



FIX DROP COPY for AIX

Version: 1.0

Publication Date: 26.09.2025



Genium INETSM

All content in this document is owned, or licensed, by Nasdaq, Inc. or its affiliates ('Nasdaq'). Unauthorized use is prohibited without written permission of Nasdaq. While reasonable efforts have been made to ensure that the contents of this document are accurate, the document is provided strictly "as is", and no warranties of accuracy are given concerning the contents of the information contained in this document, including any warranty that the document will be kept up to date. Nasdaq reserves the right to change details in this document without notice. To the extent permitted by law no liability (including liability to any person by reason of negligence) will be accepted by Nasdaq or its employees for any direct or indirect loss or damage caused by omissions from or inaccuracies in this document.

Copyright © 2018 Nasdaq, Inc. All rights reserved.

CONTENTS

References	6
Overview	6
Supported messages	7
Administrative messages	7
Inbound Application messages.....	7
The FIX Repository	7
The FIX Session	8
CompIDs	8
SenderSubID	8
Renewal of passwords	8
Expired passwords.....	8
Logon	8
Heartbeat intervals.....	9
Encryption	9
Datatypes and required fields	9
Character encoding.....	9
Session lifetime	9
Failover and message recovery	9
FIX Session Level Test Cases	10
The Standard Header	10
Possible Duplicate vs. Possible Resend.....	11
The Standard Trailer.....	11
Message Details.....	11
How to interpret the Required (Req'd) column.....	11
Repeating groups	11
Logon – inbound to Marketplace	12
Logon – outbound from Marketplace.....	14
Logout (in/out)	14
Sequence Reset (in/out).....	15
Resend Request (in/out)	15
Reject (out).....	15
Heartbeat (in/out)	16
Test Request (in/out)	16
FIX Drop Copy Sessions.....	17
Drop Copy Sessions	17
Drop Party Identifiers.....	17

Order and Trade Messages	18
Execution Report	18
Order identifiers	18
Order Cancels	18
Message Details – Execution Report.....	19
Trade Capture Report	21
Trade identifiers	21
Trade Capture Report features.....	21
Trade Types	21
Message Details – Trade Capture Report	22

REVISION HISTORY

Date	Revision	Change Description
26-09-2025	1.0	Initial version.

References

[1] FIX 5.0 SP2 Protocol Specification

<http://fixprotocol.org/specifications/fix5.0sp2spec>

[2]

FIX 5.0 SP2 Protocol Specification

<http://fixprotocol.org/specifications/fix5.0sp2spec>

[3]

FIX Protocol Limited, Market Data Optimization Working Group:

Recommended Practices for Book Management, Version 2.00, 2007

http://fixprotocol.org/documents/2518/MDOWG_Book_Mgt_v20.doc

[4]

RFC 2045, Base64 encoding for MIME

<http://www.ietf.org/rfc/rfc2045.txt>

Overview

This document contains the specification for the FIX Drop interface to AIX trading system. This document outlines the details of the drop copy session that differ from the regular order entry FIX session. It contains a detailed lists of all fields included in every message seen on the drop session. However, it does not go into the level of detail that the regular FIX specification does, where different flavors of the same message have different listings to illustrate subtle differences.

Please refer to the regular FIX specification for such details.

The interface is based on the FIX Protocol 5.0 SP2 standard (Financial Information exchange). More detailed information about the standard can be found in FIX specification document see [1].

Supported messages

Administrative messages

Logon

Logout

Sequence Reset

Resend Request

Reject

Heartbeat

Test Request

Inbound Application messages

Execution Report

Trade Capture Report

The FIX Repository

This specification defines the full set of messages, fields and enumerated values that can be used. As with most FIX implementations, this only supports a small subset of all available messages, components, fields and enumerated values defined in FIX 4.4. An FPL-formatted repository corresponding to this specification is delivered separately.

NOTE: Inbound messages not conforming to this spec, will be rejected with a session-level Reject message.

The FIX Session

The session layer conforms to the standard FIX session. Please see the standard FIX specification for additional details.

