Turquoise

Turquoise

TQ202 - Post Trade Gateway (FIX 5.0)

Issue 3.5.7

22 May 2019

London Stock Exchange Group

Contents

1.0 Introduction – TQ202 Post Trade 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**.**

Turquoise offers an interface to its post trade system that permits participants to perform the activities outlined below:

(i) Receive real-time updates on executed trades

(ii) Receive information on executed trades via a query-based service. This service is available for participants that do not require a real-time trade feed and to facilitate a recovery after a failure.

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 post trade gateway available on Turquoise.

1.2 Readership

This document outlines how to connect to the post trade 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 which can be found on the Turquoise website here 'Document Library'.

Interfaces and information dissemination

For further information regarding Turquoise connectivity, trading and subscription to market data, please refer to the following documentation:

- TQ102 – Connectivity Guide

- TQ103 – Trading Technical Parameters
- TQ201 – Trading Gateway (FIX5.0) Specification
- **TQ202 – Post Trade Gateway (FIX5.0) Specification (this document)**
- TQ203 – Drop Copy Gateway (FIX5.0) Specification
- TQ301 – Trading Gateway (Native) Specification
- TQ501 – Guide to Reference Data Services
- TQ502 – Guide to Purchase and Sales File

Certification and Testing Services

For further information regarding Certification of Participant's software and ongoing testing obligations with Turquoise, please refer to the following documentation:

- TQ601 – Guide to Certification
- TQ602 – Certification Report
- TQ603 – Guide to Testing Services

LSEG Group Ticker Plant

For further information regarding subscription to Turquoise market data from the Group Ticker Plant (GTP), please refer to the following documentation which can be found on the GTP website here 'GTP Documentation Library':

- GTP001 – Product Guide
- GTP002 – Technical Guide
- GTP003 – Statistics Guide
- GTP004 – Parameters Guide
- GTP005 – Testing Service Guide
- GTP006 – External Source Guide
- GTP008 – Market Attributes Guide

1.4 Document History

This document has been through the follow iterations:

Issue	Date	Description
R1 1.0	17 Mar 2010	First issue of this document published.
R2 1.0	24 May 2010	First issue of CDS release 2 document published.
R2.1 1.0	09 Jul 2010	First issue of CDS release 2 document published.
R2.1 1.3	13 Aug 2010	Issue 1.3, Release 2.1 published.
R2.1 1.4	16 Sep 2010	Issue 1.4, Release 2.1 published.

1.6	13 May 2011	Issue 1.6 8.1.1 – Included DeliverToCompID and OnBehalfOfCompID to message header details. 4.1 / 4.4 / 7.3.1 Included details of Test Request sent at Login. Added clarification to SideLiquidityInd field.
1.7	5 July 2011	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.
1.8	2 August 2011	Updated section 7.3.1 to add/remove fields that pertain to Off-Book Reporting in the Trade Capture Report message.
1.9	31 October 2011	Support for clearing interoperability.
2.0	27 April 2012	Section 7.3.1 – changed description for ClOrdID (11), added tags 9050 (ClientGroup), 9051 (CounterpartyGroup) and 1094 (PegPriceType).
2.1	31 August 2012	Section 7.3.1 – added value of '4' to SideLiquidityInd for executions generated during Periodic Auction uncross, added PriceDifferential field.
2.2	13 February 2013	Updated contact details.
2.3	20 September 2013	The following sections have been updated, 2.1; 2.1.3; 2.1.3.2; 2.2.1; 2.2.2.7; 2.2.4; 2.6.1.1; 2.6.1.3; 2.6.1.4; 2.6.2.1; 6.0; 7.1.1; 7.2.2; 7.3.1; 7.3.2; 7.3.5; 7.3.6.
2.4	24 October 2013	Rebranding of the Turquoise random periodic uncrossing to **Turquoise Plato Uncross ™**.
3.0	20 October 2014	Addition of **Turquoise Plato Block Discovery**™ messages. The following section has been updated; 7.3.1. Updated terms Client, User & Parties (where appropriate) to "Participant".
3.1	24 October 2014	Changed reference of ITCH to MITCH.
3.2	16 January 2015	This document has been updated to reflect changes for Millennium 8.6 upgrade. <u>Change Highlights:</u> • TradeMatchId – changing from base 62 to base

		36 • Tag 55 (Symbol) – changing from 6 to 8 characters Sections 2.2.2.2; 7.3.1; 7.3.2 and 7.3.5. See TQ700 – Release 8.6 Message Guidelines for full details on all changes.
3.3	15 July 2016	This document has been corrected to remove all references to client on book trade cancellations and off book trade report submissions as this functionality is not currently available in Turquoise**.** This includes the removal of the client initiated Trade Capture Report message and the server acknowledgement for that request. The following sections have been updated to aid clarity and reflect changes for the Millennium 9.0 upgrade: 2.1.3 – Clarified Trade Capture Report Request behaviour. 2.2.2.2, 7.31 – Clarified that we use a G offset for encoding and decoding base 36 values. 2.2.3 – Clarified Information on Billing. 2.4 – Clarified Trade Corrections. 3.4 – Clarified Connectivity Policy. 3.5 – Clarified Message Rate throttling behaviour. 4.1 – Clarified connection behaviour when additional messages are sent prior to the exchange of Logons. Clarified that we no longer send a reject message when receiving a second connection attempt whilst a user is already logged in. Clarified server behaviour for inbound message sequence. Clarified rapid login/logout safety mechanism. 4.2.2 – Clarified Heartbeats behaviour. 5.5 – Removed section on resending trade capture reports. 7.0 – Clarified what happens when an undefined tag is sent along with Administrative and Application messages. 7.3.1 – Removed PriceDifferential tag. 9.0 – Clarified Turquoise availability times. 10.1 – Removed Appendix A 'Error and Reject Messages'.
3.4	26 October 2016	Updated Turquoise to **Turquoise Plato™** where appropriate for **Turquoise Plato™** Order Book and **Turquoise Plato Block Discovery™** services, and updated Turquoise to Turquoise where appropriate.

