations	24
2.5 Timestamps and Dates	24
2.6 Party Identification	25
2.7 Information for Billing	26
2.8 Order Capacity	26
2.9 Repeating Groups (Components/Component Block)	26
2.10 Auto Cancel on Disconnect	27
2.11 Generating Reject Messages	27

---

## 3.0 Connectivity 28

3.1 ComplIDs	28
3.2 Production IP Address and Ports	28
3.3 Failover and Recovery	28
3.4 Message Rate Throttling	28

---

## 4.0 FIX Connections and Sessions 29

4.1 Establishing a FIX Connection	29
4.2 Maintaining A FIX Session	30
4.3 Terminating a FIX Session	31
4.4 Re-Establishing a FIX Session	31
4.5 Dormant Account Policy	32

---

## 5.0 Recovery 32

5.1 Resend Requests	32
5.2 Possible Duplicates	32
5.3 Possible Resends	33
5.4 Transmission of Missed Messages	33
5.5 Resending Previous Execution Reports	33

---

## 6.0 Message Formats 34

6.1 Supported Message Types	34
6.2 Message Header and Trailer	36
6.3 Administrative Messages	38
6.4 Application Messages: Order Handling	43
6.5 Application Messages: Others	65
6.6 Components of Application Messages	66

---

## 7.0 Reject Codes 67

7.1 Order Handling	67
7.2 Others	68

---

## 8.0 Appendix A 70

8.1 Order routing logic if RoutingInst (9303) not specified	70
---	----

---

## 9.0 Appendix B 73

9.1 Converting FIX TradeMatchID (880) toITCH TradeMatchID	73
---	----

---

## 10.0 Appendix C 75

10.1 Error & Reject Messages	75
------------------------------	----

---

## 1.0 Introduction – TQ201 Trading Gateway (FIX 5.0)

The Financial Information Exchange (FIX) protocol enables access to Turquoise using a messaging standard developed for real-time electronic exchange of security transactions.

FIX enables access to the trading services and security information within Turquoise. This specification describes a conceptual overview of the FIX 5.0 SP2 protocol as well as providing technical guidance on adopting FIX 5.0 SP2 to connect to Turquoise.

The interface is a point-to-point service based on the technology and industry standards TCP/IP, FIXT and FIX. The session and application event models and messages are based on versions 1.1 and 5.0 (Service Pack 2) of the FIXT and FIX protocols respectively.

FIX specification: <http://www.fixprotocol.org>

### 1.1 Purpose

The purpose of this document is to provide a technical description of the FIX trading gateway available at Turquoise.

### 1.2 Readership

This document outlines how to connect to the FIX trading gateway and the detailed message types and fields used.

When read in conjunction with the other technical specifications, it is intended that these documents provide all of the details directly connected Turquoise participants require to develop to the trading services.

This document is particularly relevant to technical staff within the MTF's member firms.

### 1.3 Document Series

This document is part of series of technical documents providing a holistic view of full trading and information services available from Turquoise.

For reference the full range of technical documents is outlined below:

- TQ102 – Guide to Connectivity
- TQ103 – Trading Technical Parameters
- **TQ201 - Trading Gateway (FIX 5.0) Specification (this document)**
- TQ202 - Post Trade Gateway (FIX 5.0) Specification
- TQ203 – Drop Copy Gateway (FIX 5.0) Specification

- TQ301 – Trading Gateway (Native) Specification
- TQ401 – ITCH Level-2 Market Data Specification
- TQ501 – Guide to Reference Data Services
- TQ502 – Guide to Purchase and Sales file
- TQ601 – Guide to Certification
- TQ602 – TQ Certification Report

#### 1.4 Document History

This document has been through the follow iterations:

Issue	Date	Description
R1 1.0	17 March 2010	First issue of this document published.
R2 1.0	24 May 2010	First issue of CDS release 2 document published.
R2 1.1	16 June 2010	Issue 1.1, Release 2 published
R2 1.2	17 June 2010	Issue 1.2, Release 2 published
R2.1 1.0	09 July 2010	First issue of CDS release 2.1 document published.
R2.1 1.3	13 August 2010	Issue 1.3, Release 2.1 published
R2.1 1.4	16 September 2010	Issue 1.4, Release 2.1 published
1.5	12 November 2010	Issue 1.5 published
1.6	18 February 2011	<p>Section 4.1 TCP/IP disconnection if additional client messages sent before exchange of Logon messages.</p> <p>Section 2.1.2.1 Priority of OrderID over OrigClOrdID</p> <p>Section 2.1.1 Definition of an Iceberg</p> <p>Appendix A Removed footnote</p> <p>Section 6.4.1 Changed description of ExpireDate and ExpireTime</p>

1.7	7 April 2011	Section 6.4.7 Added Partition 3
1.8	11 May 2011	<p>Section 2.1.1 – Clarity added for Iceberg Orders.</p> <p>Section 2.1.2.2 – Update added for Mass Cancellations</p> <p>Section 2.1.5.3 – Clarity added</p> <p>Section 2.1.5.5 – Clarity added to accommodate multiple partitions.</p> <p>Section 2.10 – Clarity added</p> <p>Section 4.1 – Updated section for establishing a FIX connection.</p> <p>Section 4.4 – Updated section for re-establishing a FIX Session.</p> <p>Section 6.4.7 – Add reference to 3<sup>rd</sup> Partition</p> <p>Section 7.2.1, 7.2.2 – Updated error code lists</p>
1.9	6 July 2011	<p>Updated sections to 4.1 and 4.4 to remove the Test Request message sent at Logon. The Test Request message at Logon will be re-introduced in a later release.</p> <p>Update to 10.1 Error &amp; Reject Messages</p>
2.0	31 October 2011	Support for clearing interoperability.
2.1	4 January 2012	<p>Added section 2.8 – Order Capacity</p> <p>Section 6.4.1 – Added CFD Give Up capacity</p>
2.2	27 April 2012	<p>Section 2.1.1 – Updated details of minimum fill functionality and continuous only orders</p> <p>Section 2.1.2.3 – added attributes of an order that can be amended</p> <p>Section 2.2 – Change to matching priority in Dark Midpoint book</p> <p>Added Section 4.4.1.3 – Dormant Account Policy</p> <p>Section 6.4.1, 6.4.4, 6.4.5 - Added exec instruction</p> <p>Section 6.4.5 – Added PegPriceType</p>

		Section 10.1 – added additional error messages
2.3	04 July 2012	Appended section 3.4 Message Rate Throttling
2.4	31 August 2012	Section 2.1.1 - Added details of Passive Only Order type  Section 2.1.2.3 – Added Passive Only Order to amendable attributes  Section 2.10 – Clarified generation of rejects  Section 6.4.1, 6.4.4 – Added PassiveOnlyOrder field  Section 6.4.5 – Added PassiveOnlyOrder and PriceDifferential fields. Added TradeLiquidityIndicator enum of 'C' for Periodic Uncross.
2.5	3 October 2012	6.4.1, 6.4.4 – Clarified PassiveOnlyOrder only supported for Integrated order book  6.4.5 – Removed references to 'dark'
2.6	13 February 2013	Update contact details
2.7	20 September 2013	The following sections have been updated; 1.3, 2.1.1; 2.1.2.1; 2.1.2.3; 2.1.2.5; 2.1.2.6; 2.1.4; 2.1.5.7; 2.1.5.8; 2.3.3; 5.1; 6.4.1; 6.4.2; 6.4.4; 6.4.5; 7.2.1; 7.2.2

In subsequent issues, where amendments have been made to the previous version, these changes will be identified using a series of side bars as illustrated opposite.

## 1.5 Enquires

Contact Technical Account Management at Turquoise for any functional queries regarding the services outlined in this document. Technical Account Management can be contacted Monday to Friday between 07:30UK and 18:00UK:

- Telephone: +44 (0)20 7797 3939
- Email: [londontam@lseg.com](mailto:londontam@lseg.com)

---

## 2.0 Service Description

### 2.1 Order Handling

#### 2.1.1 Order Types

Clients may submit the order types outlined below via the [New Order Single](#) message.

Order Type	Description	Relevant FIX Tags
Market	<p>Market orders will execute at the best available prices in the Integrated book and any remainder will be cancelled. Orders will be subject to Price Band and Maximum Order Value validations.</p> <p>Market orders in the Dark Midpoint order book will execute at the PBBO midpoint.</p>	OrderType (40) = 1
Limit	<p>Limit orders will execute at or better than the specified price in the Integrated book. Orders will be subject to Price Band and Maximum Order Value validations.</p> <p>Limit orders will execute in the Dark Midpoint order book at the PBBO midpoint only if the limit price is equal to or better than the midpoint. Orders will be subject to Maximum Order Value validations.</p>	OrderType (40) = 2  Price (44)
Pegged	An order that will execute at the Mid-point of the Visible Best Bid and Offer. These will always be Hidden Orders. Pegged orders are not applicable to the Lit Book.	OrderType (40) = P
Iceberg	An order that contains a disclosed quantity which will be the maximum quantity displayed on the order book. Once the displayed quantity is reduced to zero, it will be replenished by the lower of the disclosed quantity and the remainder. The displayed quantity will be replenished by the lower of the disclosed quantity and the remainder once the quantity is reduced to zero in case of a passive order or immediately if executed on aggression.	DisplayQty (1138)  OrderQty (38)
Hidden	An order that meets MiFID large in scale requirements that is not displayed in the order book. These orders will receive the lowest priority within a price point when executing in the Integrated book.	DisplayQty (1138) = 0



Midpoint Pegged (Dark)	An order that will execute at the midpoint of the Primary Best Bid and Offer. This order type can behave as a Limit or Market order depending on order entry parameters.	OrderType (40) = P
Minimum Fill	<p>In the Integrated order book, MAQ (Minimum Acceptable Quantity) will be used. This means that a firm can execute against multiple counterparties if the order's MAQ requirement is satisfied. For the Integrated order book this quantity is valid for non persistent orders only.</p> <p>In the Dark Midpoint order book, MES (Minimum Execution Size) will be used. This means that a firm will only execute against another order if that order alone meets the order's MES requirement. For the Dark Midpoint order book this quantity is valid for both persistent and non-persistent orders.</p> <p>Firms can also specify whether they want MES to apply for the first execution only or to persist for the lifetime of the order.</p> <p>Where MAQ/MES is greater than remaining Order Quantity, the MAQ/MES will be reduced to equal the remaining Order Quantity.</p>	MinQty (110)
Periodic Uncrossing Only	<p>These orders will only execute during a periodic uncrossing in the Dark Midpoint order book.</p> <p>This instruction will be ignored for the Integrated book.</p>	ExecInst (18) = z
Continuous Only	<p>These orders will only execute during continuous trading and will not match during Randomised Periodic Uncrossing events.</p> <p>This instruction will be ignored for the Integrated book.</p>	ExecInst (18) = y
Periodic Uncrossing & Continuous	These orders will execute both in continuous matching and in periodic un-crossings in the Dark Midpoint order book. This instruction will be ignored for the Integrated book.	ExecInst (18) = x
Day	An order that will expire at the end of the day.	TimeInForce (59) = 0
Immediate or Cancel (IOC)	An order that will be executed on receipt and the remainder immediately cancelled.	TimeInForce (59) = 3

Fill or Kill (FOK)	<p>An order that will be fully executed on receipt or immediately cancelled.</p> <p>An IOC order with MAQ set to order size will behave as a FOK order.</p>	<p>TimeInForce (59) = 4</p> <p>OR</p> <p>TimeInForce (59) = 3 and MinQty (110) = OrderQty (38)</p>
Good Till Time (GTT)	An order that will expire at a specified time during the current day, or at the end of day, which ever occurs earliest.	<p>TimeInForce (59) = 6</p> <p>ExpireTime (126)</p>
Good Till Date (GTD)	<p><b>No Longer supported by Turquoise.</b></p> <p>GTD orders will be accepted by the system but will be cancelled on market close.</p>	<p>TimeInForce (59) = 6</p> <p>ExpireDate (432)</p>
Good Till Cancelled (GTC)	<p><b>No Longer supported by Turquoise.</b></p> <p>GTC orders will be accepted by the system but will be cancelled on market close.</p>	TimeInForce (59) = 1

<p>Good For Auction (GFA)</p>	<p><b>Only applicable to TQ Dark Book.</b></p> <p>GFA orders with matching Instructions:</p> <ul style="list-style-type: none"> <li>• Continuous and Periodic Uncrossing will participate in the next Periodic Uncross within the Dark Midpoint Order Book for which it was intended to participate if submitted between a Call Market message and Periodic Uncrossing. Any unexecuted balance will be expired. Continuous and Periodic orders with TIF of GFA are not amendable.</li> <li>• Continuous and Periodic Uncrossing will act as IOC orders executing and expiring immediately if received after the next Periodic Uncross which it was intended to participate if submitted between a Periodic Uncrossing and Call Market message. Any unexecuted balance will be expired. Continuous and Periodic orders with TIF of GFA are not amendable.</li> <li>• Periodic Only will participate in the next Periodic Uncross within the Dark Midpoint Order Book after the next Call Market message. Any unexecuted balance will be expired. Periodic Only orders with TIF of GFA are amendable.</li> </ul>	<p>TimeInForce (59) = 9</p>
<p>Passive Only Order</p>	<p><b>Only applicable to persistent limit orders.</b></p> <p>These orders will not match with visible orders upon entry, and will expire if they will aggress.</p> <p>These orders <u>can</u> match on entry against large in scale hidden orders sat within the BBO.</p>	<p>PassiveOnlyOrder (27010) = 0, 99, 100, 1, 2, 3</p>

## 2.1.2 Order Management

### 2.1.2.1 Cancellation

The remainder of a live order may be cancelled via the [Order Cancel Request](#) message. The server will respond with an [Execution Report](#) or [Order Cancel Reject](#) to confirm or reject the cancellation request respectively.

