CAT Reporting Technical Specifications for Plan Participants

9/10/2024

Version 4.1.0-r22

Table of Contents

Executive Summary

The Consolidated Audit Trail (CAT) tracks orders throughout their lifecycle and identifies the exchanges and broker-dealers handling them. This allows regulators to more efficiently and accurately track activity in eligible securities — those under the jurisdiction of the Securities and Exchange Commission (the "SEC") — throughout the U.S. markets. CAT is created by a joint plan (CAT NMS Plan) of the Plan Participants or simply "Participants."

This document provides Participants with the necessary information to fulfill their reporting obligations to CAT in compliance with SEC Rule 613 and the CAT NMS Plan. It describes the requirements for reporting data to CAT, including detailed information about data elements and file formats of each reportable event. This document also describes how Participants should submit files to CAT, including access instructions, network and transport options, and testing requirements.

This document does not include information related to the provision of data by Industry Members. The *CAT Reporting Technical Specifications for Industry Members* can be found on the CAT NMS website (www.catnmsplan.com).

Table 1: Summary of Document Revisions

See Appendix I for the Summary of Document Revisions for earlier versions.

Version	Date	Author	Description
4.1.0-r15	7/29/2022	FINRA CAT	*Spec Updates for Reject Message Event (RME):* • Added Section 3.7.4 for the new Reject Message Event (RME) • Added Section 10.6.7 for RejectMessageEvents file kind data ingestion feedback • Updated references to file kinds to include the new RejectMessageEvents file kind • Updated Appendix B.1: Data Ingestion Errors to reflect errors applicable to the new RejectMessageEvents file kind • Updated Appendix F: Data Dictionary to add new fields and to reference RME on existing fields as applicable *Spec Updates to* Support Use of FINRA ADF: • Added Section 6.3 for the new Equity Best Bid and Offer Event (EBBO) • Updated Appendix F: Data Dictionary to add new fields and to reference EBBO on existing fields as applicable
4.1.0-r16	11/21/2022	FINRA CAT	Moved Summary of Document Revision content for Releases 4.1.0-r9 through 4.1.0-r14 to Appendix I (changes not tracked) Updated *sentTimestamp* on Options Quote (OQ) event from Optional to Conditional; added language to clarify condition under which the field can be omitted

Version	Date	Author	Description
			Updated *sentTimestamp* on Options Quote Cancel (OQC) event from Optional to Conditional to better reflect description of field Updated Error Codes 7017, 7019, 7021, 7023 to clarify that they are Named Errors Updated description of 'FLOOR' *executionCode* to clarify that the value optionally may be reported beginning September 29, 2022 and must be reported beginning no later than November 14, 2022 Updated Section 9.8.4 and Appendix G to include information for Disaster Recovery Updated Appendix F: Data Dictionary to: • Add *cancelReasons* 156-160 and *executionCodes>liquidityCode* 53-57 for Nasdaq Mercury, ISE, and Gemini • Add *cancelReasons* for NASDAQ Mercury • Add *orderType* PrimaryPegAvailWhenLocked and PrimaryPegUnavailWhenLocked for MIAX PEARL Equities • Add *executionType* Name/Value Pairs SOT and STT for BOX • Updated *handlingInstructions* > crossType values 31 – 41 to apply to Mercury *The following changes will be effective in the Production Environment on December 5, 2022:* • Updated Appendix F: Data Dictionary to add *rejectReason* codes for BOX, CBOE, IEX, MEMX, MIAX, and NYSE • Updated Appendix B.3: Linkage Discovery errors updated to add Error Codes 6022, 6024, 6026, 6028 to support linkage of (exchange) OT to (Industry Member) MOOT *The following changes will be effective in the Production Environment in March 2023:* • Added information for new BestBidAndOffer file kind to support the reporting of the EBBO event
4.1.0-r17	2/21/2023	FINRA CAT	Updated Section 3.7.3 Supplemental Trade Event Added an example Reject Message Event in Section 3.7.4 Updated Appendix F: Data Dictionary to: • Add *cancelReason* for MIAX Pearl Equities • Add *handlingInstructions* for NYSE Equities (NoRetail) and MIAX and MIAX Emerald (AutoMatchLimit and AutoMatchMarket) • Add *orderAttributes* > PWASH value of 'X' for CBOE, ClientID for NYSE • Add *orderAttributes* > AIQ value of 'b', 'd', 'n', 'o', 'w', 'y' and clarifying descriptors for NASDAQ Equities • Add/remove *rejectReason* values for CBOE, LTSE, MEMX, NASDAQ Equities, and NYSE • Add *executionCodes/orderAttributes* for BOX (FLEX)
4.1.0-r18	4/17/2023	FINRA CAT	Updated Summary of Document Revisions for version 4.1.0-r17 to include the date of publication Updated Tables 63, 101, and 102 to include Equity Best Bid and Offer (EBBO): • Table 63: replaced #2 from 'exchange' to 'marketCenterId' data type and description