		The following sections have been amended to aid clarity and also to reflect the changes introduced in Millennium 9.1 upgrade: 2.2.2.2 – Clarified TradeID and TradeMatchID values and tags for the different gateways. 4.1 – Clarified Establishing a connection behaviour. 7.2.1 – Added missing enum 2 and added new enum 3 to SessionStatus tag. 7.3.1 – Clarified SideLiquidityInd (1444); **Turquoise Plato Uncross™** execution is now enum 10 instead of enum 4.
3.5	07 April 2017	The following sections have been amended to aid clarity and also to reflect the changes introduced in the Millennium 9.2 (MiFID II compliant) upgrade: 2.1.3 – Added a link to the Trading Technical Parameters document. 2.2.1, 2.7.4 – Clarified Party Identification behaviour. 2.5, 2.7.1 – Clarified Timestamps and dates behaviour. 2.7.3, 7.3.1 – Clarified Order Capacities behaviour. 7.3.1 – Removed SettlDate (64) from the message. 7.3.1 – Clarified PartyID (448), PartyIDSource (447), PartyRole (452) behaviour and added new enums. Added PartyRoleQualifier (2376), AlgorithmicTradeIndicator (2667) and OrigTradeDate (1125) tags. Added OrderOrigination (1724) repeating group. Added MarketID (1301) and NoTrdRegPublications (2668) repeating group. Renamed all instances of enum 12 to 100 for Trader ID in PartyRole (452).
3.5.1	22 August 2017	The following sections have been amended to aid clarity: 7 – Removed Reject Code section since TQ801 has all the applicable Reject reasons and codes. 7.2.6 – Clarified SessionRejectReason behaviour Updated all references of **Turquoise®** to Turquoise**.**

3.5.2	8 September 2017	The following sections have been amended to remove references to off-book trades: 2.2.4, 2.4 –Reference to off-book is removed 7.3.1 – Tags applicable to off-book trade are removed: TradeReportType (856), TradePublishIndicator (1390), TradeHandlingInstr (1123), TrdSubTypeTrdSubType(829), OrigTradeHandlingInstr (1124)
3.5.2.A	25 September 2017	1.3 Document Series updated. 2.2.3 updated to reference **Turquoise Lit Auctions™**. 7.3.1 Tag 574 MatchType updated to reference **Turquoise Lit Auctions™**. 7.3.1 Tag 1444 SideLiquidityInd enum 4 – **Turquoise Lit Auctions™** Execution added. 7.3.1 Tag 1301 MarketID, TRQA for **Turquoise Lit Auctions™** added. 7.3.1 Tag 1094 PegPriceType updated to include **Turquoise Lit Auctions™**. 7.3.1 Clarified that EMCF is EuroCCP 7.3.2 Removed section – as participant-initiated TCR is not applicable to Turquoise.
3.5.3 A	2 October 2017	7.3.1 Tags TradeReportType (856), TradeHandlingInstr (1123) have been added back
3.5.4.A	13 November 2017	7.3.2 Trade Capture Report Request – Participant – Initiated has been added back. It has been mistakenly deleted in the version 3.5.2.A. All following sections have been re-numbered.
3.5.5	13 July 2018	7.3.1 Value in PartyID(448) changed from ‘EMCF’ to ‘ECCP’ 7.3.2 Vaue in PartyID(448) changed from ‘EMCF’ to ‘ECCP’
3.5.6	31 August 2018	7.3.1 Addition of new CCP ‘LCH SA’ 7.3.2 Addition of new CCP ‘LCH SA’
3.5.7	22 May 2019	From the version onwards, the changes will be highlighted in red instead of using the side bars. 1.3 - Reference to TQ401 – MITCH Level-2 Market Data Specification was removed 2.2.2.2, 7.3.1 – TVTIC Field - MITCH Trade Match ID is

		replaced with GTP Trade ID.

In subsequent issues, where amendments have been made to the previous version, these changes will be highlighted in red.

1.5 Enquiries

Please contact either the Technical Account Management Team or your Technical Account Manager if you have any questions about the Millennium Exchange services outlined in this document: Client Technology Services (UK) can be contacted at:

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

2.0 Service Description

Participants will receive real-time information on the trades executed on Turquoise along with notifications of any trade bust. The details of trades executed on previous trading days are not available via this service.

2.1 Connection Configuration

A member firm connection will be configured by Turquoise to receive all of its trades. The connection of a service bureau will be configured to receive the trades for all the firms it serves. A clearing firm connection will be configured to receive the trades it will clear. If required, a firm, service bureau or clearing firm connection could be configured to only receive trades for selected Participants (trader groups) and instruments. Each connection will be configured as either a real-time Participant or a query-based Participant or both.

For the purpose of redundancy, the service supports the configuration of multiple post trade connections to send the same information on the activity of the selected firms/mnemonics.

2.1.1 Real-Time Connections

A real-time Participant will receive the details of each eligible trade immediately after it is executed. Refer to sections 2.1.3 Trade Capture Report Requests, 5.4.4 Requesting Retransmission of Missed Trades and 5.4.6 Disconnection Prior to Completion of Retransmission for a description of how the trades executed during the time a real-time Participant is disconnected from the server may be recovered.

2.1.2 Query-Based Service

A query-based Participant will not receive any real-time notifications of its eligible trades. Such Participants are expected to request the server for the details of trades as and when they are needed as outlined in section 2.1.3 Trade Capture Report Requests.

2.1.3 Trade Capture Report Requests

A Participant may use the Trade Capture Report Request message to request the details of all eligible trades or those that meet certain criteria. The server will respond with a Trade Capture Report Request Ack to indicate, via the TradeRequestStatus (750) and TradeRequestResult (749) fields, whether the request is successful or not.

If the Trade Capture Report Request message is rejected due to it containing incorrect/invalid validations, a Reject message will also be generated as the response.

The total number of Trade Capture Report Requests that a Participant may send is limited each day to a configurable value defined by Turquoise**.** This limit is defined in the TQ103 - Trading Technical Parameters document. However, a Participant may request Turquoise to reset its request count.

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

If a request is accepted, the number of Trade Capture Reports that will be sent in response to the request will be indicated in the TotNumTradeReports (748) field of the Trade Capture Report Request Ack. The server will transmit the requested Trade Capture Reports immediately

after the Trade Capture Report Request Ack. Each Trade Capture Report will include the TradeRequestID (568) of the request it is sent in response to. The last Trade Capture Report will include a LastRptRequested (912) of Last Message (Y).

When serving a Trade Capture Report Request that affects multiple partitions, the original sequence of TCRs is only guaranteed within the results from each partition.

If a request is rejected, the reason will be specified in the field TradeRequestResult (749) of the Trade Capture Report Request Ack.

The Trade Capture Report Request message can only be used to request a snapshot of the current eligible trades. It cannot be used to subscribe to Trade Capture Reports.

2.1.3.1 Request for All Trades

The Trade Capture Report Request should include a TradeRequestType (569) of All Trades (0) if the Participant wishes to request the details of all eligible trades.

2.1.3.2 Request for Selected Trades

The Trade Capture Report Request should include a TradeRequestType (569) of Trades Matching Specified Criteria (1) if the Participant wishes to request the details of eligible trades for a selected set of instruments or for a specified party, trade type or Order.

The MTF Common Symbol (55) field or SecurityID (48), SecurityIDSource (22), Currency (15), SecurityExchange (207) fields of the Trade Capture Report Request may be used if the request relates to a single instrument.

The TrdType (828) field of the Trade Capture Report Request may be used if the request relates to either all trades or only off Book trade reports.

The Participant may also use the PartyID block of the message to indicate the parties (i.e. executing firm, trader group), if any, the request relates to. A Trade Capture Report Request may include multiple PartyIDs.

In addition, the Side (54) field of the Trade Capture Report Request may be used if the Participant only wishes trades for a particular side (i.e. buy or sell).