CompIDs

The Sender- and TargetCompID uniquely define the FIX session. A session can only be active (established) between two hosts simultaneously. Any attempts to establish a second FIX session using the same CompIDs (for instance to a backup gateway) in parallel will be rejected.

- The TargetCompID for transactions sent inbound to the Exchange will be “AI” for production and “AI_TEST” for test systems.
- The SenderCompID for transactions sent outbound from the Exchange will be “AI” for production and “AI_TEST” for test systems.

SenderSubID

Each inbound business transaction must have the SenderSubID (tag 50) field set to an authenticated user. One user can be authenticated by setting the Username and Password field in the Logon message. The SenderSubID on incoming transactions will be echoed back in TargetSubID (tag 57) on outbound transactions.

Renewal of passwords

A new password may be set by setting the NewPassword (925) field along with the current password in the Password (554) field. This can be done with the Logon message. The SessionStatus (1409) field of the Logon returned to the client can be checked to see if the new password was accepted.

Expired passwords

If the password has expired when a client tries to log in, the system will respond with a Logout message with SessionStatus set to 8 – Password expired. To gain access, the client must issue a new Logon message with NewPassword set (along with the expired password in Password). If the new password is not valid, the system will respond with another Logout message. SessionStatus will be set to 3 – New session password does not comply with policy. The client will be able to log in again with another new password.

Logon

At Logon, clients are identified by:

- CompIDs
- IP Address

The Logon Username and Password fields are used to authenticate the client. When the client is authenticated, the system responds with a Logon message to the client.

Heartbeat intervals

Heartbeat intervals are negotiated at Logon using the HeartBtInt (108) field. The system allows heartbeat intervals greater than 10 seconds. Recommended heartbeat interval is 30 s. A heartbeat interval set lower than 10 seconds will result in a Logout response.

Encryption

The system does not support encryption.

Datatypes and required fields

This specification does not change the datatype on any fields defined in the standard FIX specification. There may be places where this specification restricts the value range of a field further than specified in standard FIX. This will be clearly marked in the spec.

All fields listed in this specification that are marked as required in the standard specification, are required also in this specification. This document specifies additional fields as required. These fields are marked with a 'Q' in the required column of the message listings.

Character encoding

FIX gateway will use the 8-bit standard ISO-8859-9 encoding, often called Latin-9. The lower 7 bits are compatible with the standard 7-bit ASCII character encoding.

Session lifetime

The FIX session lifetime is restricted to one trading day. The session lifetime is not ended at connectivity loss or even Logouts. The sequence numbers are reset to one each morning.

Failover and message recovery

At reconnect and Logon standard FIX message recovery is performed. All FIX sessions have at least one primary and one secondary gateway to which the session states are fully replicated. This means that regardless to which gateway a client connects, full message recovery is provided.

A client cannot have the same FIX session active towards multiple gateway instances simultaneously.

FIX API Drop Copy channel works in active-active structure. In other words, the secondary gateway is also actively serving as the primary gateway. With this architectural structure, the time required for the passive gateway to be active is eliminated, and a faster transition from the primary gateway to the secondary gateway is aimed.

The user is normally expected to connect to the primary gateway. If the user has a problem connecting to the primary gateway, it will be possible to switch to the secondary gateway with the same user and port. It is not possible to initiate a session with the same user to both the primary and secondary gateways at the same time.

FIX DC users can connect to the secondary gateway with the last sequence number which is valid on the primary server. If requested, the messages can be recovered from the beginning or for a certain sequence interval on the secondary gateway.

Exchange may consider to shut down or restart the gateway(s) depending on the scope of the problem on the gateway(s).

FIX Session Level Test Cases

This implementation is fully compliant with the session-level test cases specified in the standard FIX

4.4 Specification, Volume 2, section “FIX Session-level Test Cases and Expected Behaviors”. The only exception is the encryption test cases.

The Standard Header

All FIX messages contain a Standard Header. The header contains important information such as session identifiers (CompIDs), sequence numbers and message type and length etc.