Version	Date	Author	Description
			• Table 101: added 'Display-Only Facility' to heading; #1-6, and 8 added 'Order Accepted' and 'Equity Best Bid and Offer' • Table 102: #1-6 added 'Equity Best Bid and Offer' Updated Appendix F: Data Dictionary to: • Add *handlingInstructions* > CxlPxBack value of 'N', 'C', 'U' • Add *marketCenterId* > ADF for EBBO events; updated description of 'D' to 'ADF-TRF' • Add *orderAttributes* > OverrideAIQDLO boolean value for IEX • Add *orderAttributes* > orgID name/value and PriorityUpdate boolean value for Nasdaq Equities • Add *rejectReason* for NYSE American Options • Add *rejectReason* for Nasdaq Options • Add *originalAskQuoteID* description
4.1.0-r19	7/18/2023	FINRA CAT	Updated Summary of Document Revisions for version 4.1.0-r18 to include the date of publication; made minor formatting changes for consistency Clarified submission requirements: • Section 9.1.1, Table 79: File Kinds: added 'Record Type' • Section 10.9 Corrections, Deletions, and Replacements: clarified correction records; added 'Correction Submission Schedule' table; updated section title; removed list of events in the first sentence; added cross-reference to section 10.8.1 • Section 10.9.1.1 Delete Records: removed outdated footnote (16) • Section 10.9.2 File Replacement: clarified late file submission and number of replacements; added cross-reference to section 10.8.1 Updated Appendix F: Data Dictionary to: • Add *handlingInstructions* > SigVersion value of 'SignalV5" and "SignalV6" for IEX • Add *orderAttributes* > FloorOrderID value for Cboe (C1) • Add and Update *rejectReason* for MIAX Options, MIAX Pearl Options, and MIAX Emerald Options • Add *rejectReason* for Nasdaq Options
4.1.0-r20	9/25/2023	FINRA CAT	Added *initiator* field to section 5.1.1, Table 32: Quote Events Updated Appendix F: Data Dictionary to: Onboard MEMXOP: • Add *Participant ID* > Members Options Exchange value of 'MEMXOP' • Add *handlingInstructions* > Price Adjustment value of 'PA' • Add *orderAttributes* > values of 'PAF', 'PAB', 'MTP' • Add *cancelReason* > values of '0 – 19' • Add *exchOriginCode* > values of '1 – 7' • Add *rejectReason* > various values in the range of '1001 – 3019' Other Updates: • Add *initiator* > Event(s) description 'Option Quote Event' • Add *rejectReason* > 'PEARLEQ_ORR_0078' for MIAX Pearl Equities • Update *rejectReason* > 'PEARLEQ_ORR_0019' for MIAX Pearl Equities