An order can be cancelled either by specifying the OrderID (37) or by specifying the OrigClOrdID (41) in the [Order Cancel Request](#) message. Only the OrderID will be considered if both OrderID and OrigClOrdID are included in the message. The order book always needs to be explicitly specified using RoutingInst (9303). If an order submitted under a different SenderCompID (49) is being cancelled, the [Order Cancel Request](#) should include its OrderID (37).

### 2.1.2.2 Mass Cancellation

A client may mass cancel live orders via the [Order Mass Cancel Request](#) message. The server will respond with an [Order Mass Cancel Report](#) to indicate, via the Response (531) field, whether the request is successful or not. Clients may receive more than one Mass Cancel Report having different ApplIDs to distinguish the order cancellations carried out for each partition.

A mass cancellation request sent without the RoutingInst (9303) will only cancel orders in the Integrated order book. If a client wishes to cancel orders in the Dark Midpoint order book, 'M' should be specified in the RoutingInst (9303) or individual cancel requests must be sent as described in section [2.1.2.1 Cancellation](#) above.

If the cancellation request is accepted, the server will then immediately transmit [Execution Reports](#) for each order that is cancelled.

The ClOrdID(11) of all such messages will be the ClOrdID (11) of the [Order Mass Cancel Request](#).

If the mass cancel request is rejected, the reason will be specified in the MassCancelRejectReason(532) field of the [Order Mass Cancel Report](#).

When mass cancelling by instrument, the order book needs to be explicitly specified using RoutingInst (9303). Mass cancellation of all orders across both the Integrated and Dark Midpoint Books requires two mass cancel messages to be submitted, one for each book.

Clients may use the [Order Mass Cancel Request](#) to mass cancel all orders or only those for a particular instrument or segment. A mass cancel request may apply to all the orders of the member firm or only to those of a particular Trader Group. If the target party is not specified, the server will apply the request to the orders of the trading party that the [Order Mass Cancel Request](#) is submitted under. The FIX fields relevant to each of the supported mass cancel combinations are outlined below.

	Target Party	
	Other Party	Member Firm
All Orders	MassCancelRequestType (530) = 7  TargetPartyRole (1464) = 76  TargetPartyID (1462)	MassCancelRequestType (530) = 7  TargetPartyRole (1464) = 1  TargetPartyID (1462)
All Orders for an Instrument	MassCancelRequestType (530) = 1  Symbol (55)  RoutingInst (9303)  OR  SecurityID (48)  SecurityIDSource (22) = 4  RoutingInst (9303)  TargetPartyRole (1464) = 76  TargetPartyID (1462)	MassCancelRequestType (530) = 1  Symbol (55)  RoutingInst (9303)  OR  SecurityID (48)  SecurityIDSource (22) = 4  RoutingInst (9303)  TargetPartyRole (1464) = 1  TargetPartyID (1462)
All Orders for a Segment	MassCancelRequestType (530) = 9  MarketSegmentID (1300)  TargetPartyRole (1464) = 76  TargetPartyID (1462)	MassCancelRequestType (530) = 9  MarketSegmentID (1300)  TargetPartyRole (1464) = 1  TargetPartyID (1462)

**Example**

- To cancel all orders of VODI Dark Midpoint book of the party submitting the request
  - MassCancelRequestType (530) = 1
  - Symbol (55)= VODI
  - RoutingInst (9303) = M
  
- To cancel all orders of VODI Integrated book of the party submitting the request
  - MassCancelRequestType (530) = 1
  - Symbol (55) = VODI
  - RoutingInst (9303) = I
  
- To cancel all orders of VODI Dark Midpoint book of TraderGroup "TQ001". The request can be sent by a user having privileges to mass cancel firm orders.
  - MassCancelRequestType (530) = 1
  - Symbol (55) = VODI
  - RoutingInst (9303) = M
  - TargetPartyID (1462) = TQ001
  - TargetPartyRole (1464) = 76

### 2.1.2.3 Amending an Order

An open order may be amended via the [Order Cancel/Replace Request](#) message. The server will respond with an [Execution Report](#) or [Order Cancel Reject](#) to confirm or reject the amendment request respectively.

An order can be cancelled or amended either by specifying the OrderID (37) or by specifying the OrigClOrdID (41) in the [Order Cancel/Replace Request](#) message. Only the OrderID will be considered if both OrderID and OrigClOrdID are included in the message. The order book always needs to be explicitly specified using RoutingInst (9303).

The following attributes of a live order may be amended via the [Order Cancel/Replace Request](#) message:

- (i) Order quantity
- (ii) Displayed quantity \*
- (iii) Price
- (iv) Expiration time (GTT orders)
- ~~(v) Expiration date (GTD orders)~~
- (vi) Client reference
- (vii) Minimum Execution Size (valid for Dark Midpoint book)
- (viii) Execution Instruction (valid for Dark Midpoint book)
- (ix) Passive Only Order

\* The following restrictions apply. Participants may not:

- (i) amend a hidden order to become an Iceberg order (By specifying a Display Qty >0 on amend when Display Qty = 0 on original Order Submission to Dark or Lit Book)
- (ii) amend an Iceberg order to become a hidden order (By specifying a Display Qty = 0 on amend when Display Qty > 0 and <Order Qty on original Order Submission to Lit Book)
- (iii) amend a visible order to a hidden order (By specifying a Display Qty = 0 on amend when Display Qty = Order Qty on original Order Submission to Lit Book)
- (iv) amend a hidden order to a visible order (By specifying a Display Qty = Order Qty on amend when Display Qty =0 on original Order Submission to Lit Book)

Participants may:

- (i) amend a fully visible order to become an Iceberg order (By specifying a Display Qty < Order Qty and on amend when Display Qty = Order Qty on original Order Submission to Lit Book)
- (ii) amend an Iceberg order to become a visible order (By specifying a Display Qty = Order Qty on amend when Display Qty < Order Qty on original Order Submission to Lit Book)

An order's Passive Only Order value will not be re-evaluated unless the order's price is amended.

The server will respond with an [Execution Report](#) or [Order Cancel Reject](#) to confirm or reject the amendment request respectively.

An order will be reinstated and may lose priority if the following attributes are amended:

- (i) Increasing disclosed quantity of orders in the Integrated book
- (ii) Increasing order quantity of orders in the Integrated book
- (iii) Any amendment to order quantity of orders in the Dark Midpoint book
- (iv) Any amendment to Price
- (v) Any amendment to Minimum Execution Size (MES) in the Dark Midpoint book

A Passive Only order amended for price may expire if it falls into a worse price point or is to aggress a visible price point on the other side of the order book.

Clients can only amend open orders. When an order amended for price re-aggresses the order book where it gets fully filled the sender will receive only an Execution Report for the trade and not the amendment.

#### **2.1.2.4 Identifying Own Orders**

Clients can use the value specified under SecondaryOrderID (198) of the [Execution Report](#) message to identify own orders on the ITCH market data feed. MES of order to be submitted

#### **2.1.2.5 Identifying Own Orders**

Clients can use the value specified under SecondaryOrderID (198) of the [Execution Report](#) message to identify own orders on the ITCH market data feed.

#### **2.1.2.6 Cancellation by Market Supervision**

An unsolicited [Execution Report](#) will be sent to the client if an order is cancelled by Market Supervision. The ExecRestatmenetReason (378) of such a message will be Market (Exchange) option (8). It will not include an OrigClOrdID (41).

### **2.1.3 Order Status**

As specified in the FIX protocol, the OrdStatus (39) field is used to convey the current state of an order. If an order simultaneously exists in more than one order state, the value with highest precedence is reported as the OrdStatus (39). The relevant order statuses are given below from the highest to lowest precedence.



Value	Meaning
2	Filled
4	Cancelled
C	Expired
1	Partially Filled
0	New
8	Rejected

#### 2.1.4 Execution Reports

The [Execution Report](#) message is used to communicate many different events to clients. The events are differentiated by the value in the ExecType (150) field as outlined below.

ExecType	Usage	Ord Status
0	<p><b>Order Accepted</b></p> <p>Indicates that a new order has been accepted.</p> <p>This message will also be sent unsolicited if an order was submitted by Market Operations on behalf of the client.</p>	0
8	<p><b>Order Rejected</b></p> <p>Indicates that an order has been rejected. The reason for the rejection is specified in the field OrdRejReason (103).</p>	8
F	<p><b>Order Executed</b></p> <p>Indicates that an order has been partially or fully filled. The execution details (e.g. price and quantity) are specified.</p>	1, 2

C	<p><b>Order Expired</b></p> <p>Indicates that an order has expired in terms of its time qualifier or due to an execution limit.</p> <p>This message will also be sent in the following scenarios:</p> <ul style="list-style-type: none"> <li>(i) When orders are expired upon entering the order book when the number of orders in the order book is at the maximum allowed level. The reason for the expiration is specified in the Text (58) field.</li> <li>(ii) When the remaining orders are expired at market close.</li> <li>(iii) When orders are expired based on the auto cancellation on disconnect/log out feature.</li> <li>(iv) When the incoming order is configured with the self execution prevention specifying CIO or CRO.</li> <li>(v) When a Periodic Only GFA Order has not been fully executed in the Periodic Uncrossing to which it was expected to participate,</li> <li>(vi) When a Continuous and Periodic GFA Order has not been fully executed in the Periodic Uncrossing to which it was expected to participate,</li> <li>(vii) When a Continuous and Periodic GFA Order is submitted between a Periodic Uncrossing and a Call Market, it will act as an IOC, with any remaining quantity being expired.</li> </ul>	C
4	<p><b>Order Cancelled</b></p> <p>Indicates that an order cancel request has been accepted and successfully processed.</p> <p>This message will also be sent unsolicited if the order was cancelled by Market Operations. In such a scenario the Execution Report will include an ExecRestatementReason (378) of Market Option (8). It will not include an OrigClOrdID (41).</p> <p>This message will also be sent if Market Operations has cancelled a trade that previously fully filled the order (which would also result in a Trade Cancel Execution Report for that trade).</p>	4
5	<p><b>Order Cancel/Replaced</b></p> <p>Indicates that an order cancel/replace request has been accepted and successfully processed.</p>	0, 1

D	<b>Order Restatement</b>  Indicates that an order has been amended or a trade cancelled by Market Operations that previously partially filled the order (which would also result in a Trade Cancel Execution Report for that trade). The unsolicited message will include an ExecRestatement Reason (378) of Market Option (8). It will not include an OrigClOrdID (41).	0, 1
H	<b>Trade Cancel</b>  Indicates that an execution has been cancelled by Market Operations. An ExecRefID (19) to identify the execution being cancelled will be included.	0, 1, 4, C

## 2.1.5 Order and Execution Identifiers

### 2.1.5.1 Client Order IDs

The server does not validate the ClOrdID (11) for uniqueness. Clients should comply with the FIX protocol and ensure unique ClOrdIDs across all messages (e.g. New Order – Single, Order Cancel Request, etc.) sent under a particular SenderCompID (49).

Although GTC and GTD orders are accepted by the system they are expired on market close. Therefore ClOrdID (11) does not have to be unique across trading days. If a client submits multiple orders with the same ClOrdID on the same day, they will only be able to cancel/amend the most recent order.

Clients must, in terms of the FIX protocol, either specify the ClOrdID (11) or OrderID (37) when submitting an [Order Cancel Request](#), [Order Mass Cancel Request](#) or [Order Cancel/Replace Request](#). Only the OrderID will be considered if they send both IDs. Clients also need to specify RoutingInst (9303).

### 2.1.5.2 Order IDs

The server will use the OrderID (37) field of the [Execution Report](#) to affix the order identification numbers of the trading engine. Order IDs will be unique across trading days.

This is an 11 character 62 base string with an 'O' prefix. After removal of the prefix, when converted to an 8 byte binary format, it will match the corresponding ITCH Order ID. This will be identical to the SecondaryOrderID (198) when converted to a 16 character hexadecimal format. Thus, FIX OrderID (37), FIX SecondaryOrderID (198), and ITCH OrderID (37) are all representations of the same identifier [in base 62 (plus 'O' prefix), hexadecimal, and binary formats respectively].

In terms of the FIX protocol, unlike ClOrdID (11) which requires a chaining through cancel/replace requests and cancel requests, the OrderID (37) of an order will remain constant throughout its life.

Clients have the option of specifying the OrderID (37) when submitting an [Order Cancel Request](#) or [Order Cancel/Replace Request](#).

### 2.1.5.3 Execution IDs

The server will use the ExecID (17) field to affix a unique identifier for each [Execution Report](#). ExecIDs will be unique across trading days.

TradeMatchID (880) will correspond to the unique trade identifier sent with each trade to the CCPs. The unique trade identifier sent to the CCPs will contain an additional prefix to indicate the side and a '1' to indicate if the trade was cancelled. Clients are expected to derive this information by looking at the Side (54) and ExecType (150).