The Participant may also use the ClOrdID (11) or OrderID (37) field if the request is for trades related to a specific Order.

The ExecType (150) field of the message may be used if the request is limited to cancelled trades or trades that have not been cancelled.

If a particular Trade Capture Report Request contains multiple criteria (e.g. symbol and side), the server will treat it as a request for trades that match all of the specified criteria. If no trades match the specified criteria, the server will reject the request with a TradeRequestResult (749) of Cannot Match Selection Criteria (100).

2.2 Trade Information

The FIX Trade Capture Report message is utilised by the server to transmit the details of each trade. A separate Trade Capture Report will be sent for each side of a trade. In the case of a trade, TradeReportTransType (487) will be New (0) and TradeReportType (856) will be Submit (0).

Each message will contain both basic and value added information on the trade (e.g. price quantity, consideration), the security (e.g. MTF Common Symbol, ISIN, Currency, Security

Exchange, etc.) and the parties (e.g. Executing Firm, Counterparty etc.). It will also contain information related to the computation of execution fees (e.g. maker or taker).

2.2.1 Party Identifiers

ID	Description	PartyRole (452)
Executing Firm	The trading firm the executed order was submitted under.	1
Trading Group	The unit of the firm the executed order was submitted under. Must be specified when reporting a trade.	76
Counterparty	Identifier of the counterparty firm in a trade.	17
Trader ID	Trader ID of the trader who executed the trade.	100
Clearing Organisation	Clearing member for the particular trade.	21
Entering Firm	The firm which entered the trade.	7
Executing trader	Identifier of the Executing trader relevant to the order	12
Client ID	Identifier of the client of the order	3
Investment Decision Maker	Identifier of the investment decision relevant to the order	122

2.2.2 Trade, Execution and Order Identifiers

2.2.2.1 Trade Report ID

The TradeReportID (571) of each Trade Capture Report is unique across trading days. The Trade Capture Reports published to report the two sides of a trade will contain different TradeReportIDs. A Trade Capture Report published to notify a Participant of a trade bust includes the TradeReportID of the message that was published to report the trade in the TradeReportRefID (572) field.

2.2.2.2 Trade ID (TVTIC)

The Trade Capture Reports published to report the two sides of a particular trade will contain the same TradeID (1003). Trade IDs are unique across trading days. A Trade Capture Report published to notify a Participant of a trade bust includes the TradeID (1003) of the relevant trade.

The TradeID (1003) published by the system is a base 36 (G offset) string that refers to the same value published as Trade ID via GTP Market Data Gateway and Trade Match ID via Native Order Management Gateway. Additionally the TradeMatchID of an Execution Report message published via FIX 5.0 SP2 Order Management and Drop Copy gateways' will also refer to the TradeID(1003) of the Trade Capture Report.

TradeID(1003) of the Trade Capture Report refers to the Execution ID reported to the CCPs with the exception of buy/sell indicators and trade modification indicators being present as prefixes.

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

Post Trade	FIX Trading	Native Trading	Drop Copy	GTP Trade ID
TradeID (1003)	TrdMatchID (880)	Trade Match ID	TrdMatchID (880))	Trade ID
G5DIF33YV0	G5DIF33YV0	73120274710544	G5DIF33YV0	73120274710544

2.2.2.3 Execution ID

A Trade Capture Report will contain the Execution ID of the Execution Report message sent by the Order Management Gateway to report the execution of an Order to the firm that submitted it. This Execution ID will be specified in the SideExecID (1427) field of the Trade Capture Report.

The Execution Reports published to report the two sides of an execution will contain different Execution IDs which are unique across trading days.

2.2.2.4 Trade Link ID

A Trade Capture Report will contain the TradeLinkID (820) which will be the same for all Orders within aggression of an Order.

If non-binary reporting is enabled for a particular instrument, then the aggregated Trade Capture Report and the corresponding non-aggregated Trade Capture Reports will have the same TradeLinkID (820). If the same trade was to be reported under binary reporting, all Trade Capture Reports will have the same TradeLinkID (820).

All the Trade Capture Reports which were generated from an auction will have the same Trade Link ID.

2.2.2.5 Order IDs

The matching system's Order identification number for the executed Order will be included in the OrderID (37) field of the Trade Capture Report.

Order IDs are unique across trading days. 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.

2.2.2.6 Client Order IDs

In the case of Orders, the ClOrdID (11) included in the Trade Capture Report will be that specified when the Order was submitted. An Order's ClOrdID (11) will be updated each time an Order Cancel/Replace Request or an Order Cancel Request is accepted.

2.2.2.7 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 – ClOrdID (11)
- OrderCancelRequest – OriginalClOrdID (41)
- NewOrderSingle – SecondaryClOrdID (526)
- NewOrderSingle – ClOrdLinkID (583)

2.2.3 Information for Billing

Each Trade Capture Report will specify whether an Order executed normally in continuous trading was a maker or taker of liquidity via the SideLiquidityInd (1444) field. For the **Turquoise Plato™** Order Book this field is also used to classify **Turquoise Plato Block Discovery™** executions and executions in the **Turquoise Plato Uncross™**. It is also used to identify **Turquoise Lit Auctions™** Order Book executions.

2.2.4 Trade Type

An indication of whether the trade was executed on or off the Order Book will be specified in the TrdType (828) field of a Trade Capture Report.

2.3 Trade Cancellations

Cancellations of Trades cannot be made by Participants at any time.

Turquoise may in certain circumstances, including if requested by a Participant, intervene and cancel a Trade.

In the case of an on-Book trade cancel by Market Operations, the server will transmit Trade Capture Reports to the relevant Participants to notify them of a trade cancellation.

The cancelled trade will be identified in the TradeReportRefID (572) and TradeID (1003) fields.

TradeReportTransType (487) will be Cancel (1) and TradeReportType (856) will be Trade Break (7).

2.4 Trade Corrections

Turquoise will never amend trades.

2.5 Timestamps and Dates

The matrix below clarifies the expectations for timestamps and dates.

Fix Tag	Client Generated tag– accepted format	Server Generated format – sent format
SendingTime (52)	UTC,	UTC,

OrigSendingTime (122)	YYYMMDD-HH:MM:SS.uuuuuu and YYYYMMDD-HH:MM:SS.sss	YYYYMMDD-HH:MM:SS.uuuuuu
TransactTime (60)		
OrigTradeDate (1125)	N/A	

2.6 Encryption

The encryption of messages between the Participant and server is not supported.

2.7 MiFID II changes

2.7.1 Timestamping at Microsecond granularity

All server generated timestamps will now be in microsecond granularity. It is not mandatory for client generated timestamps to be in microsecond granularity. Further details are described in the Timestamps and dates section.

2.7.2 Pre-trade Waiver Flags

Pre-trade Waiver Flags have been added to the server generated Trade Capture Report.

2.7.3 Order capacities

The changes to Order capacities are shown below.