Tag num	FIX Field name	Req'd	Comment
8	BeginString	Y	Identifies beginning of new message and protocol version. ALWAYS FIRST FIELD IN MESSAGE. Valid values: FIXT.1.1
9	BodyLength	Y	Message length, in bytes, forward to the CheckSum field. ALWAYS SECOND FIELD IN MESSAGE.
35	MsgType	Y	Defines message type ALWAYS THIRD FIELD IN MESSAGE.
49	SenderCompID	Y	As specified in separate agreement
50	SenderSubID		Required on inbound transactions. Must be set to a valid authenticated user.
56	TargetCompID	Y	As specified in separate agreement
57	TargetSubID		Should not be populated on inbound transactions. Will contain the value of incoming SenderSubID on outbound transactions. In some cases, such as in unsolicited cancels, TargetSubID will not be set.
34	MsgSeqNum	Y	Integer message sequence number.
43	PossDupFlag		Indicates possible retransmission of message with this sequence number. Always required for retransmitted messages

Tag num	FIX Field name	Req'd	Comment
97	PossResend		Indicates that message may contain information that has been sent under another sequence number. Required when message may be duplicate of another message sent under a different sequence number.
52	SendingTime	Y	Time of message transmission (always expressed in UTC (Universal Time Coordinated, also known as "GMT"))
122	OrigSendingTime		Original time of message transmission (always expressed in UTC (Universal Time Coordinated, also known as "GMT"). Required for message resent as a result of a ResendRequest.

Possible Duplicate vs. Possible Resend

The two FIX fields PossDupFlag (43) and PossResend (97) of the Standard Header have different purposes. The PossDupFlag is set on messages retransmitted as a result of a Resend Request. These messages have the original sequence numbers (MsgSeqNum).

PossResend is set on messages resent with a new sequence number. This may be used to resend an order which no response has been received. The gateway will check whether the client identifier (such as the ClOrdID, TradeReportID etc) in the message has been received before. If the client identifier has been seen before, the message will be dropped.

The Standard Trailer

All FIX messages end with a Standard Trailer. The trailer only includes a simple checksum field. The details on how to calculate the checksum can be found in the standard FIX specification.

Tag num	FIX Field name	Req'd	Comment
10	Checksum	Y	

Message Details

How to interpret the Required (Req'd) column

A 'Y' marks the field as required in standard FIX (and of course also in this implementation). A 'Q' means that the field is required in this implementation although it is not required in standard FIX. No entry at all means the field is optional.

Repeating groups

The fields in a FIX Repeating group are marked in the message listings with an arrow. Example (Parties block):

453	NoPartyIDs		Optional repeating group only used for on behalf of transactions.
→	448	PartyID	Q Party identifier.

→	447	PartyIDSource		Q	Valid values:
→	452	PartyRole		Q	Identifies the type of role for the PartyID specified.
→	802	NoPartySubIDs			Number of PartySubIDs present. Only used for PartyRole=Executing Firm. Will always be 1.
→	→	523	PartySubID	Q	Sub-identifier of party. Here Exchange code of the party.
→	→	803	PartySubIDType	Q	Type of PartySubID (523) value

In the above example nested repeating groups can also be seen.

Also notice that the req'd flag on the NumInGroup field (NoPartyIDs, NoPartySubIDs). If it is present (either Y or Q), it means that the whole repeating group will always be present.

A Q or Y set on an individual field in a repeating group means that it will always be present if the repeating group is present.

Logon – inbound to Marketplace

The response to a logon is either a Logon, which denotes a successful logon, or a Logout. A client must be prepared to handle failure scenarios including (but not limited to):

A Logon attempt may fail or be rejected for several reasons. The FIX gateway will react differently depending on the kind of failure. The two different actions it may take are:

Silently ignore the Logon.

- If authentication fails (for security reasons).
- If the wrong Sender or Target CompID is specified.
- For other reasons specified in the standard FIX specifications.
- If the FIX gateway has no connection with the back-end system.

Respond with a Logout.

- Logon failure for other reasons than authentication/security.