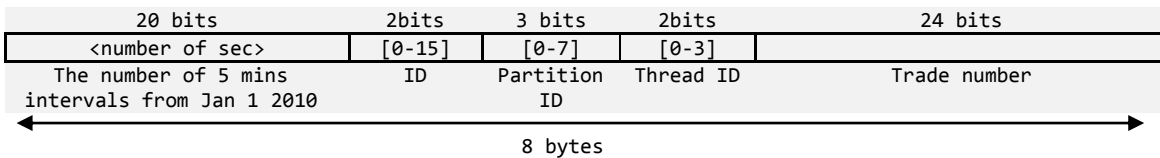
### 2.1.5.4 Trade Match ID

The TradeMatchID (880) in the FIX trading gateway matches exactly with the TradeID (1003) on the Trade Capture Report of Post Trade gateway. This also matches the TradeMatchID field from the Native Trading gateway as well as the ITCH gateway which are in binary format. However this is in base 62 and needs converting to an 8 byte integer for comparison after removing the prefix (T/M for 'on book' / 'off book').

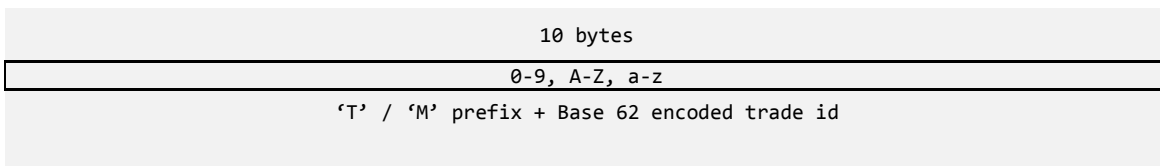
### 2.1.5.5 Mapping FIX TradeMatchID to ITCH TradeMatchID

Remove prefix 'T' / 'M' and convert from base 62 to binary (decimal)

- Trade ID format (binary)



- Trade ID format (ASCII)



Example:

ASCII trade ID for FIX	T0VsRW8NQq
Binary trade ID (decimal) for ITCH	112262475939900

Character	T	0	V	s	R	W	8	N	Q	q
Value	29	0	31	54	27	32	8	23	26	52

TradeMatchID generated above for a normal trade being disseminated through each gateway.

FIX Trading	Native Trading	Drop Copy	Post Trade	ITCH
TradeMatchID (880)	TradeMatchID	TradeMatchID (880)	TradeID (1003)	TradeMatchID
T0VsRW8NQq	112262475939900	T0VsRW8NQq	T0VsRW8NQq	112262475939900

### 2.1.5.6 Application ID (AppID)

The trading system consists of a series of parallel partitions each of which services an exclusive set of instruments. Each application message transmitted by the server will include the identity of the partition that generated the message. The number of partitions could increase/decrease in the future.

### 2.1.5.7 MDEntryID

MDEntryID(278) is a secondary order ID that will be maintained by the matching engine, which will be unique for each replenishment of a particular iceberg order.

For example for a single iceberg order, the Order ID will be the same, but a unique new Public Order ID will be generated for each replenishment.

### 2.1.5.8 Order ID tag length.

The system will accept a maximum length of 20 characters. If the ID is longer than 20 characters then it will be rejected. This is valid for the following.

NewOrderSingle – CIOrdID (11)

OrderCancelRequest – OriginalCIOrdID (41)

NewOrderSingle – SecondaryCIOrdID (526)

NewOrderSingle – CIOrdLinkID (583)

Quote – QuoteMsgID (1166)

## 2.2 Liquidity Pools

The Turquoise MTF supports the following liquidity pools for clients to execute their interest:

- (i) Integrated Order Book – The Integrated Order Book will execute orders in a continuous price-time method with large in scale hidden orders getting the lowest priority. Clients have the option to specifying the minimum fill size per order for non-persistent orders only.
- (ii) Dark Midpoint Order Book – The Dark Midpoint Order Book accepts only dark orders. Orders will execute at the Primary Market Midpoint, on entry and at uncrossings done at randomized time intervals, midpoint changes or when a firm amends order price, order size or MES. Clients have the option of specifying a minimum fill size per order.

Orders in the Dark Midpoint Order Book will be matched and prioritised on a Size then Time basis.

Clients can submit orders to the Integrated or Dark Midpoint book by explicitly specifying the order book in the RoutingInst (9303) tag. If the tag is not specified the order will be routed as described in section [2.2.3 Routing Orders when RoutingInst \(9303\) is not specified](#).

## 2.3 Symbology Schemes

Clients can use one or both of the following symbology schemes to manage their trading interest.

### 2.3.1 MTF Common Symbol

Clients can submit and manage orders by specifying the MTF Common Symbol. If using the MTF Common Symbol, the client:

- Must specify the Common Symbol in the Symbol (55) tag.
- Optionally specify the order book in the RoutingInst (9303) tag as 'I' for Integrated order book or 'M' for Dark Midpoint order book.
- Optionally specify the corresponding ISIN (48) with IDSource (22) set to '4', Currency (15) and Security Exchange (207).

The system will validate ISIN, Currency and Security Exchange values with the MTF Common Symbol for consistency.

### 2.3.2 ISIN, Currency and Security Exchange

Clients wishing to use only the ISIN, Currency and Security Exchange to uniquely identify an instrument:

- (i) Must not specify a value for Symbol (55)
- (ii) Must specify the ISIN value as the SecurityID (48) with SecurityIDSource (22) set to '4'
- (iii) Must specify the Currency (15)
- (iv) Must specify the SecurityExchange (207)
- (v) Optionally specify the order book in the RoutingInst (9303) tag as 'I' for Integrated order book or 'M' for Dark Midpoint order book.

### 2.3.3 Routing Orders when RoutingInst (9303) is not specified

Orders without a RoutingInst (9303) will be sent to the Integrated and TQ Dark Midpoint order book based on values specified for TIF (59), OrdType (40), DisplayQty (1138), Price (44), OrderQty (38), MinQty (110). Note that it is mandatory to stamp the Routing Instruction when routing orders to the client Dark Book.

#### DisplayQty = 0

- (i) With one exception, any order with DisplayQty=0 (irrespective of OrdType, TIF, MinQty) will be routed to the TQ Dark Midpoint Book.
- (ii) The exception is for Limit orders with a MinQty of Zero or Null, a TIF of DAY/ /GTT, and a value (OrderQty x Price) greater than the LIS threshold – these will be routed to the Integrated Book as hidden orders.

## **DisplayQty <> 0 (including Null values)**

- (i) Any order with OrdType=Peg will be routed to the TQ Dark Midpoint Book
- (ii) All other orders will be routed to the Integrated Book

## **2.4 Market Operations**

### **2.4.1 Order Deletion**

Market Operations are able to cancel orders on behalf of a client in accordance with the Turquoise Rulebook.

The client will be notified of the Order Cancel Request submitted on its behalf if and when it is accepted. The client will not be notified if the action is rejected.

This feature is intended to help a client manage an emergency situation and should not be relied upon as a normal business practice.

### **2.4.2 Trade Cancellations**

Market Operations may cancel any on-book trade. The server will transmit [Execution Report](#) messages to the relevant clients to notify them of a trade cancellation.

If an execution that partially filled an order is cancelled the order will be restated to reduce its order quantity by the cancelled quantity. The client will receive two notifications in such a scenario; one for the trade cancel and another for the restatement. The LeavesQty (151) and CumQty (14) of a live order will always add up to its OrderQty (38).

If an execution that fully filled an order is cancelled, the order will be cancelled. The client will also receive two notifications in such a scenario; one for the trade cancel and another for the order cancel.

## **2.5 Timestamps and Dates**

The timestamps SendingTime (52), OrigSendingTime (122) and TransactTime (60) should be in UTC and in the YYYYMMDD-HH:MM:SS.sss format.

ExpireTime (126) should be in UTC and in the YYYYMMDD-HH:MM:SS format.

All dates (i.e. ExpireDate (432)) should be in the YYYYMMDD format and specified in the local date for the server (i.e. not in UTC)).



## 2.6 Party Identification

ID	Description	Relevant FIX Tags
Trader Group	Identifier of the trader group the order is submitted under.	PartyRole (452) = 76  PartyID (448)
Trader ID	Identifier of the trader the order is submitted under.	PartyRole (452) = 12  PartyID (448)
Counterparty Firm	Identifier of the firm with whom the order was executed	PartyRole (452) = 17  PartyID (448)
Client Reference	Client reference information applicable to an order	Account (1)

Trading privileges are, depending on how the participant is set up, assigned at the level of the SenderCompID (49) or Trader Group.

It will be mandatory to specify a Trader Group (Party Role (452) = 76) in New Order – Single, Order Cancel and Order Cancel/Replace messages; it will be optional to specify a Trader ID (Party Role (452) = 12) in these messages. Counterparty Firm (Party Role (452) = 17) should never be specified in New Order – Single, Order Cancel and Order Cancel/Replace messages.

For the New Order Single (D), Order Cancel Request (F) and Order Cancel/Replace Request (G) messages, the message will be rejected if the Trading Party Component does not include a Party ID (448) Tag without a corresponding Party Role (452) Tag equal to 76 (Trader Group) within the same repeating group. Any messages rejected will be acknowledged to the client with a Business Message Reject (j) message with the following tags specified:

Business Reject Reason (380) = '0'

Text (58) = "Trader Group not specified on message"

In Execution Report messages, if the Exec Type (150) is F (Trade) or H (Trade Cancel), both the Trader Group (Party Role (452) = 76) and Counterparty Firm (Party Role (452) = 17) will be populated; if the Exec Type (150) is not F, G or H, then only the Trader Group (Party Role (452) = 76) will be populated. All the time Trader ID (Party Role (452) = 12) will be populated in the Execution Report message if the client has specified one in the New Order Single message.

## 2.7 Information for Billing

Customers may use the FIX Execution Report to estimate billing. For the current Turquoise rebates and fees, please refer to the [TQ Equity Tariff Schedule](#).

In general, rebates and fees can be determined via FIX tags 9303 (RoutingInst) and 9730 (TradeLiquidityIndicator):

- (i) Integrated Order Book – aggressive trades
  - 9303=I and 9730=R
- (ii) Integrated Order Book – passive trades
  - 9303=I and 9730=A
- (iii) Dark Midpoint Order Book – all trades
  - 9303=M only (tag 9730 is not required for calculation)
- (iv) In addition, the following FIX tags may be relevant for rebates during new market segment promotions:
  - 55 (MTF Common Symbol)
  - 48 (Security ID)
  - 207 (Security Exchange)

## 2.8 Order Capacity

Turquoise recognises four order capacities; Agency, Principal, Riskless Principal and CFD Give Up.

The capacity “CFD Give Up” will be applicable only for Sponsored Access users who are given privileges for capacity conversion. If the submitter with appropriate user privileges submits an order with dealing capacity “CFD Give Up”, such orders will be converted to “Principal” upon successful submission of the order. This new capacity type and related behaviour is relevant for users of the Native interface but has been added to this document for consistency purposes.

## 2.9 Repeating Groups (Components/Component Block)

If a repeating group is used in a message, the NoXXX field (for example NoPartyIDs field in the trading party repeating group) should be specified first before the repeating group starts. This is applicable for both the messages generated by the client and the server.

The messages generated by the server will have the fields within a repeating group in order.

The messages generated by a client should have the first field in a repeating group in order. If the first field in a repeating group is in order, a message generated by a client will be accepted; else the message will be rejected.

## **2.10 Auto Cancel on Disconnect**

In enabling the feature Mass Cancel on Disconnect, all open orders belonging to the respective client would get cancelled.

With the subsequent login the client will receive execution reports for each order with the ExecType: 'Expired', as opposed to 'Cancelled' which would be for orders manually cancelled by the client.

## **2.11 Generating Reject Messages**

If a required tag is missing in a message sent by a client, the server will send a session reject message.

If a conditionally required tag is missing in a message sent by a client, the server will send a business reject message.

The server will also send a session reject message if the same FIX tag has been repeated within the client request.

If an unsupported value is sent with a tag, an Execution Report or an Order Cancel Reject is sent by the server.

Session level validations are done first, and Business Rejects and rejections via Execution Reports follow in that order.

---

## 3.0 Connectivity

### 3.1 ComplIDs

The ComplID of each client must be registered with Turquoise before FIX communications can begin. A single client may have multiple connections to the server (i.e. multiple FIX sessions, each with its own ComplID).

The ComplID of the server will be FGW. The messages sent to the server should contain the ComplID assigned to the client in the field SenderComplID (49) and FGW in the field TargetComplID (56). The messages sent from the server to the client will contain FGW in the field SenderComplID (49) and the ComplID assigned to the client in the field TargetComplID (56).

#### 3.1.1 Passwords

Each new ComplID will be assigned a password on registration. Clients are strongly encouraged to change the password to one of their choosing via the [Logon](#) message. The status of the new password (i.e. whether it is accepted or rejected) will be specified in the SessionStatus (1409) field of the [Logon](#) sent by the server to confirm the establishment of a FIX connection. The new password will, if accepted, be effective for subsequent logins.

In terms of the password policy of Turquoise, the password of each ComplID should be changed. If not, the password will expire and the client will be unable to login to the server. In such a case, the client should contact Turquoise to have its password reset. The SessionStatus (1409) of the server's [Logon](#) message will be 'Password Due to Expire (2)'.

### 3.2 Production IP Address and Ports

The IP addresses and ports for the post trade gateway will be published in a separate configuration document.

### 3.3 Failover and Recovery

The system has been designed with fault tolerance and disaster recovery technology that ensures that trading should continue in the unlikely event of a process or site outage.

If the client is unexpectedly disconnected from the server, it should attempt to re-connect to primary site within a few seconds. The client should only attempt to connect to the secondary IP address and port if so requested by Turquoise.

### 3.4 Message Rate Throttling

Turquoise has implemented a scheme for throttling message traffic where each client is only permitted to submit up to a specified number of messages per second.

Every message which exceeds the maximum rate of a ComplID will be rejected via a [Reject](#).