Pre-MiFID II name	MiFID II name
Principal	Dealing on own account (DEAL)
Agency	Any other trading capacity (AOTC)
Riskless Principal	N/A
N/A	Matched Principal (MTCH)

Until MiFID II go-live, tag OrderCapacity(528) = R will be treated as Riskless Principal. After MiFID II go-live, it will be treated as Matched Principal (MTCH).

Pre-trade Waiver Flags have been added to the server generated Trade Capture Reports.

2.7.4 Order Record Keeping Information

The existing Party identification tags will be used to capture data on Client ID, Investment decision within firm and Execution within firm. Refer to the Party identifiers section for details about the new tags introduced.

FIX tags have been introduced to the server generated Trade Capture Report to capture additional order attributes, such as if the order submitted was part of DEA involvement (Sponsored Access or DMA), if it was generated via an algorithm and if it was part of a market making strategy. Here are details of the tags:

- OrderOrigination (1724) – the same value that was submitted in the order.
- OrderAttributeType (2594) – the same value as submitted in the order. This tag is part of the NoOrderAttributes repeating group, and will have the value '2 - 'Liquidity Provision'.
- AlgorithmicTradeIndicator (2667) – will be set to '1 -True' if OrderAttributeType (2594) = '4 - 'Algorithm' is submitted in the order.

3.0 Connectivity

3.1 CompIDs

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

The gateway server will be assigned a CompID. The messages sent to the server should contain the CompID assigned to the Participant in the field SenderCompID (49) and CompID of the server in the field TargetCompID (56). The messages sent from the server to the Participant will contain CompID of the server in the field SenderCompID (49) and the CompID assigned to the Participant in the field TargetCompID (56).

3.1.1 Passwords

Each new CompID will be assigned a password on registration. Participants 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 message sent by the server to confirm the establishment of a FIX connection. The new password will, if accepted, be effective for subsequent logins.

3.2 Production IP Address and Ports

The IP addresses and ports for the post trade gateway will be published in the Turquoise Connectivity Guide.

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

3.4 Connectivity Policy

An application should attempt to connect a maximum of 3 times to the primary gateway with a minimum time out value of 3 seconds between attempts before attempting to connect to the secondary gateway – and this should be retried a maximum of a further 3 times. After 6 failed connection attempts (3 on each gateway) the clients should contact London Stock Exchange for further guidance.

3.5 Message Rate Throttling

Turquoise has implemented a scheme for throttling message traffic where each Participant is only permitted to submit up to a specified number of messages per second. Every message which exceeds the maximum rate of a CompID will be rejected via a Business Message Reject (with BusinessRejectReason (380) of Other (0) and Text (58) field = “Message rate exceeded”). A client’s connection will be disconnected by the server if its message rate exceeds the maximum rate for a specific time duration. The rates can be seen in the Turquoise Trading Technical Parameters document. In such a case, the server will transmit a Logout message (with SessionStatus (1409) = 102 (Logout by market operations) and Text (58) = "Maximum Message Rate Exceeded”) and 5 seconds afterwards will terminate the TCP/IP connection.

Please note that client Heartbeat messages, reject messages and any other client-initiated administrative messages are not counted towards the throttling limits.

4.0 FIX Connections and Sessions

4.1 Establishing a FIX Connection

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

Each Participant will use the assigned IP address and port to establish a TCP/IP session with the server. The Participant will initiate a FIX session at the start of each trading day by sending the Logon message. The connection will be terminated if the first message received after establishing the connection is incorrect (i.e. not the Logon message) or is garbled. The Participant will identify itself using the SenderCompID (49) field. The server will validate the CompID and password of the Participant.

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

When the Participant 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 Participant and server is in sync before sending out any missed or new application messages.

The Participant must wait for the server's Logon response before sending additional messages. If the Participant sends messages prior to sending the Logon message or prior to receiving the Logon response, the server will break the TCP/IP connection with the Participant without sending any message.

If a logon attempt fails because of an invalid SenderCompID, TargetCompID, invalid password or because the Participant does not have the appropriate privileges, the server will break the TCP/IP connection with the Participant without sending a Logout or Reject message.

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 Participant.

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 break the TCP/IP connection with the Participant without sending a Logout or Reject message. The server will not increment the next inbound message sequence number expected from the Participant 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 DefaultApplVerID…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 Participant 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.

A protection mechanism is in place in order to protect the gateway from rapid login/logouts. If a user reaches the thresholds for rapid login/logouts, any future logins/logouts will be delayed exponentially.

4.2 Maintaining a FIX Session

4.2.1 Message Sequence Numbers

As outlined in the FIX protocol, the Participant 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 Participants to identify and react to missed messages and to gracefully synchronize applications when reconnecting during a FIX session.

If any message sent by the Participant 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 Participant is expected to employ the same logic.

4.2.2 Heartbeats

The Participant 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 Participant's Logon message.

The server will send a Heartbeat anytime it has not transmitted a message for the heartbeat interval. The Participant is expected to employ the same logic.

As a safety mechanism, the system will not allow the user to login if the HeartBtInt is set to 0. Therefore, if the server receives a logon with HeartBtInt = 0, the user will receive a logout message with SessionStatus = 101 (Logout due to session level failure) and Text = 'HeartBtInt should be greater than zero'.

If the server detects inactivity for a period longer than three heartbeat intervals it will send a Test Request message to force a Heartbeat from the Participant. If inactivity continues for another three heartbeat intervals, the server will send a Logout and break the TCP/IP connection with the Participant. The Participant is expected to employ similar logic if inactivity is detected on the part of the server.

4.2.3 Increasing Expected Sequence Number

The Participant 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 Participant 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 Participant is expected to terminate each FIX connection at the end of each trading day before the server shuts down. The Participant will terminate a connection by sending the Logout message. The server will respond with a Logout to confirm the termination. The Participant 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 Participant 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 Participant is authenticated, the server will respond with a Logon message. The SessionStatus (1409) of this message will be Session Active (0). If the Participant's Logon message included the field NewPassword (925) and the Participant is authenticated, the SessionStatus (1409) of the Logon sent by the server will be Session Active (0).

When the Participant 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 Participant and server is in sync before sending out any missed or new application messages.

The Participant must wait for the server's Logon before sending additional messages. If additional messages are received from the Participant before the exchange of Logon messages, the TCP/IP connection with the Participant 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: Starting a New FIX Session

4.4.1.1 Reset Initiated by the Participant

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

These features are intended to help a Participant 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 Participant in the unlikely event of an outage.

However, Participants are required to support a manual request by Turquoise to initialize sequence numbers prior to the next login attempt.

5.0 Recovery

5.1 Resend Requests

The Participant 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).

5.2 Possible Duplicates

The server handles possible duplicates according to the FIX protocol. The Participant 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

The server may, in the circumstances outlined in sections 5.4.4 Requesting Retransmission of Missed Trades and 5.4.6 Disconnection Prior to Completion of Retransmission use the PossResend (97) field to indicate that a Trade Capture Report may have already been sent under a different MsgSeqNum (34). The Participant should validate the TradeReportID (571) of such a message against that of previous Trade Capture Reports received from the server during the current trading day.