The Logout response to a Logon will always contain a note on why in the Text (58) field.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = A
98	EncryptMethod	Y	Encryption not supported. Valid values: Valid values: 0 = None / Other
141	ResetSeqNumFlag		Indicates that both sides of a FIX session should reset sequence numbers. NOTE: Resetting the sequence numbers will result in all prior messaging being lost. Valid values: Y = Yes
108	HeartBtInt	Y	Heartbeat interval. Any value greater than 10 s is accepted. A lower value will result in a Logout response.
553	Username	Q	user
554	Password	Q	password (unencrypted)
925	NewPassword		Specifies a new password for the FIX Logon. The new password is used for subsequent logons.
1137	DefaultApplVerID	Y	The default version of FIX messages used in this session. Valid values: 9 = FIX50SP2

	Standard Trailer		
--	------------------	--	--

Logon – outbound from Marketplace

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = A
98	EncryptMethod	Y	Encryption not supported. Valid values: 0 = None / Other
141	ResetSeqNumFlag		Indicates that both sides of a FIX session should reset sequence numbers. Will only be set as a response to an inbound Logon with this flag set. Valid values: Y = Yes
108	HeartBtInt	Y	As specified in inbound Logon. Valid range: Greater than 10 s
1409	SessionStatus	Q	Status of the FIX session. Valid values: 0 = Session Active 1 = Session password changed 3 = New session password does not comply with policy 9= Invalid Sendersubid or sendercompid 100 = Invalid BodyLength, session suspended 101 = HeartBt interval too low
1137	DefaultApplVerID	Y	The default version of FIX messages used in this session. Valid values: 9 = FIX50SP2
20002	DaysToPwdExpiry		Number of days to password expiration.
	Standard Trailer	Y	

Logout (in/out)

The Logout message is used to gracefully disconnect a FIX session. When receiving a Logout, the counterparty should respond with a Logout. A Logout can also be the response to an unsuccessful Logon attempt.

SessionStatus = 100 means that a critical formatting error has been detected in an inbound transaction. The gateway is unable to reliably continue parsing further messages on the session. The connection is closed and can only be enabled by manual intervention.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 5
1409	SessionStatus		Status of the FIX session. Only set on outbound Logouts. Valid values: 3 = New session password does not comply with policy 4 = Session logout complete 5 = Invalid password or user name 6 = Account locked 8 = Password expired 9= Invalid Sendersubid or sendercompid 100 = Invalid body length in received message, session suspended 101 = Heartbeat interval too low.
58	Text		Free text
	Standard Trailer	Y	

Sequence Reset (in/out)

This message has two uses. The common usage is with GapFillFlag set to 'Y', which is used in a response to a Resend Request to indicate that a range of messages will not be resent. This is commonly used to avoid resending administrative messages like Heartbeats.

The other (very rare) usage is to reset the sequence numbers to a higher number to get out of a deadlock. This is only triggered by manual intervention.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 4
123	GapFillFlag		
36	NewSeqNo	Y	
	Standard Trailer	Y	

Resend Request (in/out)

Resend Request is used to recover messages when a sequence number gap has been detected.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 2
7	BeginSeqNo	Y	
16	EndSeqNo	Y	
	Standard Trailer	Y	

Reject (out)

The Reject, or session-level reject, message is sent whenever the FIX gateway is able to at least partially parse the message, but the message does not adhere to the specification and cannot be delivered to the back-end system.

It is strongly recommended that the Reject <3> message should not be sent by client.