A client's connection will be disconnected by the server if its message rate exceeds the maximum rate for a specific time duration. In such a case, the server will transmit a [Logout](#) message and immediately terminate the TCP/IP connection.

---

## 4.0 FIX Connections and Sessions

### 4.1 Establishing a FIX Connection

FIX connections and sessions between the client and server are maintained as specified in the FIX protocol.

Each client will use the assigned IP address and port to establish a TCP/IP session with the server. The client will initiate a FIX session at the start of each trading day by sending the [Logon](#) message. The client will identify itself using the SenderCompID (49) field. The server will validate the CompID, password and IP address of the client.

Once the client is authenticated, the server will respond with a [Logon](#) message. The SessionStatus (1409) of this message will be Session Active (0). If the client's [Logon](#) message included the field NewPassword (925) and the client is authenticated, the SessionStatus (1409) of the [Logon](#) sent by the server will be Session Active (0).

When the client sends a logon with a sequence number higher than expected by the FIX Gateway, the FIX gateway will send a [Resend Request](#) and once the response/s to the [Resend Request](#) is processed by the FIX Gateway, the FIX Gateway would send a [Test Request](#) to make sure both the client and server is in sync before sending out any missed or new application messages.

The client must wait for the server's [Logon](#) before sending additional messages. If additional messages are received from the client before the exchange of [Logon](#) messages, the TCP/IP connection with the client will be disconnected.

If a logon attempt fails because of an invalid SenderCompID, TargetCompID, IP address, invalid password or because the user does not have the appropriate privileges, the server will break the TCP/IP connection with the client without sending a [Logout](#) or [Reject](#) message. If during a logon of a SenderCompID, the server receives a second connection attempt via different TCP/IP connection while a valid FIX session is already underway for that same SenderCompID, the server will break the TCP/IP connection with the second connection without sending a [Logout](#) or [Reject](#) message. As the logon attempt failed, the server will not increment the next inbound message sequence number expected from the client.

If a logon attempt fails because of an invalid or expired password a locked CompID or if logins are not currently permitted, the server will send a [Logout](#) message and then break the TCP/IP connection with the client. If during a logon of a SenderCompID, the server receives a second connection attempt via the same TCP/IP connection while a valid FIX session is already underway for that same SenderCompID, the server will send a [Reject](#) message and then break the TCP/IP connection with the client. The server will increment the next inbound message sequence number expected from the client as well as its own outbound message sequence number.

If a logon attempt fails because of a session level failure (e.g. due to invalid EncryptMethod or DefaultAppVerID...etc) the inbound sequence number and the outbound sequence number both will not be incremented. In this scenario the message sequence number 1 will be sent with the [Logout](#) message.

However if a session level failure occurs due to a message sent by a client which contains a sequence number that is less than what is expected and the PossDupFlag (43) not being set to “Y”, then the server will send a Logout message and terminate the FIX connection. In this scenario the inbound sequence number will not be incremented but the outbound sequence number will be incremented.

If during a logon of a SenderCompID, the server receives a second connection attempt via the same TCP/IP connection while a valid FIX session is already underway for that same SenderCompID, the server will send a [Reject](#) message and then break the TCP/IP connection with the client. The server will increment the next inbound message sequence number expected from the client as well as its own outbound message sequence number.

## 4.2 Maintaining A FIX Session

### 4.2.1 Message Sequence Numbers

As outlined in the FIX protocol, the client and server will each maintain a separate and independent set of incoming and outgoing message sequence numbers. Sequence numbers should be initialized to 1 (one) at the start of the FIX session and be incremented throughout the session.

Monitoring sequence numbers will enable parties to identify and react to missed messages and to gracefully synchronize applications when reconnecting during a FIX session.

If any message sent by the client contains a sequence number that is less than what is expected and the PossDupFlag (43) is not set to “Y”, the server will send a [Logout](#) message and terminate the FIX connection. The [Logout](#) will contain the next expected sequence number in the Text (58) field.

A FIX session will not continue to the next trading day. The server will initialize its sequence numbers at the start of each day. The client is expected to employ the same logic.

### 4.2.2 Heartbeats

The client and server will use the [Heartbeat](#) message to exercise the communication line during periods of inactivity and to verify that the interfaces at each end are available. The heartbeat interval will be the HeartBtInt (108) specified in the client’s [Logon](#) message.

The server will send a [Heartbeat](#) anytime it has not transmitted a message for the heartbeat interval. The client is expected to employ the same logic.

If the server detects inactivity for a period longer than the heartbeat interval plus a reasonable transmission time, it will send a [Test Request](#) message to force a [Heartbeat](#) from the client. If inactivity continues for a second heartbeat interval plus a reasonable transmission time, the server will send a [Logout](#) and break the TCP/IP connection with the client. The client is expected to employ similar logic if inactivity is detected on the part of the server.

### 4.2.3 Increasing Expected Sequence Number

The client or server may use the [Sequence Reset](#) message in Gap Fill mode if it wishes to increase the expected incoming sequence number of the other party. The client or server may also use the [Sequence Reset](#) message in Sequence Reset mode if it wishes to increase the expected incoming sequence number of the other party. The Sequence Reset mode should only be used to recover from an emergency situation. It should not be relied upon as a regular practice.

### 4.3 Terminating a FIX Session

The client is expected to terminate each FIX connection at the end of each trading day before the server shuts down. The client will terminate a connection by sending the [Logout](#) message. The server will respond with a [Logout](#) to confirm the termination. The client will then break the TCP/IP connection with the server.

All open TCP/IP connections will be terminated by the server when it shuts down (a [Logout](#) will not be sent). Under exceptional circumstances the server may initiate the termination of a connection during the trading day by sending the [Logout](#) message.

If, during the exchange of [Logout](#) messages, the client or server detects a sequence gap, it should send a [Resend Request](#).

### 4.4 Re-Establishing a FIX Session

If a FIX connection is terminated during the trading day it may be re-established via an exchange of Logon messages.

Once the client is authenticated, the server will respond with a [Logon](#) message. The SessionStatus (1409) of this message will be Session Active (0). If the client's [Logon](#) message included the field NewPassword (925) and the client is authenticated, the SessionStatus (1409) of the [Logon](#) sent by the server will be Session Active (0).

When the client sends a logon with a sequence number higher than expected by the FIX Gateway, the FIX gateway will send a [Resend Request](#) and once the response/s to the [Resend Request](#) is processed by the FIX Gateway, the FIX Gateway would send a [Test Request](#) to make sure both the client and server is in sync before sending out any missed or new application messages.

The client must wait for the server's [Logon](#) before sending additional messages. If additional messages are received from the client before the exchange of [Logon](#) messages, the TCP/IP connection with the client will be disconnected.

Once the FIX session is re-established successfully, the message sequence numbers will continue from the last message successfully transmitted prior to the termination.

#### 4.4.1 Resetting Sequence Numbers

##### 4.4.1.1 Reset Initiated by the Client

If the client requires both parties to initialize (i.e. reset to 1) sequence numbers, it may use the ResetSeqNumFlag (141) field of the [Logon](#) message. The server will respond with a [Logon](#) with the ResetSeqNumFlag (141) field set to "Y" to confirm the initialization of sequence numbers.

A client may also manually inform Market Operations that it would like the server to initialize its sequence numbers prior to the client's next login attempt.

These features are intended to help a client manage an emergency situation. Initializing sequence numbers on a re-login should not be relied upon as a regular practice.

#### 4.4.1.2 Reset Initiated by the Server

The system has been designed with fault tolerance and disaster recovery technology that should ensure that the server retains its incoming and outgoing message sequence numbers for each client in the unlikely event of an outage. However, clients are required to support a manual request by Turquoise to initialize sequence numbers prior to the next login attempt.

#### 4.5 Dormant Account Policy

Clients are advised that CompIDs for both the Native and FIX Trading services will automatically be deactivated after a period of 100 days without a successful logon.

If a client is unable to connect because a CompID has been marked as inactive, they should contact Turquoise Market Operations who will reactivate CompIDs as required.

Clients that may have allocated specific Trading CompIDs for a disaster recovery site are strongly advised to take note of the above.

---

## 5.0 Recovery

### 5.1 Resend Requests

The client may use the [Resend Request](#) message to recover any lost messages. As outlined in the FIX protocol, this message may be used in one of three modes:

- (i) To request a single message. The BeginSeqNo (7) and EndSeqNo (16) should be the same.
- (ii) To request a specific range of messages. The BeginSeqNo (7) should be the first message of the range and the EndSeqNo (16) should be the last of the range.
- (iii) To request all messages after a particular message. The BeginSeqNo (7) should be the sequence number immediately after that of the last processed message and the EndSeqNo (16) should be zero (0).

The server caches the last 65,000 messages transmitted to each CompID. Clients are unable to use a [Resend Request](#) to recover messages not in the server's cache. If the client requests for a range of messages that have sequence numbers falling outside the cache size, a [Sequence Reset](#) message in Gap Fill mode will be sent for the missing messages and will send the available messages as per the request after that.

### 5.2 Possible Duplicates

The server handles possible duplicates according to the FIX protocol. The client and server will use the PossDupFlag (43) field to indicate that a message may have been previously transmitted with the same MsgSeqNum (34).



## 5.3 Possible Resends

### 5.3.1 Client-Initiated Messages

The server does not handle possible resends for the client-initiated messages (e.g. New Order Single) and ignores the value in the PossResend (97) field of such messages.

### 5.3.2 Server-Initiated Messages

The server may, in the circumstances outlined in sections [5.4 Transmission of Missed Messages](#) and [5.5 Resending Previous Execution Reports](#), use the PossResend (97) field to indicate that an application message may have already been sent under a different MsgSeqNum (34). The client should validate the contents (e.g. ExecID) of such a message against those of messages already received during the current trading day to determine whether the new message should be ignored or processed.

## 5.4 Transmission of Missed Messages

The [Execution Report](#), [Order Cancel Reject](#), [Order Mass Cancel Report](#), and [Business Message Reject](#) messages generated during a period when a client is disconnected from the server will be sent to the client when it next reconnects. In the unlikely event the disconnection was due to an outage of the server, all such messages will include a PossResend (97) of "Y".

The application messages (e.g. [Execution Report](#), [Order Cancel Reject](#) etc.) are automatically generated when a client reconnects. Clients are not required to explicitly request for the messages. The resend request applies only when the server has sent messages that a client has not received.

## 5.5 Resending Previous Execution Reports

A client may manually inform Market Operations that it would like the server to resend all of the [Execution Report](#) messages that were generated for the client during the current trading day when it next logs in. All resent [Execution Report](#) messages will include a PossResend (97) of "Y".

This feature is intended to help a client manage an emergency situation and it should not be relied upon as a regular practice.

---

## 6.0 Message Formats

This section provides details on the header and trailer, the seven administrative messages and eight application messages utilized by the server. Any message not included in this section will be ignored by the server.

### 6.1 Supported Message Types

#### 6.1.1 Administrative Messages

All administrative messages may be initiated by either the client or the server.

Message	MsgType	Usage
<a href="#">Logon</a>	A	Allows the client and server to establish a FIX session.
<a href="#">Logout</a>	5	Allows the client and server to terminate a FIX session.
<a href="#">Heartbeat</a>	0	Allows the client and server to exercise the communication line during periods of inactivity and verify that the interfaces at each end are available.
<a href="#">Test Request</a>	1	Allows the client or server to request a response from the other party if inactivity is detected.
<a href="#">Resend Request</a>	2	Allows for the recovery of messages lost during a malfunction of the communications layers.
<a href="#">Reject</a>	3	Used to reject a message that does not comply with FIXT.
<a href="#">Sequence Reset</a>	4	Allows the client or server to increase the expected incoming sequence number of the other party.

#### 6.1.2 Application Messages: Order Handling

##### 6.1.2.1 Client-Initiated

Message	MsgType	Usage
<a href="#">New Order Single</a>	D	Allows the client to submit a new order.
<a href="#">Order Cancel Request</a>	F	Allows the client to cancel a live order.

<a href="#">Order Mass Cancel Request</a>	q	<p>Allows the client to mass cancel:</p> <ul style="list-style-type: none"> <li>i) All live orders.</li> <li>ii) All live orders for a particular instrument.</li> <li>iii) All live orders for a particular segment.</li> </ul> <p>The mass cancel may apply to the orders of a particular trading party or to all orders of the member.</p>
<a href="#">Order Cancel/Replace Request</a>	G	<p>Allows the client to cancel/replace a live order.</p>

### 6.1.2.2 Server-Initiated

Message	MsgType	Usage
---------	---------	-------

<a href="#">Execution Report</a>	8	Indicates one of the following:  i) Order accepted.  ii) Order rejected.  iii) Order executed.  iv) Order expired.  v) Order cancelled.  vi) Order cancel/replaced.  vii) Trade cancel.
<a href="#">Order Cancel Reject</a>	9	Indicates that an order cancel request or order cancel/replace request has been rejected.
<a href="#">Order Mass Cancel Report</a>	r	Indicates one of the following:  i) Mass order cancel request accepted.  ii) Mass order cancel request rejected.
<a href="#">Business Message Reject</a>	j	Indicates that an application message could not be processed.

## 6.2 Message Header and Trailer

### 6.2.1 Message Header

Tag	Field Name	Req	Description
8	BeginString	Y	FIXT.1.1
9	BodyLength	Y	Number of characters after this field up to and including the delimiter immediately preceding the CheckSum.
35	MsgType	Y	Message type.
49	SenderCompID	Y	CompID of the party sending the message.