If a Trade Capture Report with same TradeReportID (571) had been processed, the resent Trade Capture Report should be ignored. If the same TradeReportID (571) had not been processed, the message should be processed.

The server does not handle possible resends for Participant-initiated messages and ignores the value in the PossResend (97) field of such messages.

5.4 Transmission of Missed Messages

The Trade Capture Reports generated during a period when a Participant is disconnected from the server will be sent to the Participant 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”.

5.4.1 Application Sequencing and Recovery

The server supports the application sequencing and recovery features introduced in Service Pack 2 for FIX 5.0. A Participant may use the Application Message Request to recover missed trades in scenarios such as the following:

(i) Trades are missed due to a late connection or disconnection during the day.

(ii) Session level recovery via a Resend Request is unavailable due to a sequence number reset initiated by the Participant or server.

(iii) All or some of the trades transmitted by the server during the current day are lost due to a failure at the Participant site.

5.4.2 Application Sequencing by Server

The matching system consists of a series of parallel partitions each of which provide the matching service for an exclusive set of securities.

Each Trade Capture Report transmitted by the server will include the identity of the matching partition that generated the trade and the partition's internal sequence number for the trade in the fields ApplID (1180) and ApplSeqNum (1181) respectively. As the matching partitions operate in parallel and employ the same application sequencing scheme, an ApplSeqNum (1181) is only unique per ApplID (1180). The ApplSeqNum of each ApplID will be initialized to "1" at the start of each trading day.

As a Participant will only receive a subset of the trades executed by each matching partition, the field ApplLastSeqNum (1350) is also included in each Trade Capture Report. This field will contain the ApplSeqNum of the last Trade Capture Report generated for Participant. This will enable Participants to distinguish deliberate sequence gaps from application errors by comparing the value of ApplLastSeqNum (1350) to the ApplSeqNum (1181) of the last received Trade Capture Report from the same ApplID (1180).

Turquoise may change the number of partitions and the securities each serves with due notice to Participants.

5.4.3 Detecting an Application Sequence Gap

A Participant can detect a dropped message by comparing the ApplLastSeqNum (1350) of each new Trade Capture Report against the ApplSeqNum (1181) of the last trade received from the same ApplID (1180).

In the case of a reconnection, the Participant can either wait for the next Trade Capture Report to determine whether trades have been missed or issue a request for the most current ApplSeqNum for each ApplID.

Requesting the Latest ApplSeqNum

The Participant may use the Application Message Request to request the latest ApplSeqNum for one or more ApplIDs. The ApplReqType (1347) of the message should be Request for Last ApplLastSeqNum (2).

Response to Request for Latest ApplSeqNum

The server will respond to the Application Message Request with an Application Message Request Ack. If the request was unsuccessful for a particular ApplID, the reason will be specified in the field ApplResponseError (1354). In the case of a successful request, the ApplSeqNum of the last trade generated for the Participant by each ApplID will be specified in the field RefApplLastSeqNum (1357).

5.4.4 Requesting Retransmission of Missed Trades

The Participant may use the Application Message Request to recover any lost trades. The ApplReqType (1347) of the message should be Retransmission of Application Messages (0). The message may be used in one of four modes:

(i) To request a single trade. The ApplBegSeqNum (1182) and ApplEndSeqNum (1183) should be the same.

(ii) To request a specific range of trades. The ApplBegSeqNum (1182) should be the first trade of the range and the ApplEndSeqNum (1183) should be the last of the range.

(iii) To request all trades after a particular trade. The ApplBegSeqNum (1182) should be the application sequence number immediately after that of the last processed trade and the ApplEndSeqNum (1183) should be zero (0).

(iv) To request all trades for the day. The ApplBegSeqNum (1182) should be one (1) and the ApplEndSeqNum (1183) should be zero (0).

In all cases, the Participant should identify the matching partition to which the request relates via the field RefApplID (1355).

5.4.5 Response to a Trade Retransmission Request

The server will respond to the Application Message Request with an Application Message Request Ack to indicate whether the retransmission request is successful or not. If the request was unsuccessful for a particular ApplID, the reason will be specified in the field ApplResponseError (1354).

In the case of a successful retransmission request, the server will resend the requested Trade Capture Reports immediately after the Application Message Request Ack. Each Trade Capture Report will include an ApplResendFlag (1352) of “Y” to indicate that it is resent in response to an Application Message Request. The resent messages will not include the field ApplLastSeqNum (1350).

5.4.6 Disconnection Prior to Completion of Retransmission

If the FIX connection is terminated prior to the completion of the Trade Capture Report retransmission, the Participant should submit a new Application Message Request once it reconnects to the server.

6.0 Supported Message Types

This section provides details on the header and trailer, the seven administrative messages and five application messages utilized by the server. Also, please note that administrative messages are not validated for undefined TAGs. However, if a required filed is missing, the message will be rejected via a Session Reject. Any message not included in this section will be ignored by the server.

6.1 Administrative Messages

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

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

6.2 Application Messages (Participant-Initiated)

Message	MsgType	Usage
Application Message Request	BW	Allows the Participant to request one of the following: (i) Retransmission of missed trades (ii) Latest ApplSeqNum of each ApplID
Trade Capture Report Request	AD	Allows the Participant to request for a set of Trade Capture Reports from the server

6.3 Application Messages (Server-Initiated)

Message	MsgType	Usage
Trade Capture Report	AE	Indicates one of the following: (i) Trade (ii) Trade bust
Application Message Request Ack	BX	Indicates whether a request to retransmit trades or for the latest ApplSeqNum is successful or not.
Trade Capture Report Request Ack	AQ	Acknowledges the receipt of a Trade Capture Report Request message from the Participant.

7.0 Message Formats

This section provides details on the header and trailer, the seven administrative messages and three application messages utilized by the post trade gateway. The system will ignore an undefined tag sent along with any Administrative message and will process the rest of the message. However if an undefined tag is sent along with an Application message, then the system will completely reject the message.

7.1 Message Header and Trailer

7.1.1 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	The message type. **Value** **Meaning** 0 Heartbeat 1 Test Request 2 Resend Request 3 Reject 4 Sequence Reset 5 Logout A Logon AD Trade Capture Report Request AE Trade Capture Report AQ Trade Capture Report Request Ack BW Application Message Request BX Application Message Request Ack
49	SenderCompID	Y	CompID of the party sending the message.
56	TargetCompID	Y	CompID of the party the message is sent to. **Value** **Meaning** PTGW FIX Post Trade Gateway
34	MsgSeqNum	Y	The sequence number of the message.