Reject <3> message is unexpected and cannot be parsed. And also processing of the next valid FIX message might cause detection of a sequence gap and Resend Request <2> will be generated. Once a gap is detected and a resend request initiated, server ignores all requests/messages that are sent from the client during the retransmission period.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 3
45	RefSeqNum	Y	
371	RefTagID		
372	RefMsgType		

Tag num	FIX Field name	Req'd	Comment
373	SessionRejectReason	Q	Valid values: 0 = Invalid Tag Number 1 = Required Tag Missing 2 = Tag Not Defined For This Message Type 3= Undefined Tag 4= Tag Specified Without A Value 5= Value Is Incorrect Out Of Range For This Tag 6 = Incorrect Data Format For Value 7 = Description problem 8 = Signature problem 9 = Comp ID Problem 10 = Sending Time Accuracy Problem 11 = Invalid Msg Type 12 = XML Validation Error 13 = Tag appearance more than once 14 = Tag specified out of required order 15 = Repeating group fields out of order 16 = Incorrect NumInGroup count for repeating group 17 = Non Data value includes field delimiter SOH character 18 = Invalid Unsupported Application Version 99 = Other
58	Text		
	Standard Trailer	Y	

Heartbeat (in/out)

A heartbeat message is sent at the interval set at Logon. It is also the response to a Test Request message.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 0
112	TestReqID		Identifier included in Test Request message to be returned in resulting Heartbeat. Required when the heartbeat is the result of a Test Request message.
	Standard Trailer	Y	

Test Request (in/out)

Test Request is used to “ping” the counterparty whenever a heartbeat has not arrived at the negotiated heartbeat interval.

Tag num	FIX Field name	Req'd	Comment
	Standard Header	Y	MsgType = 1
112	TestReqID	Y	Identifier included in Test Request message to be returned in resulting Heartbeat.
	Standard Trailer	Y	

FIX Drop Copy Sessions

Drop Copy Sessions

Drop Copy Sessions, or Drops, are used to replicate the private outbound order and trade flow for one or more participants, and/or users and/or markets.

A Drop Copy Session is not limited to the transactions entered via FIX. All orders and trades are provided regardless of input mechanism. For example, a Drop session is configured to send updates on all orders and trades for one or more participants. When an order is entered via Omnet(Web Trader), an Execution Report – Order Ack would be sent on the Drop session. A regular FIX session for the same participant in contrast, would not get order or trade acks for orders entered via Omnet.

The following business-level messages can be seen on a Drop session:

- Execution Reports
- Trade Capture Reports

NOTE: Rejects (on orders, cancels, cancel replaces and trade reports) will not be seen on the Drop.

Drop Copy Sessions are authenticated just like regular sessions using the Logon message with a username and password supplied.

All copied messages will have the CopyMsgIndicator (797) tag set to “Y”.

Drop Party Identifiers

To be able to identify the legal owner of an order or trade, all business-level messages on a drop will have the Parties block (Execution Reports, Trade Capture Reports) or RootParties block (used for OBO Trade Capture Reports). The Parties/RootParties block will contain:

453	NoPartyIDs			Set to 1 or 2
→	448	PartyID		Trader id or participant id.
→	447	PartyIDSource		Valid values: D = Proprietary /Custom code
→	452	PartyRole		Identifies the type or role of the PartyID (448) specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 17 = Contra Firm 28 = Custodian 36 = Entering Trader

Party Identifiers

The party identifier always consists of the two-character market code followed by the up to 5 character firm identifier. Example: A participant on the AIX (market code AI) with a firm ID of XYZ, would have the party identifier of “AIXYZ”.

Order and Trade Messages

This chapter contains listings of all business-level messages that are provided on the Drop Copy session.

Execution Report

The Execution Reports are message carrying order updates. Execution Reports are also used to carry fills information. Execution Reports are used for all types of orders including (availability of these types will depend on marketplace):

- updates and fills

Order identifiers

OrderID (tag 37) is the order identifier to be used on the Drop Copy session. It will remain the same during the lifetime of the order.

The commonly used ClOrdID (tag 11) is not used on the drop session, since it is specific to orders entered via FIX. The same is true for OrigClOrdID (tag 41), commonly used in FIX replace and cancel scenarios.

Order Cancels

Order cancels will be sent out as an unsolicited cancel on fix drop copy sessions.