56	TargetCompID	Y	<p>CompID of the party the message is sent to.</p> <p><b>Value    Meaning</b></p> <hr/> <p>FGW      FIX Trading Gateway</p>
34	MsgSeqNum	Y	Sequence number of the message.
43	PossDupFlag	N	<p>Whether the message was previously transmitted under the same MsgSeqNum (34). Absence of this field is interpreted as Original Transmission (N).</p> <p><b>Value    Meaning</b></p> <hr/> <p>Y          Possible Duplicate</p> <hr/> <p>N          Original Transmission</p>
97	PossResend	N	<p>Whether the message was previously transmitted under a different MsgSeqNum (34). Absence of this field is interpreted as Original Transmission (N).</p> <p><b>Value    Meaning</b></p> <hr/> <p>Y          Possible Resend</p> <hr/> <p>N          Original Transmission</p>
52	SendingTime	N	Time the message was transmitted. Not required for incoming messages sent by the clients (even if sent by a client, no validation will be done). Required for outgoing messages sent by the server.
122	OrigSendingTime	N	Time the message was originally transmitted. If the original time is not available, this should be the same value as SendingTime (52). Required if PossDupFlag (43) is Possible Duplicate (Y).
1128	ApplVerID	N	<p>Version of FIX used in the message. Required if the message is generated by the server.</p> <p><b>Value    Meaning</b></p> <hr/> <p>9          FIX50SP2</p>
115	OnBehalfOfCompID	N	The ID of the party on whose behalf the message is sent; will only be used in client initiated messages

128	DeliverToCompID	N	The value specified in the OnBehalfOfCompID(115) field will be stamped; will only be used in server initiated messages
-----	-----------------	---	--

### 6.2.2 Message Trailer

Tag	Field Name	Req	Description
10	Checksum	Y	

## 6.3 Administrative Messages

### 6.3.1 Logon

Tag	Field Name	Req	Description
<u>Standard Header</u>			
35	MsgType	Y	A = Logon
Message Body			
98	EncryptMethod	Y	Method of encryption.  <b>Value    Meaning</b> <hr/> 0        None
108	HeartBtInt	Y	Indicates the heartbeat interval in seconds.
141	ResetSeqNum Flag	N	Indicates whether the client and server should reset sequence numbers. Absence of this field is interpreted as Do Not Reset Sequence Numbers (N).  <b>Value    Meaning</b> <hr/> Y        Reset Sequence Numbers <hr/> N        Do Not Reset Sequence Numbers
554	Password	N	Password assigned to the CompID. Required if the message is generated by the client.

925	NewPassword	N	New password for the CompID.										
1409	SessionStatus	N	Status of the FIX session or the request to change the password. Required if the message is generated by the server.  <table border="0"> <tr> <td><b>Value</b></td> <td><b>Meaning</b></td> </tr> <tr> <td colspan="2"><hr/></td> </tr> <tr> <td>0</td> <td>Session Active</td> </tr> <tr> <td colspan="2"><hr/></td> </tr> <tr> <td>2</td> <td>Password Due to Expire</td> </tr> </table>	<b>Value</b>	<b>Meaning</b>	<hr/>		0	Session Active	<hr/>		2	Password Due to Expire
<b>Value</b>	<b>Meaning</b>												
<hr/>													
0	Session Active												
<hr/>													
2	Password Due to Expire												
1137	DefaultAppVerID	Y	Default version of FIX messages used in this session.  <table border="0"> <tr> <td><b>Value</b></td> <td><b>Meaning</b></td> </tr> <tr> <td colspan="2"><hr/></td> </tr> <tr> <td>9</td> <td>FIX50SP2</td> </tr> </table>	<b>Value</b>	<b>Meaning</b>	<hr/>		9	FIX50SP2				
<b>Value</b>	<b>Meaning</b>												
<hr/>													
9	FIX50SP2												
<a href="#">Standard Trailer</a>													

### 6.3.2 Logout

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	5 = Logout
Message Body			

1409	SessionStatus	N	<p>Status of the FIX session. Required if the message is generated by the server.</p> <p><b>Value    Meaning</b></p> <hr/> <p>4            Session logout complete</p> <hr/> <p>5            Invalid password</p> <hr/> <p>6            Account locked</p> <hr/> <p>7            Logons are not allowed at this time</p> <hr/> <p>8            Password expired</p> <hr/> <p>100        Other</p> <hr/> <p>101        Logout due to session level failure</p> <hr/> <p>102        Logout by Market Operations</p>
58	Text	N	<p>The field will contain the next expected sequence number if the server terminated the connection after receiving a sequence number that was less than what was expected.</p> <p>In other cases the field will contain the reason for the logout.</p>
<a href="#">Standard Trailer</a>			

### 6.3.3 Heartbeat

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	0 = Heartbeat
Message Body			
112	TestReqID	N	Required if the heartbeat is a response to a Test Request. The value in this field should echo the TestReqID (112) received in the Test Request.
<a href="#">Standard Trailer</a>			



### 6.3.4 Test Request

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	1 = Test Request
Message Body			
112	TestReqID	Y	Identifier for the request.
<a href="#">Standard Trailer</a>			

### 6.3.5 Resend Request

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	2 = Resend Request
Message Body			
7	BeginSeqNo	Y	Sequence number of first message in range.
16	EndSeqNo	Y	Sequence number of last message in range.
<a href="#">Standard Trailer</a>			

### 6.3.6 Reject

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	3 = Reject

Message Body			
45	RefSeqNum	Y	MsgSeqNum (34) of the rejected message.
372	RefMsgType	N	MsgType (35) of the rejected message.
371	RefTagID	N	If a message is rejected due to an issue with a particular field its tag number will be indicated.
373	SessionReject Reason	N	Code specifying the reason for the reject. Refer to section <a href="#">7.2.1 Reject</a> for a list of reject codes.
58	Text	N	Text specifying the reason for the rejection.
<a href="#">Standard Trailer</a>			

### 6.3.7 Sequence Reset

Tag	Field Name	Req	Description						
<a href="#">Standard Header</a>									
35	MsgType	Y	4 = Sequence Reset						
Message Body									
36	NewSeqNo	Y	Sequence number of the next message to be transmitted.						
123	GapFillFlag	N	<p>Mode in which the message is being used. Absence of this field is interpreted as Sequence Reset (N).</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>Y</td> <td>Gap Fill</td> </tr> <tr> <td>N</td> <td>Sequence Reset</td> </tr> </tbody> </table>	Value	Meaning	Y	Gap Fill	N	Sequence Reset
Value	Meaning								
Y	Gap Fill								
N	Sequence Reset								
<a href="#">Standard Trailer</a>									

## 6.4 Application Messages: Order Handling

### 6.4.1 New Order Single

Tag	Field Name	Req	Description						
<u>Standard Header</u>									
35	MsgType	Y	D = New Order - Single						
Message Body									
11	ClOrdID	Y	Client specified identifier of the order.  (Max length 20 bytes)						
<u>Component Block &lt;Trading Party&gt;</u>		Y	Identifier of the trading party.						
1	Account	N	Client reference information.  (Max length: 10 bytes)						
55	Symbol	N	MTF Common Symbol.  Not required if 15, 48, 22 and 207 are specified.						
48	SecurityID	N	Identifier of the instrument.  Not required if 55 is specified						
22	SecurityIDSource	N	Identifier of the source of the SecurityID (48) value.  <table border="0"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>4</td> <td>ISIN</td> </tr> </tbody> </table>	Value	Meaning	4	ISIN		
Value	Meaning								
4	ISIN								
9303	RoutingInst	N	Indicate the liquidity pool  <table border="0"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>I</td> <td>Integrated order book</td> </tr> <tr> <td>M</td> <td>Dark Midpoint order book</td> </tr> </tbody> </table>	Value	Meaning	I	Integrated order book	M	Dark Midpoint order book
Value	Meaning								
I	Integrated order book								
M	Dark Midpoint order book								

15	Currency	N	Currency Code as per ISO 4217 Currency Code List  Not required if 55 is specified												
207	SecurityExchange	N	Market Identifier Code as per ISO 10383  Not required if 55 is specified												
18	ExecInst	N	Only applicable to the Dark Midpoint order book. If not specified, the default setup for order's owner will apply.  <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>x (default)</td> <td>Continuous and Periodic un-crossings</td> </tr> <tr> <td>z</td> <td>Periodic un-crossings only</td> </tr> <tr> <td>y</td> <td>Continuous only</td> </tr> </tbody> </table>	Value	Meaning	x (default)	Continuous and Periodic un-crossings	z	Periodic un-crossings only	y	Continuous only				
Value	Meaning														
x (default)	Continuous and Periodic un-crossings														
z	Periodic un-crossings only														
y	Continuous only														
40	OrdType	Y	Type of the order.  <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Market</td> </tr> <tr> <td>2</td> <td>Limit</td> </tr> <tr> <td>P</td> <td>Pegged</td> </tr> </tbody> </table>	Value	Meaning	1	Market	2	Limit	P	Pegged				
Value	Meaning														
1	Market														
2	Limit														
P	Pegged														
59	TimeInForce	N	Time qualifier of the order. Absence of this field is interpreted as DAY (0). Enum 6 (GTD) will only be supported with Expiry Time (GTT).  <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>DAY</td> </tr> <tr> <td>3</td> <td>Immediate or Cancel (IOC)</td> </tr> <tr> <td>4</td> <td>Fill or Kill (FOK)</td> </tr> <tr> <td>6</td> <td>Good Till Date (GTD)</td> </tr> <tr> <td>9</td> <td>Good for Auction (GFA)</td> </tr> </tbody> </table>	Value	Meaning	0	DAY	3	Immediate or Cancel (IOC)	4	Fill or Kill (FOK)	6	Good Till Date (GTD)	9	Good for Auction (GFA)
Value	Meaning														
0	DAY														
3	Immediate or Cancel (IOC)														
4	Fill or Kill (FOK)														
6	Good Till Date (GTD)														
9	Good for Auction (GFA)														

126	ExpireTime	N	<p>Time the order expires which must be a time during the current trading day. Clients who want to submit GTT orders must specify the time in this field and specify TimeInForce (59) as GTD (6).</p> <p>Required if TimeInForce (59) is GTD (6) and ExpireDate (432) is not specified.</p> <p>If both the ExpireTime (126) and ExpireDate (432) are specified, the system will ignore ExpireDate (432).</p>						
432	ExpireDate	N	<p><b>Not longer supported by Turquoise.</b></p> <p>Date the order expires. Required if TimeInForce (59) is GTD (6) and ExpireTime (126) is not specified. If both ExpireTime (126) and ExpireDate (432) are specified, the system will ignore ExpireDate (432)</p>						
54	Side	Y	<p>Side of the order.</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Value</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Buy</td> </tr> <tr> <td>2</td> <td>Sell</td> </tr> </tbody> </table>	Value	Meaning	1	Buy	2	Sell
Value	Meaning								
1	Buy								
2	Sell								
38	OrderQty	Y	Total order quantity.						
1138	DisplayQty	N	Maximum quantity that may be displayed.						
1084	DisplayMethod	N	<table border="0"> <thead> <tr> <th style="text-align: left;">Value</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td>4</td> <td>Undisclosed (Hidden Order)</td> </tr> <tr> <td>3</td> <td>Random (randomize value)</td> </tr> </tbody> </table> <p>If this is populated with value "4" while a value which is greater than 0 is populated in DisplayQty (1138), the order will be considered as a Hidden (Reserve) Order.</p> <p>If this is populated with value "3" while a value which is greater than 0 and less than the Order Quantity is populated in DisplayQty (1138), the DisplayQty (1138) after a replenishment will be random.</p> <p>If blank while a value which is greater than 0 and less than the Order Quantity is populated in DisplayQty (1138), the DisplayQty (1138) after a replenishment will be "fixed peak"</p>	Value	Meaning	4	Undisclosed (Hidden Order)	3	Random (randomize value)
Value	Meaning								
4	Undisclosed (Hidden Order)								
3	Random (randomize value)								

110	MinQty	N	Minimum fill size.										
44	Price	N	Limit price. Required if OrderType (40) is Limit (2)										
581	AccountType	Y	Type of account associated with the order.  <table border="0"> <thead> <tr> <th style="text-align: left;">Value</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Client</td> </tr> <tr> <td>3</td> <td>House</td> </tr> </tbody> </table>	Value	Meaning	1	Client	3	House				
Value	Meaning												
1	Client												
3	House												
528	OrderCapacity	Y	Capacity of the order.  <table border="0"> <thead> <tr> <th style="text-align: left;">Value</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>Agency</td> </tr> <tr> <td>P</td> <td>Principal</td> </tr> <tr> <td>R</td> <td>Riskless Principal</td> </tr> <tr> <td>C</td> <td>CFD Give Up</td> </tr> </tbody> </table>	Value	Meaning	A	Agency	P	Principal	R	Riskless Principal	C	CFD Give Up
Value	Meaning												
A	Agency												
P	Principal												
R	Riskless Principal												
C	CFD Give Up												
60	TransactTime	Y	Time the order was created.										
526	SecondaryClOrdID	N	A secondary ID assigned by the trading party. This field has a maximum of 20 characters.										
583	ClOrdLinkID	N	Permits order originators to tie together groups of orders in which trades resulting from orders are associated for a specific purpose.  e.g.. Calculation of average execution price.  This field has a maximum of 20 characters.										