43	PossDupFlag	N	Whether the message was previously transmitted under the same MsgSeqNum (34). Absence of this field is interpreted as Original Transmission (N). **Value** **Meaning** Y Possible Duplicate N Original Transmission
97	PossResend	N	Whether the message was previously transmitted under a different MsgSeqNum (34). Absence of this field is interpreted as Original Transmission (N). **Value** **Meaning** Y Possible Resend N Original Transmission
52	SendingTime	N	Time the message was transmitted. Not required on Messages sent by Participant (even if sent by a Participant, no validation will be done). Will be stamped on 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	Version of FIX used in the message. Required if the message is generated by the server. **Value** **Meaning** 9 FIX50SP2
115	OnBehalfOfCompID	N	The ID of the party whose on behalf the message is sent; will only be used in Participant initiated messages.
128	DeliverToCompID	N	Will be included in the Login Response with the value specified on OnBehalfOfCompID on Participant's login message. Will be used in server initiated messages only.

7.1.2 Message Trailer

Tag	Field Name	Req	Description
10	CheckSum	Y	

7.2 Administrative Messages

7.2.1 Logon

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	A = Logon
Message Body			
98	EncryptMethod	Y	Method of encryption. **Value** **Meaning** 0 None
108	HeartBtInt	Y	Indicates the heartbeat interval in seconds.
141	ResetSeqNum Flag	N	Indicates whether the Participant and server should reset sequence numbers. Absence of this field is interpreted as Do Not Reset Sequence Numbers (N). **Value** **Meaning** Y Reset Sequence Numbers N Do Not Reset Sequence Numbers
554	Password	N	Password assigned to the CompID. Required if the message is generated by the Participant.
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. **Value** **Meaning** 0 Session Active 2 Password Due to Expire 3 New session password does not comply with policy
1137	DefaultApplVerID	Y	Default version of FIX messages used in this session. **Value** **Meaning** 9 FIX50SP2
Standard Trailer			

7.2.2 Logout

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	5 = Logout
Message Body			
1409	SessionStatus	N	Status of the FIX session. Required if the message is generated by the server. **Value** **Meaning** 4 Session logout complete 6 Account locked 7 Logons are not allowed at this time 8 Password expired 100 Other 101 Logout due to session level failure 102 Logout by Market Operations
58	Text	N	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. In other cases the field will contain the reason for the logout.
Standard Trailer			

7.2.3 Heartbeat

Tag	Field Name	Req	Description
Standard Header			
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.
Standard Trailer			

7.2.4 Test Request

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	1 = Test Request
Message Body			
112	TestReqID	Y	Identifier for the request.
Standard Trailer			

7.2.5 Resend Request

Tag	Field Name	Req	Description
Standard Header			
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.
Standard Trailer			

7.2.6 Reject

Tag	Field Name	Req	Description
Standard Header			
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	SessionRejectReason	N	Code specifying the reason for the reject. Refer to TQ801 for a list of reject codes.
58	Text	N	Text specifying the reason for the rejection.
Standard Trailer			

7.2.7 Sequence Reset

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	4 = Sequence Reset
Message Body			
36	NewSeqNo	Y	Sequence number of the next message to be transmitted.
123	GapFillFlag	N	Mode in which the message is being used. Absence of this field is interpreted as Sequence Reset (N). **Value** **Meaning** Y Gap Fill N Sequence Reset
Standard Trailer			

7.3 Application Messages

7.3.1 Trade Capture Report - Server-Initiated

The server uses the message to relay trades and trade busts. A server generated TCR message will be generated for the following conditions, one TCR for the buyer and one for the seller (hence if a firm is on both sides of the trade they will receive 2 TCRs):

I. An on book trade

II. An on book trade cancellation (by market operations)

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	AE = Trade Capture Report
Message Body			
1180	ApplID	Y	Identifier of the matching partition
1181	ApplSeqNum	Y	Matching partition's sequence number for trade.
571	TradeReportID	Y	Identifier of this message.
1003	TradeID (TVTIC)	Y	Server assigned identifier of the trade. Refers to the base 36 (G offset) encoded value of Trade ID field in GTP Market Data and Trade Match ID in Native Order Management Gateways.
820	TradeLinkID	Y	A unique ID for all the trades relating to this transaction.

487	TradeReportTransType	Y	Type of transaction being reported. **Value** **Meaning** 0 New 1 Cancel
856	TradeReportType		Type of trade report. **Value** **Meaning** 0 Submit 7 On Book Trade Cancel
1123	TradeHandlingInstr		Handling instructions to Participant or the server. **Value** **Meaning** 0 Trade Confirmation
828	TrdType	Y	Type of the trade **Value** **Meaning** 0 Regular Trade
60	TransactTime	Y	Time the trade or trade bust occurred.
32	LastQty	Y	Traded quantity.
31	LastPx	Y	Traded price specified in the instrument's trading currency.
381	GrossTradeAmt	Y	Total consideration due from the buyer to the seller.
574	MatchType	Y	Point in matching process trade was matched. **Value** **Meaning** 4 Continuous Trading **Note:** All Turquoise trades irrespective of whether they are matched during Continuous Trading, **Turquoise Plato Uncross™**, **Turquoise Lit Auctions™** Order Book will have the value 4. Trades in the **Turquoise Lit Auctions™** Order Book or **Turquoise Plato Uncross™** will have one of the following SideLiquidityInd: 4 - **Turquoise Lit Auctions™** Order Book 9 - **Turquoise Plato Block Discovery™** Execution - **Turquoise Plato Uncross™** Execution 10 - **Turquoise Plato Uncross™** Execution

150	ExecType		Y	Type of execution that is being referred to by this message. **Value** **Meaning** F Trade H Trade Cancel
20111	NovatedIndicator		Y	Defines whether the trade needs to be sent to clearing system. Will be set to 0 (No) for internalised trades. **Value** **Meaning** 0 No 1 Yes
55	MTFCommonSymbol		Y	MTF Common Symbol of the instrument. (Max. length 8 bytes)
48	SecurityID		Y	ISIN of the instrument.
22	SecurityIDSource		Y	**Value** **Meaning** 4 ISIN
15	Currency		Y	ISO Currency Code.
207	SecurityExchange		Y	Market Identifier Code.
552	NoSides		Y	The number of sides in the Trade Capture Report. This will be set to “1”
➡	54	Side	Y	Side of the executed Order. **Value** **Meaning** 1 Buy 2 Sell
➡	1427	SideExecID	N	Identifier of the execution received by the Order. Only populated in on book trades and cancellations
➡	453	NoPartyIDs	Y	Number of party identifiers.

<table>
<tr><td>➡</td><td>➡</td><td>448</td><td>PartyID</td><td>Y</td><td>Identifier of the party. Refer to TQ201 for details on this values will be populated.
If a trade is internalized, the Counterparty Firm (17) will be populated with the user’s own Firm ID.
If an on book trade is cleared, the Counterparty Firm (17) will be populated with the CCP as below:

ECCP
LCH
X-Clear
LCH SA

If an on book trade is cleared, the Clearing Organization (24) will be populated with the Clearing Member.
If an on book trade is not cleared, the Counterparty Firm (17) will be populated with the Contra Party Firm ID.
<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>0</td><td>None</td></tr><tr><td>1</td><td>AGGR</td></tr><tr><td>2</td><td>PNAL</td></tr><tr><td>3</td><td>CLIENT</td></tr></table></td></tr>
<tr><td>➡</td><td>➡</td><td>447</td><td>PartyIDSource</td><td>Y</td><td>Value 'P' will only be applicable to on-book trades.