Message Details – Execution Report

Tag	FIX tag name		Req'd	Comment
	Standard Header		Y	MsgType = 8
37	OrderID		Y	
453	NoPartyIDs			Repeating group used for business owner of order and/or optional Clearing Firm and Clearing Account.
→	448	PartyID	Q	Party identifier.
→	447	PartyIDSource	Q	Valid values: D = Proprietary/Custom code
→	452	PartyRole	Q	Identifies the type of role for the PartyID specified. Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 17 = Contra Firm 28 = Custodian 36 = Entering Trader
17	ExecID		Y	
880	TrdMatchID			Match ID assigned by the matching engine. Only set on fills.
150	ExecType		Y	Valid values: 0 = New 4 = Canceled 5 = Replaced 9 = Suspended C = Expired D = Restated F = Trade L = Activated by the system
39	OrdStatus		Y	Valid values: 0 = New 1 = Partially Filled 2 = Filled 4 = Canceled 5 = Replaced 9 = Suspended C = Expired
1	Account			This account field contains either the actual account number or various classifications. All possible values are listed below: Account number HOUSE DO FO FORU FOBY
55	Instrument/Symbol		Q	Sysmbol
48	Instrument/SecurityID		Q	Assigned by system. Orderbook ID
22	Instrument/SecurityIDSource		Q	Assigned by system. M = Marketplace-assigned identifier

Tag	FIX tag name	Req'd	Comment
32	LastQty	Y	Traded quantity
31	LastPx	Y	Trade Price
54	Side	Y	Valid values: 1 = Buy 2 = Sell 5 = Short Sell
38	OrderQty	Q	
40	OrdType	Q	Valid values: 1 = Market 2 = Limit
44	Price		
59	TimeInForce	Q	Valid values: 0 = Day 1 = Good Till Cancel (GTC) 3 = Immediate Or Cancel (IOC) 4 = Fill Or Kill (FoK) 6 = Good Till Date (GTD)
432	ExpireDate		Date of order expiration. Conditionally required if TimeInForce = GTD
528	OrderCapacity		Define account types. Valid values; BL = Broker Local BF = Broker Foreign CL = Client Local CF = Client Foreign CB = Client Belarus CR = Client Russian ML = Market Maker Local MF = Market Maker Foreign
151	LeavesQty	Y	Quantity open for further execution.
14	CumQty	Y	Total quantity filled.
6	AvgPx	Y	Note: Always set to 0.0
60	TransactTime	Q	
378	ExecRestatementReason		Set on unsolicited order updates/cancels and restatements of overnight orders. Valid values: 1 = GT renewal / restatement (no corporate action) 4 = Broker option 8 = Market (Exchange) option
70	AllocID		Free-text pass-thru field.
797	CopyMsgIndicator	Q	Will be set to 'Y' on all Drop Copy messages
111	MaxFloor		For hidden orders.
	Standard Trailer	Y	

Trade Capture Report

The Trade Capture Report (TCR) message is used for the following purposes:

- Confirmations on reported trades
- Confirmations on auto-matched orders
- Notifications (and deletes of notifications) to counterparty that somebody reported a trade with you as counterparty

Workflows and more detailed descriptions of how Trade Capture reports are used can be found in the Fix specification for order entry.

Trade identifiers

TradeID (1003) is an identifier unique per day and orderbook, assigned by the marketplace on confirmed trades. TradeID is formatted as a string containing two long values (Omnet ID and Clearing Trade number) separated by a colon “:”.

The TrdMatchID (880) contains the match id generated by the system. TrdMatchID will hold the 64 bit binary match id encoded as a 16 byte hex string. TrdMatchID is also present in Execution Reports for fills.

The encoding is performed according to RFC 2045 [4].

TradeReportID (571) is similar to the ClOrdID used for orders and executions. The marketplace sets its own TradeReportIDs on outbound TCRs (like confirmed trades).

TradeReportRefID (572) is used to refer to a previous TradeReportID. SecondaryTradeReportID (818) is set by the marketplace on Trade Capture Report and TCR Ack messages. The Secondary Trade Report ID carries the order_number.