27010	PassiveOnlyOrder	N	<p>Used to specify whether an order will rest prior to execution, with flexibility for visible orders to rest at a specified price level on the book.</p> <p>No protection is provided against order execution with large in scale hidden orders sat within the BBO.</p> <p>A hidden order will be rejected if it does not have a value of 0 or 99 (if tag 27010 is specified).</p> <p>A Dark Midpoint order will be rejected if it does not have a value of 0 (if tag 27010 is specified).</p> <table border="0"> <thead> <tr> <th data-bbox="639 757 703 786">Value</th> <th data-bbox="756 757 855 786">Meaning</th> </tr> </thead> <tbody> <tr> <td data-bbox="639 831 655 860">0</td> <td data-bbox="756 831 903 860">No constraint</td> </tr> <tr> <td data-bbox="639 898 727 927">(default)</td> <td data-bbox="756 898 1046 927">(i.e. aggressive or passive)</td> </tr> <tr> <td data-bbox="639 972 671 1001">99</td> <td data-bbox="756 972 1054 1055">Accept order only if passive upon order entry. Otherwise expire.</td> </tr> <tr> <td data-bbox="639 1099 679 1128">100</td> <td data-bbox="756 1099 1102 1151">Accept order if setting new BBO. Otherwise expire.</td> </tr> <tr> <td data-bbox="639 1196 647 1225">1</td> <td data-bbox="756 1196 1094 1279">Accept order if setting new BBO or joining existing BBO. Otherwise expire.</td> </tr> <tr> <td data-bbox="639 1323 647 1352">2</td> <td data-bbox="756 1323 1086 1406">Accept order if joining existing BBO or within one visible price point. Otherwise expire.</td> </tr> <tr> <td data-bbox="639 1451 647 1480">3</td> <td data-bbox="756 1451 1078 1534">Accept order if joining existing BBO or within two visible price points. Otherwise expire.</td> </tr> </tbody> </table>	Value	Meaning	0	No constraint	(default)	(i.e. aggressive or passive)	99	Accept order only if passive upon order entry. Otherwise expire.	100	Accept order if setting new BBO. Otherwise expire.	1	Accept order if setting new BBO or joining existing BBO. Otherwise expire.	2	Accept order if joining existing BBO or within one visible price point. Otherwise expire.	3	Accept order if joining existing BBO or within two visible price points. Otherwise expire.
Value	Meaning																		
0	No constraint																		
(default)	(i.e. aggressive or passive)																		
99	Accept order only if passive upon order entry. Otherwise expire.																		
100	Accept order if setting new BBO. Otherwise expire.																		
1	Accept order if setting new BBO or joining existing BBO. Otherwise expire.																		
2	Accept order if joining existing BBO or within one visible price point. Otherwise expire.																		
3	Accept order if joining existing BBO or within two visible price points. Otherwise expire.																		
9020	OrderSubType	N	<p>Used to specify the order type.</p> <table border="0"> <thead> <tr> <th data-bbox="639 1644 703 1673">Value</th> <th data-bbox="815 1644 914 1673">Meaning</th> </tr> </thead> <tbody> <tr> <td data-bbox="639 1718 767 1747">0/undefined</td> <td data-bbox="815 1718 879 1747">Order</td> </tr> </tbody> </table>	Value	Meaning	0/undefined	Order												
Value	Meaning																		
0/undefined	Order																		
<a href="#">Standard Trailer</a>																			

## 6.4.2 Order Cancel Request

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	F = Order Cancel Request
Message Body			
11	ClOrdID	Y	Client specified identifier of the cancel request.
41	OrigClOrdID	N	ClOrdID (11) of the order being cancelled. Required if OrderID (37) is not specified.
37	OrderID	N	Server specified identifier of the order being cancelled. Required if OrigClOrdID (41) is not specified.  This is an 11 character base 62 string with an 'O' prefix. After removal of the prefix, when converted to an 8 byte binary format, it will match the corresponding ITCH Order ID. This will be identical to the SecondaryOrderID when converted to a 16 character hexadecimal format
55	Symbol	N	MTF Common Symbol.  Not required if 15, 48, 22 and 207 are specified.
48	SecurityID	N	Identifier of the instrument.  Not required if 55 is specified
22	SecurityIDSource	N	Identifier of the source of the SecurityID (48) value.  <b>Value      Meaning</b> <hr/> 4            ISIN
15	Currency	N	Currency Code as per ISO 4217 Currency Code List  Not required if 55 is specified
207	SecurityExchange	N	Market Identifier Code as per ISO 10383  Not required if 55 is specified



9303	RoutingInst	Y	Indicate the liquidity pool.  <b>Value    Meaning</b> <hr/> I            Integrated order book <hr/> M            Dark Midpoint order book
<a href="#">Component Block &lt;Trading Party&gt;</a>		Y	Identifier of the trading party.
54	Side	Y	Must match the value in the order.
60	TransactTime	Y	Time the order cancel request was created.
<a href="#">Standard Trailer</a>			

#### 6.4.3 Order Mass Cancel Request

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	q = Order Mass Cancel Request
Message Body			
11	ClOrdID	Y	Client specified identifier of mass cancel request.
530	MassCancel RequestType	Y	Scope of the mass cancel request.  <b>Value    Meaning</b> <hr/> 1            Cancel All Orders for Instrument <hr/> 7            Cancel All Orders <hr/> 9            Cancel All Orders for Segment
55	Symbol	N	MTF Common Symbol.  Not required if 15, 48, 22 and 207 are specified.

48	SecurityID	N	Identifier of the instrument.  Not required if 55 is specified	
22	SecurityIDSource	N	Identifier of the source of the SecurityID (48) value.  <b>Value    Meaning</b> <hr/> 4        ISIN	
15	Currency	N	Currency Code as per ISO 4217 Currency Code List  Not required if 55 is specified	
207	SecurityExchange	N	Market Identifier Code as per ISO 10383  Not required if 55 is specified	
9303	RoutingInst	N	Indicate the liquidity pool. Required if MassCancelRequestType (530) = 1  <b>Value    Meaning</b> <hr/> I        Integrated order book <hr/> M        Dark Midpoint order book	
1461	NoTargetPartyIDs	Y	Number of parties the mass cancel relates to. The value in this field will always be "1".	
➔	1462	TargetPartyID	Y	Identifier of the party the mass cancel relates to. Required if NoTargetPartyIDs (1461) is specified.
➔	1463	TargetPartyIDSource	Y	The value in this field will always be "D".  <b>Value    Meaning</b> <hr/> D        Proprietary/Custom Code
➔	1464	TargetParty Role	Y	Role of the TargetPartyID (1462).  <b>Value    Meaning</b> <hr/> 1        Member ID (Firm) <hr/> 76      Trader Group

1300	MarketSegmentID	N	Identifier of the segment the mass cancel relates to. Required if MassCancelRequestType (530) is Cancel All for Segment (9).
60	TransactTime	Y	Time the mass cancel request was created.
<a href="#"><u>Standard Trailer</u></a>			

#### 6.4.4 Order Cancel/Replace Request

Tag	Field Name	Req	Description				
<u>Standard Header</u>							
35	MsgType	Y	G = Order Cancel/Replace Request				
Message Body							
11	ClOrdID	Y	ClOrdID (11) of the order being amended. Required if OrderID (37) is not specified.				
41	OrigClOrdID	N	ClOrdID (11) of the order being amended. Required if OrderID (37) is not specified. Will be ignored if OrderID is specified				
37	OrderID	N	Server specified identifier of the order being amended. Required if OrigClOrdID (41) is not specified.  This is an 11 character base 62 string with an 'O' prefix. After removal of the prefix, when converted to an 8 byte binary format, it will match the corresponding ITCH Order ID. This will be identical to the SecondaryOrderID when converted to a 16 character hexadecimal format				
<u>Component Block &lt;Trading Party&gt;</u>		Y	Identifier of the trading party.				
1	Account	N	Client reference information.				
55	Symbol	N	MTF Common Symbol.  Not required if 15, 48, 22 and 207 are specified.				
48	SecurityID	N	Identifier of the instrument.  Not required if 55 is specified.				
22	SecurityIDSource	N	Identifier of the source of the SecurityID (48) value.  <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>4</td> <td>ISIN</td> </tr> </tbody> </table>	Value	Meaning	4	ISIN
Value	Meaning						
4	ISIN						

15	Currency	N	Currency Code as per ISO 4217 Currency Code List  Not required if 55 is specified
207	SecurityExchange	N	Market Identifier Code as per ISO 10383  Not required if 55 is specified
9303	RoutingInst	Y	Indicates the liquidity pool.  <b>Value      Meaning</b> <hr/> I            Integrated order book <hr/> M            Dark Midpoint order book
18	ExecInst	N	Indicates if the order should participate only in the uncrossing. Applicable to the Dark Midpoint order book only. If unspecified the order will participate in both continuous and uncrossing.  <b>Value            Meaning</b> <hr/> x (default)      Continuous and Periodic un-crossings <hr/> z                    Periodic un-crossings only <hr/> y                    Continuous only
40	OrdType	Y	Must match the value in the order.
126	ExpireTime	N	Time the order expires which must be a time during the current trading day. If the client amends both ExpireTime (126) and ExpireDate (432) then the system will ignore the value set of ExpireDate (432). Required if TimInForce (59) is GTD (6) and ExpireDate (423) is not specified.
432	ExpireDate	N	<b>This is no longer supported by Turquoise.</b>  Date the order expires. Required if TimInForce (59) is GTD (6) and ExpireTime (126) is not specified.
54	Side	Y	Must match the value in the order.
38	OrderQty	Y	Total order quantity.

1138	DisplayQty	Y	Maximum quantity that may be displayed. It is mandatory to specify the intended display quantity.
1084	DisplayMethod	N	<p>Whether the order was a hidden order and if the order was randomized.</p> <p>Please note that, when amending a randomized iceberg order, the amend request must contain 3 on this field, even if the amend would be converting the order to a fully visible one.</p> <p>If the order is not a randomized iceberg order, it cannot be amended to be one.</p> <p>Enum 3 will be accepted on Cancel/Replace Request only if the original order contained 1084 = 3.</p> <p><b>Value    Meaning</b></p> <hr/> <p>4            Undisclosed (Hidden Order)</p> <hr/> <p>3            Random (randomize value)</p>
110	MinQty	N	Minimum fill size.
44	Price	N	Limit price. Required if OrderType (40) is Limit (2).
60	TransactTime	Y	Time the cancel/replace request was created.

27010	PassiveOnlyOrder	N	<p>Used to specify whether an order will rest prior to execution, with flexibility for visible orders to rest at a specified price level on the book.</p> <p>No protection is provided against order execution with large in scale hidden orders sat within the BBO.</p> <p>A hidden order will be rejected if it does not have a value of 0 or 99.</p> <p>A Dark Midpoint order will be rejected if it does not have a value of 0.</p> <table border="0"> <thead> <tr> <th data-bbox="628 757 699 786">Value</th> <th data-bbox="746 757 847 786">Meaning</th> </tr> </thead> <tbody> <tr> <td data-bbox="628 831 644 860">0</td> <td data-bbox="746 831 895 860">No constraint</td> </tr> <tr> <td colspan="2" data-bbox="628 898 719 927">(default)</td> </tr> <tr> <td data-bbox="628 972 660 1001">99</td> <td data-bbox="746 972 1050 1055">Accept order only if passive upon order entry. Otherwise expire.</td> </tr> <tr> <td data-bbox="628 1099 676 1128">100</td> <td data-bbox="746 1099 1098 1151">Accept order if setting new BBO. Otherwise expire.</td> </tr> <tr> <td data-bbox="628 1196 644 1225">1</td> <td data-bbox="746 1196 1091 1279">Accept order if setting new BBO or joining existing BBO. Otherwise expire.</td> </tr> <tr> <td data-bbox="628 1323 644 1352">2</td> <td data-bbox="746 1323 1075 1406">Accept order if joining existing BBO or within one visible price point. Otherwise expire.</td> </tr> <tr> <td data-bbox="628 1451 644 1480">3</td> <td data-bbox="746 1451 1075 1534">Accept order if joining existing BBO or within two visible price points. Otherwise expire.</td> </tr> </tbody> </table>	Value	Meaning	0	No constraint	(default)		99	Accept order only if passive upon order entry. Otherwise expire.	100	Accept order if setting new BBO. Otherwise expire.	1	Accept order if setting new BBO or joining existing BBO. Otherwise expire.	2	Accept order if joining existing BBO or within one visible price point. Otherwise expire.	3	Accept order if joining existing BBO or within two visible price points. Otherwise expire.
Value	Meaning																		
0	No constraint																		
(default)																			
99	Accept order only if passive upon order entry. Otherwise expire.																		
100	Accept order if setting new BBO. Otherwise expire.																		
1	Accept order if setting new BBO or joining existing BBO. Otherwise expire.																		
2	Accept order if joining existing BBO or within one visible price point. Otherwise expire.																		
3	Accept order if joining existing BBO or within two visible price points. Otherwise expire.																		
<a href="#">Standard Trailer</a>																			

## 6.4.5 Execution Report

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	8 = Execution Report
Message Body			
17	ExecID	Y	Server specified identifier of the message.
880	TradeMatchID	N	This is the unique identifier of the Trade. This is a base 62 encoded value in ASCII format.
11	ClOrdID	Y	Client specified identifier of the order.
41	OrigClOrdID	N	OrigClOrdID (41), if any, which was submitted with the order cancel or cancel/replace request.
37	OrderID	Y	Server specified identifier of the order.  This is an 11 character base 62 string with an 'O' prefix. After removal of the prefix, when converted to an 8 byte binary format, it will match the corresponding ITCH Order ID. This will be identical to the SecondaryOrderID when converted to a 16 character hexadecimal format
198	SecondaryOrderID	Y	Indicates the corresponding Market Data (MITCH) Order ID. This is 16 characters long, in the hexadecimal format. Since the ITCH order ID will be disseminated in binary format via the ITCH gateway, this hexadecimal value needs to be converted to the binary format to compare against it.