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>D</td><td>Proprietary/Custom Code</td></tr><tr><td>P</td><td>Short Code</td></tr></table></td></tr>
</table>

<table>
<tr><td>➡</td><td>➡</td><td>452</td><td>PartyRole</td><td>Y</td><td>Role of the specified PartyID (448).

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>1</td><td>Executing Firm</td></tr><tr><td>7</td><td>Entering Firm</td></tr><tr><td>17</td><td>Contra Firm</td></tr><tr><td>100</td><td>Trader ID</td></tr><tr><td>76</td><td>Trader Group</td></tr><tr><td>24</td><td>Clearing Organization</td></tr></table>Below values will only be applicable to on book trades and will always be populated:<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>3</td><td>Client ID</td></tr><tr><td>122</td><td>Investment Decision Maker</td></tr><tr><td>12</td><td>Executing Trader</td></tr></table></td></tr>
<tr><td>➡</td><td>➡</td><td>2376</td><td>PartyRoleQualifier</td><td>N</td><td>Provides a further qualification for the value specified in the PartyRole (452).

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>22</td><td>Algorithm</td></tr><tr><td>23</td><td>Firm or Legal Entity</td></tr><tr><td>24</td><td>Natural Person</td></tr></table>This field is only applicable to on-book trades.</td></tr>
<tr><td>➡</td><td>1</td><td colspan="2">Account</td><td>N</td><td>Participant Reference specified at Order entry</td></tr>
<tr><td>➡</td><td>581</td><td colspan="2">AccountType</td><td>Y</td><td>Clearing Account Type

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>1</td><td>Client</td></tr><tr><td>3</td><td>House</td></tr></table></td></tr>
<tr><td>➡</td><td>2667</td><td colspan="2">AlgorithmicTradeIndicator</td><td>N</td><td>Whether the order was generated via an algorithm.

This field is only applicable to on-book trades.<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>1</td><td>True</td></tr></table></td></tr>
</table>

<table>
<tr><td>➡</td><td>1724</td><td colspan="2">OrderOrigination</td><td>N</td><td>Whether the order was generated via Direct Electronic Access (DEA) or not.

This field is only applicable to on-book trades.

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>5</td><td>DEA</td></tr></table></td></tr>
<tr><td>➡</td><td>2593</td><td colspan="2">NoOrderAttributes</td><td>N</td><td>No of Order Attributes

The NoOrderAttributes block is only applicable to on-book trades.</td></tr>
<tr><td>➡</td><td>➡</td><td>2594</td><td>OrderAttributeType</td><td>N</td><td>Provides information about the order. States if the order has been submitted as a part of liquidity provision activity (i.e. as a part of the market making strategy).

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>2</td><td>Liquidity Provision</td></tr></table></td></tr>
<tr><td>➡</td><td>➡</td><td>2595</td><td>OrderAttributeValue</td><td>N</td><td>Will be populated if OrderAttributeType (2594) is specified

<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>Y</td><td>Yes</td></tr></table></td></tr>
</table>

<table>
<tr><td>➡</td><td>1444</td><td>SideLiquidityInd</td><td>N</td><td>Whether the Order added or removed liquidity. Turquoise Lit™ Order Book trades will only have enums 1 and 2 stamped on them. All enums will be used in the Turquoise Plato™ Order Book. Only populated in on book trades and cancellations.

Value Meaning
1 Added Liquidity
2 Removed Liquidity
4 Turquoise Lit Auctions™ Execution
8 Turquoise Plato Block Discovery™ Execution – Continuous Trading
9 Turquoise Plato Block Discovery™ Execution – Turquoise Plato Uncross™
10 Turquoise Plato Uncross™ Execution</td></tr>
<tr><td>➡</td><td>11</td><td>ClOrdID</td><td>N</td><td>Identifier of the executed Order as specified by the entering firm.

The field will be populated for on book trades and on book trade cancellations sent to the Participant and on Execution Reports sent to CCP gateways.</td></tr>
<tr><td>➡</td><td>37</td><td>OrderID</td><td>N</td><td>Identifier of the executed Order as specified by matching system. The field will be populated for on book trades and on book trade cancellations</td></tr>
<tr><td>➡</td><td>528</td><td>OrderCapacity</td><td>Y</td><td>Capacity of the firm that placed the order

Value Meaning
A Any other trading capacity (AOTC)
P Dealing on own account (DEAL)
R Matched Principal (MTCH)</td></tr>
<tr><td>573</td><td colspan="2">MatchStatus</td><td>Y</td><td>Status of the pre-negotiated trade.

Value Meaning
0 Matched
1 Unmatched</td></tr>
</table>

<table>
<tr><td>1350</td><td colspan="2">ApplLastSeqNum</td><td>N</td><td>ApplSeqNum of last trade generated for Participant. Required if ApplResendFlag (1352) is not “Y”.</td></tr>
<tr><td>1352</td><td colspan="2">ApplResendFlag</td><td>Y</td><td>Whether the message is sent in response to an Application Message Request.
Value Meaning
Y Response to Application Message Request
N Original Transmission</td></tr>
<tr><td>572</td><td colspan="2">TradeReportRefID</td><td>N</td><td>Reference to trade being cancelled. Only sent in on trade cancellations.</td></tr>
<tr><td>1094</td><td colspan="2">PegPriceType</td><td>N</td><td>Only applicable to Turquoise Plato™ Order Book, will not be sent for Turquoise Lit™ Order Book, Turquoise Lit Auctions™ or Turquoise Lit Auctions™ executions.
Value Meaning
0 Midpoint</td></tr>
<tr><td>1301</td><td colspan="2">MarketID</td><td>N</td><td>Indicates the liquidity pool being used
Value | Meaning
TRQX | Turquoise Lit™ Order Book
TRQA | Turquoise Lit Auctions™ Order Book
TRQM | Turquoise Plato™ Order Book</td></tr>
<tr><td>2668</td><td colspan="2">NoTrdRegPublications</td><td>N</td><td>The number of regulatory publication rules in the repeating group. Will be set to 1 for the RFPT Pre-trade flag.</td></tr>
<tr><td>➡</td><td>2669</td><td>TrdRegPublicationType</td><td>N</td><td>Specifies the type of regulatory trade publication.
Value | Meaning
0 | Pre-trade transparency waiver</td></tr>
</table>

➡	2670	TrdRegPublicationReason	N	Populated when Execution Type is F or H. The Pre-trade Waiver Flags section describes in which scenarios the values are populated. **Value** / **Meaning** 3 / RFPT
1125		OrigTradeDate	N	Specifies the original date and time of the trade. For trade cancellations, the original trade date and time is stamped.
Standard Trailer				