Trade Capture Report features

To understand the business purpose of a TCR, two fields must be examined:

- TradeReportType (856)
- TradeReportTransType (487)

TradeReportType is set to 1 = Alleged on Notifications to counterparty and deletes of notifications. Use to TradeReportRefID to look up previous notification for deletes. For all other use cases TradeReportType is set to 0 = Submit.

TradeReportTransType is used to determine other uses of the TCR. The following values are used: 0 = New – set on trade confirmations both for reported trades and auto-matched trades.

Trade Types

The TrdType tag (828) is used to specify the type of trade being reported to the marketplace.

A list of supported Trade Type values is supplied separately by the marketplace. Note that there may be limitations on which trade types are allowed for a certain instrument and/or participant. It is out of scope of this document to fully specify all such rules. Please refer to the member trading rules for

further information. For valid values, please see “Appendix, Trade types” section in FIX Order Entry spec.

Message Details – Trade Capture Report

Tag	FIX tag name	Req'd	Comment
	Standard Header	Y	MsgType = AE
571	TradeReportID	Y	Identifier assigned by marketplace
1003	TradeID		Unique (in combination with series/instrument) identifier for trade
572	TradeReportRefID		Used to refer to original trade or notification in case of modifications or cancellations
818	SecondaryTradeReportID		order_number. Also present in previous TCR Ack message. For auto-matched trades order_number is in OrderID.
487	TradeReportTransType	Q	Valid values: 0 = New
856	TradeReportType	Q	Valid values: 0 = Submit 1 = Alleged
828	TrdType		For valid values, please see “Appendix, Trade types” section in FIX Order Entry spec.
829	TrdSubType	Q	Default value 1001 = Standard. The trade is a normally registered trade.
573	MatchStatus	Q	Valid values: 0 = Compared, matched or affirmed 1 = Uncompared, unmatched, or unaffirmed
880	TrdMatchID	Q	Match ID assigned by the matching engine.
570	PreviouslyReported	Y	Indicates if the trade capture report was previously reported to the counterparty Valid values: N = No
55	Symbol	Q	Symbol
48	SecurityID	Q	Assigned by system. Orderbook ID
22	SecurityIDSource	Q	Assigned by system. M = Marketplace-assigned identifier
32	LastQty	Y	Traded quantity
31	LastPx	Y	Trade Price
75	TradeDate	Y	Always set to date of trade.
60	TransactTime	Y	Contains Time of Trade Execution

Tag	FIX tag name			Req'd	Comment
552	NoSides			Y	Either 1 or 2 sides. Counterparty may not be shown on auto-matched trades.
→	54	Side		Y	Side. Valid values: 1 = Buy 2 = Sell 5 = Short Sell
→	37	OrderID		Y	Required in FIX. Set to “NONE”.
→	453	NoPartyIDs		Q	Number of party id entries
→	→	448	PartyID	Q	party identifier
→	→	447	PartyIDSource	Q	Valid values : D = Propr. Code
→	→	452	PartyRole	Q	Valid values: 1 = Executing Firm 7 = Entering Firm 12 = Executing Trader 17 = Contra Firm 28 = Custodian 36 = Entering Trader
→	528	OrderCapacity			Define account types. Valid values; BL = Broker Local BF = Broker Foreign CL = Client Local CF = Client Foreign CB = Client Belarus CR = Client Russian ML = Market Maker Local MF = Market Maker Foreign
→	483	TransBkdTime			Time of agreement. Only set on the own Side (where PartyRole=Executing Firm)
→	1	Account			
→	70	AllocID			
→	151	LeavesQty			Quantity open for further execution.
→	1057	AggressorIndicator			Indicates who is the aggressive party in the trade. Valid values: Y = Party is the aggressor N = Party is passive NOTE: Only set on the own Side (where PartyRole=Executing Firm)
855	SecondaryTrdType				Contains GENIUM deal_source value.
381	GrossTradeAmt				Calculated trade value
21102	DealNumber				Contains the numeric deal_number

Tag	FIX tag name	Req'd	Comment
797	CopyMsgIndicator		Set to 'Y' on Drop Copy messages
	Standard Trailer	Y	