Version	Date	Author	Description
			• Add *cancelReason* > ‘PEARLEQ_0126’ for MIAX Pearl Equities • Add *handlingInstructions* > ‘27 MEMXOP’ to DestExch value for Nasdaq Options • Update *orderType* > ‘FMPEG’ for IEX • Add *cancelReason* > ‘EMLD_0065’ for MIAX Emerald Options • Update *orderAttributes* > REJA value of ‘x’ description for Cboe
4.1.0-r21	4/15/2024	FINRA CAT	• Updated Appendix F: Data Dictionary to:Add *cancelReason* > ‘MIAMI_0065’ for MIAX Options • Add *cancelReason* > ‘PEARL_0039’ for MIAX Pearl Equities • Add *cancelReason* > ‘21’ for MEMX • Add *cancelReason* > ’85-88’ for PHLX and NOM Options • Add *cancelReason* > ’161-164’ for ISE and GEMX Options • Add *cancelReason* > ’1521-1523’ for MRX and NOM Options • Add *cancelReason* > ’PostOnlyCancelled’ for IEX • Add *cancelReason* > ’1017, 1018, 1020 – 1155 1019, 1156 - 1513’ for GEMX Options • Add *definedNoteData* > ‘AucPrc, DMM’ for NYSE Equities • Added FINRA CAT standard *executionCodes* and *orderAttributes* > ‘PCTPX’ • Added *executionCodes* > *‘Y, W’* for IEX • Add *handlingInstructions* > ‘Dest Exch’ value of ‘28’ for Nasdaq Options • Add *handlingInstructions* > ‘RoutingInst’ value of ‘0, 1, 2’ for IEX • Add *handlingInstructions* > ‘ExecBroker’ value of ‘CUSTOM-RFTY and LSTY’ for NSDQ • Add *handlingInstructions* > '*CrossType’* value of ‘42’ for NOBO, MRX, GEMX Options • Remove *handlingInstructions* > ‘NoRetail’ for NYSE Options • Add *noteType* > ‘AOCNoParticipation, AOCNoParticipationRej’ for NYSE Equities • Added *orderAttributes* > ‘REJA’ values of ‘w and Q’ for Cboe Options • Added *orderAttributes* > ‘CancelOrSlide’ values of ‘0, 3’ for IEX • Add *rejectReason* > ‘117-120’ for PHLX and NOM Options • Add *rejectReason* > ‘569-572’ for ISE and GEMX Options • Add *rejectReason* > ‘1521-1523’ for MERX and NOBO Options • Add *rejectReason* > ‘311-325, 326, 327, 906, 330, 331’ for NYSE Equities • Add *rejectReason* > ‘328, 329, 330’ for AMEROP Options • Add *rejectReason* > ‘329’ for ARCAOP Options • Add *rejectReason* > ‘1-16, 1017-1520, Other’ for GEMX Options • Add *rejectReason* > ‘InvalidTagCombinationForPostOnlyOrder’, ‘InvalidOddLot’ and ‘InvalidTradeNowInstruction’ for IEX • Add *rejectReason* > ‘PEARLEQ_MRR_0052 and ‘PEARLEQ_MRR_0053’ for MIAX Pearl Options

Version	Date	Author	Description
			• Aligned *rejectReason* and *orderAttributes* for NYSE Options to support Pillar platform transition

4.1.0-r22	9/10/2024	FINRA CAT	• Add clarification to sections:
			  ○ 2.1.1 > ‘Member Alias values that are used as a Routing Party for intervenue linkage must be seven (7) or fewer characters’
			  ○ 3.4 > ‘The text string must be a Member Alias of seven (7) or fewer characters.’
			  ○ 4.13 > ‘Previous Trade Key: date, exchange, symbol, refTradeID’
			  ○ 5.5 > ‘Previous Trade Key: date, exchange, optionID, refTradeID’
			• Add *cancelReason* > ‘0,1,4-21’ for LTSE
			• Add *cancelReason* > ‘PEARLEQ_0021-0028, 0127-0134’ for MIAX PEARLEQ
			• Add *cancelReason* > ‘SPHR_0004, 0005, 0007, 0012, 0018, 0029-0038, 0041-0044, 0061-0063’ for MIAX SPHR
			• Add *exchOr*iginCode > ‘1-6’ for MIAX SPHR
			• Add *executionCodes* > ‘B, K’ for IEX
			• Add *handlingInstructions* > ‘PegO, RML, RMO, RP, RSV’ for LTSE
			• Add *handlingInstructions* > ‘DirectedTo_ALGO’ for NYSE Equities
			• Add *orderAttributes* > ‘R, RDM, RRT, RPF, RBH, STP’ for LTSE
			• Remove *handlingInstructions* > ‘CUBEAUCF, FloorTradeNamesLater, FloorTradeNamesLaterAllocation, NOW, ND, NR, PNP, PNP+, PNPB, PNPLO’ for NYSE Options
			• Remove *handlingInstructions* > ‘ExecInst, PriceSliding, TradeThruExemptReason, MatchTradePrevention, MTPSublevelInd’ for NYSE CHX
			• Remove *handlingInstructions* > ‘NoMPL, NoMPL-IOI’ for NYSE Equities
			• Update *handlingInstructions* > descriptions for NYSE Options and Equities
			• Remove *orderAttributes* > Legacy as of 11/2019 for NYSE CHX
			• Remove *orderAttributes* > ‘Border, CROWD, MaxDiscVol, SOrder, STP, SelfTrade, MinTriggerSize, MinPegSize, MaxDiscVol, CeilingFloorPrice, DiscPriceRange’ and all ‘TypeOfInterest’ values for NYSE Equities
			• Update *orderAttriburtes*> descriptions for NYSE Options and Equities
			• Add *Participant ID* > ‘SPHR’ for MIAX Sapphire Options Exchange
			• Add *rejectReason* > ‘qa, qc-qe, qi, qm-qo, ql, qr-qt, qx-qz, qC-qF, qK-qM, qO-qS, qU, qW for Cboe Options
			• Add *rejectReason* > ‘RoutingNotAllowed’ for IEX
			• Add *rejectReason* > ‘1001-1003, 1006, 1018-1020, 1022-1023, 1027, 1099-1181, 2001, 2003, 2006, 2018, 2099-2105, 2107-2131, 3000-3011’ for LTSE
			• Add *rejectReason* > ‘PEARLEQ_MRR_0054 – 0057’, ‘PEARLEQ_ORR_0079 – 0089’, ‘PEARLEQ_ORR_0120 – 0125’ for MIAX PEARLEQ
			• Add *rejectReason* > ‘PEARL_FOR_0000 – 0008’, ‘PEARL_FOR_0011’, ‘PEARL_FCR_0000 – 0003’ for MIAX PEARL