150	ExecType	Y	<p>Reason the execution report was generated.</p> <table border="1"> <thead> <tr> <th data-bbox="660 409 751 439">Value</th> <th data-bbox="788 409 895 439">Meaning</th> </tr> </thead> <tbody> <tr> <td data-bbox="660 479 687 508">0</td> <td data-bbox="788 479 842 508">New</td> </tr> <tr> <td data-bbox="660 548 687 577">4</td> <td data-bbox="788 548 903 577">Cancelled</td> </tr> <tr> <td data-bbox="660 618 687 647">5</td> <td data-bbox="788 618 898 647">Replaced</td> </tr> <tr> <td data-bbox="660 687 687 716">8</td> <td data-bbox="788 687 890 716">Rejected</td> </tr> <tr> <td data-bbox="660 757 687 786">C</td> <td data-bbox="788 757 874 786">Expired</td> </tr> <tr> <td data-bbox="660 826 687 855">D</td> <td data-bbox="788 826 890 855">Restated</td> </tr> <tr> <td data-bbox="660 896 687 925">F</td> <td data-bbox="788 896 858 925">Trade</td> </tr> <tr> <td data-bbox="660 965 687 994">H</td> <td data-bbox="788 965 938 994">Trade Cancel</td> </tr> <tr> <td></td> <td></td> </tr> </tbody> </table>	Value	Meaning	0	New	4	Cancelled	5	Replaced	8	Rejected	C	Expired	D	Restated	F	Trade	H	Trade Cancel		
Value	Meaning																						
0	New																						
4	Cancelled																						
5	Replaced																						
8	Rejected																						
C	Expired																						
D	Restated																						
F	Trade																						
H	Trade Cancel																						
19	ExecRefID	N	Reference to the execution being cancelled. Required if ExecType (150) is Trade Cancel (H).																				
378	Exec Restatement Reason	N	<p>Reason the order was restated. Required if ExecType (150) is Restated (D).</p> <table border="1"> <thead> <tr> <th data-bbox="660 1312 751 1341">Value</th> <th data-bbox="788 1312 895 1344">Meaning</th> </tr> </thead> <tbody> <tr> <td data-bbox="660 1382 687 1411">8</td> <td data-bbox="788 1382 938 1411">Market Option</td> </tr> </tbody> </table>	Value	Meaning	8	Market Option																
Value	Meaning																						
8	Market Option																						

39	OrdStatus	Y	<p>Current status of the order.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>New</td> </tr> <tr> <td>1</td> <td>Partially Filled</td> </tr> <tr> <td>2</td> <td>Filled</td> </tr> <tr> <td>4</td> <td>Cancelled</td> </tr> <tr> <td>8</td> <td>Rejected</td> </tr> <tr> <td>C</td> <td>Expired</td> </tr> </tbody> </table>	Value	Meaning	0	New	1	Partially Filled	2	Filled	4	Cancelled	8	Rejected	C	Expired
Value	Meaning																
0	New																
1	Partially Filled																
2	Filled																
4	Cancelled																
8	Rejected																
C	Expired																
103	OrdRejReason	N	Code specifying the reason for the reject. Please refer to section <a href="#">7.1.1 Execution Report</a> for a list of reject codes. Required if ExecType (150) is Rejected (8).														
58	Text	N	This field will carry the client specified value for the order except when ExecType(150) = 4, 8, C, D or H.														
32	LastQty	N	Quantity executed in this fill. Required if ExecType (150) is Trade (F).														
31	LastPx	N	Price of this fill. Required if ExecType (150) is Trade (F).														
151	LeavesQty	Y	Quantity available for further execution. Will be "0" if OrdStatus (39) is Filled (2), Cancelled (4), Rejected (8) or Expired (C).														
14	CumQty	Y	Total cumulative quantity filled.														
55	Symbol	N	MTF Common Symbol.														
48	SecurityID	N	Identifier of the instrument.														
22	SecurityIDSource	N	<p>Identifier of the source of the SecurityID (48) value.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>4</td> <td>ISIN</td> </tr> </tbody> </table>	Value	Meaning	4	ISIN										
Value	Meaning																
4	ISIN																

9303	RoutingInst	Y	<p>Indicate the liquidity pool.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>I</td> <td>Integrated order book</td> </tr> <tr> <td>M</td> <td>Dark Midpoint order book</td> </tr> </tbody> </table>	Value	Meaning	I	Integrated order book	M	Dark Midpoint order book		
Value	Meaning										
I	Integrated order book										
M	Dark Midpoint order book										
15	Currency	N	Currency Code as per ISO 4217 Currency Code List								
207	SecurityExchange	N	Market Identifier Code as per ISO 10383								
18	ExecInst	N	<p>Only applicable to the Dark Midpoint order book. Will be the value stamped on New Order/Order Cancel replace message. If not defined on new order message, will be the default applied according to order's owner's user setup.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>x</td> <td>Continuous and Periodic un-crossings</td> </tr> <tr> <td>z</td> <td>Periodic un-crossings only</td> </tr> <tr> <td>y</td> <td>Continuous only</td> </tr> </tbody> </table>	Value	Meaning	x	Continuous and Periodic un-crossings	z	Periodic un-crossings only	y	Continuous only
Value	Meaning										
x	Continuous and Periodic un-crossings										
z	Periodic un-crossings only										
y	Continuous only										
20000	TypeOfTrade	N	<p>Indicates whether the executed portion is visible or hidden. Required only if ExecType (150) = F - Trade.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Visible</td> </tr> <tr> <td>1</td> <td>Hidden</td> </tr> <tr> <td>2</td> <td>Not specified</td> </tr> </tbody> </table>	Value	Meaning	0	Visible	1	Hidden	2	Not specified
Value	Meaning										
0	Visible										
1	Hidden										
2	Not specified										
<u>Component Block &lt;Trading Party&gt;</u>		Y	Values specified in the order.								

9730	TradeLiquidityIndicator	N	<p>Whether the order added or removed liquidity.</p> <p>Required only for messages generated for a trade or trade cancellations. Possible values are:</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>Added Liquidity</td> </tr> <tr> <td>R</td> <td>Removed Liquidity</td> </tr> <tr> <td>C</td> <td>Periodic Uncrossing execution (valid for Dark orders only)</td> </tr> </tbody> </table>	Value	Meaning	A	Added Liquidity	R	Removed Liquidity	C	Periodic Uncrossing execution (valid for Dark orders only)
Value	Meaning										
A	Added Liquidity										
R	Removed Liquidity										
C	Periodic Uncrossing execution (valid for Dark orders only)										
1	Account	N	Value submitted with the order.								
40	OrdType	Y	Value submitted with the order.								
59	TimeInForce	N	Value submitted with the order.								
126	ExpireTime	N	Value submitted with the order.								
432	ExpireDate	N	Value submitted with the order.								
54	Side	Y	Value submitted with the order.								
38	OrderQty	Y	Value submitted with the order.								
1138	DisplayQty	Y	Quantity currently displayed in the order book.								
110	MinQty	N	Value submitted with the order.								
44	Price	N	Value submitted with the order.								
581	AccountType	Y	<p>Type of account associated with the order.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Client</td> </tr> <tr> <td>3</td> <td>House</td> </tr> </tbody> </table>	Value	Meaning	1	Client	3	House		
Value	Meaning										
1	Client										
3	House										

528	OrderCapacity	Y	Capacity of the order.  <table border="0"> <thead> <tr> <th style="text-align: left;">Value</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>Agency</td> </tr> <tr> <td>P</td> <td>Principal</td> </tr> <tr> <td>R</td> <td>Riskless Principal</td> </tr> </tbody> </table>	Value	Meaning	A	Agency	P	Principal	R	Riskless Principal
Value	Meaning										
A	Agency										
P	Principal										
R	Riskless Principal										
60	TransactTime	Y	Time the transaction represented by the Execution Report occurred.								
526	SecondaryCLOrdID	N	Value submitted with the order.								
583	CIOrdLinkID	N	Value submitted with the order.								
1094	PegPriceType	Y	Only applicable to Turquoise Dark book, will not be sent for Turquoise lit executions.  <table border="0"> <thead> <tr> <th style="text-align: left;">Value</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Midpoint</td> </tr> </tbody> </table>	Value	Meaning	0	Midpoint				
Value	Meaning										
0	Midpoint										
27010	PassiveOnlyOrder	N	Value submitted with the order.								

27011	PriceDifferential	N	<b>Value</b>	<b>Meaning</b>
			A	Aggressive. Any residual visible quantity is then stamped based on its deviation from the current BBO. Set to 'P' if hidden.
			B	New visible BBO.
			1	Join visible BBO.
			2	Joining/setting 2 <sup>nd</sup> best visible price point.
			3	Joining/setting 3 <sup>rd</sup> best visible price point.
			4	Joining/setting 4 <sup>th</sup> best visible price point.
			5	Joining/setting 5 <sup>th</sup> best visible price point.
			6	Joining/setting 6 <sup>th</sup> best visible price point.
			7	Joining/setting 7 <sup>th</sup> best visible price point.
			8	Joining/setting 8 <sup>th</sup> best visible price point.
9	Joining/setting 9 <sup>th</sup> best visible price point (or a worse price point).			
P	Passive. Valid for large in scale hidden orders only.			
278	MDEntryID	Y	Public Order ID	
<a href="#">Standard Trailer</a>				

## 6.4.6 Order Cancel Reject

Tag	Field Name	Req	Description														
<a href="#">Standard Header</a>																	
35	MsgType	Y	9 = Order Cancel Reject.														
Message Body																	
11	CIOrdID	Y	CIOrdID (11) that was submitted with the order cancel or cancel/replace request being rejected.														
41	OrigCIOrdID	N	OrigCIOrdID (41), if any, which was submitted with the order cancel or cancel/replace request being rejected.														
37	OrderID	Y	<p>Server specified identifier of the order for which the cancel or cancel/replace was submitted. Will be "NONE" if the order is unknown.</p> <p>This is an 11 character base 62 string with an 'O' prefix. After removal of the prefix, when converted to an 8 byte binary format, it will match the corresponding ITCH Order ID. This will be identical to the SecondaryOrderID when converted to a 16 character hexadecimal format</p>														
39	OrdStatus	Y	<p>Current status of the order. Will be Rejected (8) if the order is unknown.</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>New</td> </tr> <tr> <td>1</td> <td>Partially Filled</td> </tr> <tr> <td>2</td> <td>Filled</td> </tr> <tr> <td>4</td> <td>Cancelled</td> </tr> <tr> <td>8</td> <td>Rejected</td> </tr> <tr> <td>C</td> <td>Expired</td> </tr> </tbody> </table>	Value	Meaning	0	New	1	Partially Filled	2	Filled	4	Cancelled	8	Rejected	C	Expired
Value	Meaning																
0	New																
1	Partially Filled																
2	Filled																
4	Cancelled																
8	Rejected																
C	Expired																

434	CxlRej ResponseTo	Y	Type of request being rejected.  <b>Value    Meaning</b> <hr/> 1            Order Cancel Request <hr/> 2            Order Cancel/Replace Request
102	CxlRejReason	Y	Code specifying the reason for the rejection. Please refer to section <a href="#">7.1.2 Order Cancel Reject</a> for a list of reject codes.
58	Text	N	Text specifying the reason for the rejection.
<a href="#">Standard Trailer</a>			

#### 6.4.7        Order Mass Cancel Report

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	r = Order Mass Cancel Report
Message Body			
1369	MassActionReportID	Y	Server specified identifier of the message.
11	ClOrdID	Y	Client specified identifier of mass cancel request.
530	MassCancel RequestType	Y	Value specified in the mass cancel request.
531	MassCancel Response	Y	Action taken by server.  <b>Value    Meaning</b> <hr/> 0            Mass Cancel Request Rejected <hr/> 1            Cancelled All Orders for Instrument <hr/> 7            Cancelled All Orders <hr/> 9            Cancelled All Orders for Segment



532	MassCancelReject Reason	N	Code specifying the reason for the rejection. Refer to section <a href="#">7.1.3 Order Mass Cancel Report</a> for a list of reject codes. Required if MassCancelResponse (531) is Mass Cancel Request Rejected (0).								
1180	AppID	Y	Partition ID to which the Order Mass Cancel Report corresponds to.  <table border="1"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Partition 1</td> </tr> <tr> <td>2</td> <td>Partition 2</td> </tr> <tr> <td>3</td> <td>Partition 3</td> </tr> </tbody> </table>	Value	Meaning	1	Partition 1	2	Partition 2	3	Partition 3
Value	Meaning										
1	Partition 1										
2	Partition 2										
3	Partition 3										
<a href="#">Standard Trailer</a>											

## 6.5 Application Messages: Others

### 6.5.1 Business Message Reject

Tag	Field Name	Req	Description
<a href="#">Standard Header</a>			
35	MsgType	Y	j = Business Message Reject.
Message Body			
379	BusinessReject RefID	N	Client specified identifier (e.g. ClOrdID) of the rejected message if it is available.
45	RefSeqNum	Y	MsgSeqNum (34) of the rejected message.
372	RefMsgType	Y	MsgType (35) of the rejected message.
371	RefTagID	N	If a message is rejected due to an issue with a particular field its tag number will be indicated.
380	BusinessReject Reason	Y	Code specifying the reason for the reject. Refer to section <a href="#">7.2.2 Business Message Reject</a> for a list of reject codes.