7.3.2 Trade Capture Report Request- Participant-Initiated

Tag	Field Name		Req	Description
Standard Header				
35	MsgType		Y	AD = Trade Capture Report Request
Message Body				
568	TradeRequestID		Y	Identifier for the trade request.
569	TradeRequestType		Y	**Value** / **Meaning** 0 / All Trades 1 / Trades Matching Specified Criteria If none of the criteria is specified, this will return all trades for the particular participant.
150	ExecType		N	**Value** / **Meaning** F / Trade H / Trade Cancel
55	MTF Common Symbol		N	MTF Common Symbol of the instrument. Required if values are missing for SecurityID (48), Currency (15) and SecurityExchange (207). (Max. length 8 bytes)
48	SecurityID		N	ISIN of the instrument. Required if MTF Common Symbol (55) is not specified.
454	NoSecurityAltID		N	If present, value in this field should always be “1”.
➡	455	SecurityAltID	N	Identification number of the instrument.

➡	456	SecurityAltID Source	N	Type of instrument identification used. Required if SecurityAltID (455) is specified. When an ISIN is specified, all trades related to that ISIN will be disseminated (irrespective of the fact that whether the same ISIN had been used in multiple instruments). **Value** **Meaning** 4 ISIN
22	SecurityIDSource		N	**Value** **Meaning** 4 ISIN
15	Currency		N	ISO Currency Code. Required if MTF Common Symbol (55) is not specified.
207	SecurityExchange		N	Market Identifier Code. Required if MTF Common Symbol (55) is not specified.
1	Account		N	Participant reference for the trade.
54	Side		N	Side of the executed Order. **Value** **Meaning** 1 Buy 2 Sell
11	ClOrdID		N	Identifier of the executed Order as specified by the entering firm.
37	OrderID		N	Identifier of the executed Order as specified by matching system.
453	NoPartyIDs		Y	Number of party identifiers.
➡	448	PartyID	Y	Set to the MemberFirmID for internalised trade executions else set to the CCP name: ECCP LCH X-Clear LCH SA
➡	447	PartyIDSource	Y	**Value** **Meaning** D Generally Accepted Market Participant Identifier

➡	452	PartyRole	Y	Role of the specified PartyID (448). **Value** **Meaning** 1 Executing Firm 76 Trading Group 17 Counterparty 100 Trader ID 24 Clearing Organisation
Standard Trailer				

7.3.3 Application Message Request - Participant-Initiated

Tag	Field Name		Req	Description
Standard Header				
35	MsgType		Y	BW = Application Message Request
Message Body				
1346	ApplReqID		Y	Participant specified unique identifier of the request.
1347	ApplReqType		Y	Type of request. **Value** **Meaning** 0 Retransmission of Application Messages 2 Request for Last ApplLastSeqNum
1351	NoApplIDs		Y	Number of ApplIDs to which the request relates.
➡	1355	RefApplID	Y	Identifier of the matching partition.
➡	1182	ApplBeg SeqNum	N	Application sequence number of first message in range to be resent. Required if ApplReqType (1347) is Retransmission of Application Messages (0).
➡	1183	ApplEnd SeqNum	N	Application sequence number of last message in range to be resent. Required if ApplReqType (1347) is Retransmission of Application Messages (0).
Standard Trailer				

7.3.4 Application Message Request Ack - Server-Initiated

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	BX = Application Message Request Ack

Message Body				
1353	ApplResponseID		Y	Server specified identifier of the acknowledgement.
1346	ApplReqID		Y	Identifier of the request being acknowledged.
1347	ApplReqType		Y	Type of request being acknowledged. **Value** **Meaning** 0 Retransmission of Application Messages 2 Request for Last ApplLastSeqNum
1351	NoApplIDs		Y	Number of ApplIDs to which the request relates.
➡	1355	RefApplID	Y	Identifier of the matching partition.
➡	1182	ApplBeg SeqNum	N	Application sequence number of first message in range to be resent. Required if ApplReqType (1347) is Retransmission of Application Messages (0).
➡	1183	ApplEnd SeqNum	N	Application sequence number of last message in range to be resent. Required if ApplReqType (1347) is Retransmission of Application Messages (0).
➡	1357	RefAppl LastSeq Num	N	ApplSeqNum of the last trade generated for the Participant. Required if ApplReqType (1347) is Request for Last ApplLastSeqNum (2) and ApplResponseError (1354) is not specified.
➡	1354	Appl Response Error	N	Reason request is rejected. **Value** **Meaning** 0 ApplID Does Not Exist 1 Requested Trades are Not Available
Standard Trailer				

7.3.5 Trade Capture Report Request Ack – Server Initiated

Tag	Field Name	Req	Description
Standard Header			
35	MsgType	Y	AQ = Trade Capture Report Request Ack
Message Body			
568	TradeRequestID	Y	Identifier of the request being acknowledged.
569	TradeRequestType	Y	Value specified in the request.
750	TradeRequestStatus	Y	Whether the request is accepted or rejected. **Value** **Meaning** 0 Accepted 2 Rejected

<table>
<tr><td>749</td><td>TradeRequestResult</td><td>Y</td><td>Reason the request is rejected.<table><tr><th>Value</th><th>Meaning</th></tr><tr><td>0</td><td>Successful</td></tr><tr><td>9</td><td>Not Authorized</td></tr><tr><td>100</td><td>Cannot Match Selection Criteria</td></tr><tr><td>200</td><td>Request Limit for Day Reached</td></tr></table></td></tr>
<tr><td>748</td><td>TotNumTradeReports</td><td>N</td><td>Number of Trade Capture Reports that will be sent in response to the request. Required if TradeRequestStatus (750) is Accepted (0).</td></tr>
<tr><td colspan="4">Standard Trailer</td></tr>
</table>

8.0 Service availability

Customer Activity	Availability
Telnet Access	04.00 – 20:15
Login Access	04:00 – 20:15

Disclaimer

This service description is being distributed by Turquoise Global Holdings Limited only to, and is directed only at (a) persons who have professional experience in matters relating to investments who fall within Article 19(1) of the FSMA 2000 (Financial Promotion) Order 2005 and (b) persons to whom it may otherwise lawfully be communicated (together “relevant persons”). Any investment or investment activity to which this document relates is available only to and will be engaged in only with, relevant persons. Any person who is not a relevant person should not act or rely on this service description or any of its contents.

Turquoise Global Holdings Limited is an authorised investment firm by the Financial Conduct Authority.

Turquoise Global Holdings Europe B.V. is an investment firm authorised and regulated by the Autoriteit Financiële Markten (AFM) of the Netherlands.

Contact Details

Turquoise Global Holdings Limited
10 Paternoster Square, London, EC4M 7LS
E: sales@tradeturquoise.com
T: +44 20 7382 7600

Turquoise