58	Text	N	Text specifying the reason for the rejection.
<a href="#">Standard Trailer</a>			

## 6.6 Components of Application Messages

### 6.6.1 Trading Party

Tag	Field Name		Req	Description
453	NoPartyIDs		Y	Number of party identifiers. The value in this field can be "1", "2" or "3".
➔	448	PartyID	Y	Set to the MemberFirmID for internalised trade executions else set to the CCP name:  EMCF <hr/> EuroCCP <hr/> LCH <hr/> X-Clear
➔	447	PartyID Source	Y	<b>Value    Meaning</b> <hr/> D          Proprietary/Custom Code
➔	452	Party Role	Y	Role of the specified PartyID (448). It will be mandatory to have Party Role Trader Group (76) for the New Order – Single, Order Cancel Request and Order Cancel/Replace Request messages.  <b>Value    Meaning</b> <hr/> 12        Trader ID <hr/> 17        Counterparty Firm <hr/> 76        Trader Group

---

## 7.0 Reject Codes

### 7.1 Order Handling

#### 7.1.1 Execution Report

OrdRej Reason	Meaning
2	Exchange closed
5	Unknown order
6	Duplicate order (i.e. duplicate ClOrdID)
16	Price exceeds current price band
18	Invalid price increment
99	Other

#### 7.1.2 Order Cancel Reject

CxlRej Reason	Meaning
0	Too late to cancel (i.e. order already filled)
1	Order not found (too late to cancel or unknown)
6	Duplicate ClOrdID
8	Price exceeds current price band
18	Invalid price increment
99	Other

#### 7.1.3 Order Mass Cancel Report

Mass Cancel Reject Reason	Meaning
1	Unknown instrument
99	Other
100	Unknown executing member
101	Unknown trading party

## 7.2 Others

### 7.2.1 Reject

Session Reject Reason	Meaning
1	Required tag missing
2	Invalid tag
4	Tag specified without a value
5	Value is out of range for this tag
6	Incorrect data format for value
9	CompID problem
11	Invalid Message Type
13	Tag appears more than once
14	Tag specified out of required order
15	Repeating group fields out of order
18	Invalid or unsupported application version
99	Other

### 7.2.2 Business Message Reject

Business Reject Reason	Meaning
0	Other
0	Message rate exceeded
0	Trader Group not specified on message
2	Unknown Security ID
3	Unsupported Message Type
4	Application not available
5	Conditionally required field missing

## 8.0 Appendix A

### 8.1 Order routing logic if RoutingInst (9303) not specified

Orders without a RoutingInst (9303) will be sent to the Integrated and Dark Midpoint order book based on values specified for TIF (59), OrdType (40), DisplayQty (1138), Price (44), OrderQty (38), MinQty (110).

#### DisplayQty = 0

- With one exception, any order with DisplayQty=0 (irrespective of OrdType, TIF, MinQty) will be routed to the Dark Midpoint Book.
- The exception is for Limit orders with a MinQty of Zero or Null, a TIF of DAY/GTC/GTD/GTT, and a value (OrderQty x Price) greater than the LIS threshold – these will be routed to the Integrated order book as a hidden order.

#### DisplayQty <> 0 (including Null values)

- Any order with OrdType=Peg will be routed to the Dark Midpoint order book
- All other orders will be routed to the Integrated order book

The below matrix describes the order routing logic if RoutingInst (9303) is not specified on the **New Order Single** message.

( RoutingInst (9303) is a mandatory tag for **Order Cancel Request** and **Order Cancel/Replace Request** messages. )

#	TIF (59)	OrdType (40)	DisplayQty (1138)	Price (44) x OrderQty (38)	MinQty (110)	Destination	
1	DAY, GTC, GTD, GTT GFA	Limit	0	< LIS	> 0	Midpoint	
2					0 or Null	Midpoint	
3				>= LIS	> 0	Midpoint	
4					0 or Null	Integrated**, as hidden order	
5			> 0		< LIS	> 0	Rejected
6						0 or Null	Integrated**
7					>= LIS	> 0	Rejected

#	TIF (59)	OrdType (40)	DisplayQty (1138)	Price (44) x OrderQty (38)	MinQty (110)	Destination
8					0 or Null	Integrated**
9			Null	< LIS	> 0	Rejected
10					0 or Null	Integrated**
11				>= LIS	> 0	Rejected
12					0 or Null	Integrated**
13		Market	0	< LIS	> 0	Midpoint
14					0 or Null	Midpoint
15				>= LIS	> 0	Midpoint
16					0 or Null	Midpoint
17			> 0	< LIS	> 0	Integrated**
18					0 or Null	Integrated**
19				>= LIS	> 0	Integrated**
20					0 or Null	Integrated**
21			Null	< LIS	> 0	Integrated**
22					0 or Null	Integrated**
23				>= LIS	> 0	Integrated**
24					0 or Null	Integrated**
26		Peg	0	< LIS	> 0	Midpoint
27					0 or Null	Midpoint
28				>= LIS	> 0	Midpoint
29					0 or Null	Midpoint
30			> 0	< LIS	> 0	Rejected*
31					0 or Null	Rejected*
32				>= LIS	> 0	Rejected*
33					0 or Null	Rejected*
34			Null	< LIS	> 0	Midpoint
35					0 or Null	Midpoint

#	TIF (59)	OrdType (40)	DisplayQty (1138)	Price (44) x OrderQty (38)	MinQty (110)	Destination
36				>= LIS	> 0	Midpoint
37					0 or Null	Midpoint
38	IOC, FOK	Any (Limit, Market, Peg)	0	< LIS	> 0	Midpoint
39					0 or Null	Midpoint
40				>= LIS	> 0	Midpoint
41				0 or Null	Midpoint	
42			> 0	< LIS	> 0	Integrated
43					0 or Null	Integrated
44				>= LIS	> 0	Integrated
45				0 or Null	Integrated	
46			Null	< LIS	> 0	Integrated
47					0 or Null	Integrated
48				>= LIS	> 0	Integrated
49				0 or Null	Integrated	

\*Order will be routed to the Midpoint book and will be rejected by matching engine since Midpoint book does not accept orders with disclosed quantity.

\*\* Order will be rejected if Time In Force (TIF) = Good For Auction (GFA)



## 9.0 Appendix B

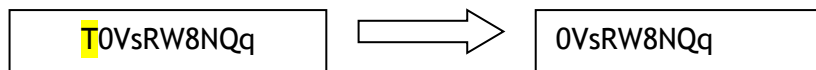
### 9.1 Converting FIX TradeMatchID (880) to ITCH TradeMatchID

#### Worked Example

<b>TradeMatchID (880) in FIX</b> (ASCII base 62 characters with 'T' prefix)	T0VsRW8NQq
<b>Same TradeMatchID in ITCH gateway</b> (Binary ID converted to decimal)	112262475939900

#### Steps to follow

1. Remove prefix 'T'



2. Convert using base 62 conversion in to decimal as depicted below

Note: Please refer to the base 62 conversion table attached below

Ascii Character	Corresponding decimal value	Multiplier		Multiplied decimal value
		62 <sup>x</sup>	value	
q	52	62 <sup>0</sup>	1	52
Q	26	62 <sup>1</sup>	62	1,612
N	23	62 <sup>2</sup>	3,844	88,412
8	8	62 <sup>3</sup>	238,328	1,906,624
W	32	62 <sup>4</sup>	14,776,336	472,842,752

R	27	62^5	916,132,832	24,735,586,464
s	54	62^6	56,800,235,584	3,067,212,721,536
V	31	62^7	3,521,614,606,208	109,170,052,792,448
0	0	62^8	218,340,105,584,896	-
Decimal value of the TradeMatchID generated in ITCH				<b>112,262,475,939,900</b>

### Base 62 Conversion Table

Decimal	ASCII	Decimal	ASCII	Decimal	ASCII	Decimal	ASCII
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		

---

## 10.0 Appendix C

### 10.1 Error & Reject Messages

Text(58)

Attached Instrument Is not Dark

Cannot amend Account Type

Cannot amend Capacity

Expired (end of day)

Failed maximum order value validation

Failed price band validation

Incorrect data format for value

Incorrect NumInGroup count for repeating group

Instrument halted

Instrument halted (invalid order book set up)

Instrument halted (invalid set up)

Instrument halted (invalid trading session)

Instrument halted (last trading day reached)

Instrument halted (market suspended)

Instrument halted (order book in invalid state)

Instrument halted or suspended

Instrument Setup Error - Invalid Order Book

Invalid account type (unknown)

Invalid amend (cannot amend MES)

---

---

Invalid amend (cannot amend order type)

Invalid amend (cannot amend TIF)

Invalid amend (iceberg/fully visible to hidden)

Invalid amend (hidden to iceberg/fully visible)

Invalid amend order message

Invalid Book Target Book in the Received Order

Invalid cancel order message

Invalid capacity (unknown)

Invalid clearing set up (clearing information not defined)

Invalid clearing set up (clearing information not defined)

Invalid display quantity

Invalid display quantity (> zero)

Invalid display quantity (greater than order quantity)

Invalid display quantity (greater than order quantity)

Invalid display size (< minimum disclosed size)

Invalid display size (< zero)

Invalid display size (> order size)

Invalid display size (not multiple of lot size)

Invalid display size (pegged orders cannot be displayed)

Invalid expire date (elapsed)

Invalid expire time (elapsed)

Invalid expiry date (maximum order duration is violated)

Invalid instrument set up (no tick structure)

---

Invalid limit price (< minimum price)

Invalid limit price (> maximum price)

Invalid limit price ( $\leq$  zero or no limit price)

Invalid limit price (not multiple of tick)

Invalid limit price (price band breached)

Invalid mass cancel type

Invalid MES (> order size)

Invalid MES (less than minimum size)

Invalid MES (negative)

Invalid MES (not a multiple of lot size)

Invalid minimum size (> order size)

Invalid minimum size (not multiple of lot size)

Invalid MsgType

Invalid new order message

Invalid Order Qualifier

Invalid order quantity

Invalid order quantity (less than filled quantity)

Invalid order size (< minimum size)

Invalid order size ( $\leq$  zero)

Invalid order size (not multiple of lot size)

Invalid order status (%d)

Invalid Order Sub Type

Invalid order type (named orders are not allowed)

---

Invalid order type (not allowed in the session)

Invalid order type (pegged orders cannot be stop orders)

Invalid order type (stop/stop limit orders are not allowed)

Invalid order type (unknown)

Invalid owner (different from original order)

Invalid Passive Only Indicator for Hidden Order

Invalid reserve value (< minimum reserve order value)

Invalid Security ID Source

Invalid session (aggressive orders are not allowed)

Invalid session (cannot cancel/amend orders/quotes)

Invalid session (cannot enter orders/quotes)

Invalid session (orders are not allowed)

Invalid side

Invalid side (different from original order)

Invalid tag

Invalid TIF (invalid date format)

Invalid TIF (maximum order duration is set)

Invalid TIF (not allowed for stop/stop limit orders)

Invalid TIF (not allowed for the session)

Invalid TIF (not permitted for pegged orders)

Invalid TIF (relevant session elapsed/not found)

Invalid TIF (unknown)

Invalid trading session (unknown)

---

Invalid Routing Instruction

ISIN/CCY /MIC provided is inconsistent with MTF Symbol

Last Trading Date of instrument elapsed

Last Trading Date of instrument elapsed

market is closed

MarketSegmentID required

Matching partition suspended

MES not allowed for persistent orders

MES Should be greater than the Minimum Size of the Book

Minimum Quantity Cannot be negative

No orders for instrument/underlying

No time qualifier specified

Non Persistent Orders Can not be Periodic Uncrossings only

Order not found (too late to cancel or unknown order)

Order rejection – PB Dynamic Tolerance Threshold breached

Order rejection - PB Static Tolerance Threshold breached

Order Value Cannot exceed the maximum value

Orders with minimum size not permitted during Pre-Open

OrigClOrdID or OrderID required

Other

PartyID required

PartyIDSource required

PartyRole required

---

Passive Only Orders Disabled for Instrument

Passive Only Indicator not Valid for TIF

Price unset for limit order

Quotes not allowed

Received Prior to First Trading Date of instrument

Received Prior to First Trading Date of instrument

Repeating group fields out of order

Repeating group fields out of order

Required tag missing

SecurityID required

SecurityIDSource required

Session is closed

StopPx unset for stop order

System suspended

Tag appears more than once

Tag contains non-numeric character

Tag not defined for this message type

Tag specified without a value

Trader Group not specified on message

Unknown Account Type

Unknown clearing mnemonic

Unknown Execution Instruction

Unknown firm

---



Unknown instrument

Unknown order book

Unknown SecurityID

Unknown SecurityID

Unknown segment

Unknown underlying

Unknown user (submitting Trader ID)

Unknown user (target Owner ID)

Unknown user (target Trader ID)

Unsupported Message Type

User not registered to manage interest for %s

User not registered to manage interest for instrument

User not registered to mass cancel interest

User not registered to mass cancel interest for firm

User not registered to submit interest for %s

User not registered to submit interest for instrument

Value is incorrect (out of range) for this tag

Value is out of range for this tag

Your account is Inactive

Your account is Suspended

---

Copyright © September 2013 Turquoise Global Holdings Limited.  
Registered in England and Wales No. 07102717.

Turquoise Global Holdings Limited has used all reasonable efforts to ensure that the information contained in this publication is correct at the time of going to press, but shall not be liable for decisions made in reliance on it.

London Stock Exchange is a registered trade mark of London Stock Exchange plc. Turquoise is a registered trade mark of Turquoise Trading Limited

**Turquoise Services  
Limited/Turquoise Trading Limited**

10 Paternoster Square  
London EC4M 7LS  
T: +44 (0)20 7797 1000

Company Details