Version	Date	Author	Description
			• *Add rejectReason* > 'SPHR_ERR_0000 – 1009, 1011 – 1021, 1024 – 1036, 1039 – 1044, 1046 – 1048, 1051, 1053 – 1077, 1079, 1081, 2001, 2102, 2105, 2108, 2110, 2119, 3000 – 3007, 3010, 3012 – 3015, 4000 – 4001, 4500, 4502, 4504 – 4506, 4509 – 4515, 4520, 4525 – 4527, 5100 – 5103', 'SPHR_FOR_0000 – 0011', 'SPHR_FCR_0000 – 0003' for MIAX SPHR • Add *rejectReason* > '332-335' for NYSE Equities and Options • Update *rejectReason* > description for '250' for NYSE Equities and Options • Remove *rejectReason* > '1-568', 'OTHER' for Nasdaq GEMX

1. Introduction

1.1. CAT Overview

The Securities and Exchange Commission (SEC) approved Rule 613 under the Securities Exchange Act of 1934, which requires national securities exchanges and national securities associations (collectively, the Participants) to submit a national market system plan to create, implement, and maintain a consolidated audit trail (CAT NMS Plan) that would capture customer and order event information for orders in NMS Securities and OTC Equity Securities (Eligible Securities), across all markets, from the time of order inception through routing, cancellation, modification, execution, and allocation. The SEC approved the CAT NMS Plan on November 15, 2016.

In accordance with SEC Rule 613, the CAT NMS Plan requires a Central Repository that will comprehensively track orders throughout their lifecycle and identify the Participants and Industry Members handling them, as well as the account holders and authorized traders for any account that originates an order (Customers [1]). Specific data elements will be submitted to the Central Repository by Participants, Industry Members, and CAT Reporting Agents. CAT Reporting Agents may be third-party firms reporting on behalf of other entities, or may be outside parties that are not required to submit data to the CAT, but from which the CAT may receive data per the CAT NMS Plan, such as the Securities Information Processors (SIPs).

The CAT NMS Plan also requires the selection of an entity as the Plan Processor to be responsible for performing the processing functions required by Rule 613 and the Plan. The Operating Committee of Consolidated Audit Trail, LLC, a governing body composed of representatives of the Participants, oversees the operation of the CAT. The duties of the Operating Committee are further described in Article IV of the CAT NMS Plan.

Refer to SEC Rule 613, available at: https://www.sec.gov/rules/final/2012/34-67457.pdf for more details. Refer also to CAT NMS Plan, available at: https://www.catnmsplan.com/wp-content/uploads/2018/02/34-79318-exhibit-a.pdf.

---

[1] Customers are defined in SEC Rule 613(j)(3) as: (i) the account holder(s) of the account at a registered broker-dealer originating the order; and (ii) any person from whom the broker-dealer is authorized to accept trading instructions for such account, if different from the account holder(s).

1.2. Change Release Management Process

Changes to this technical specification will be released as follows:

- Prior to the go-live date for system changes
  - A new specification will be posted to the CAT Public Website
  - A notice will be posted on the website with a summary of changes and links to relevant information.
  - One or more email alerts will be sent to plan participants with a summary of changes and links to relevant information.
  - In some cases, CAT may accept production reporting using the new specification in advance of the go-live date.
  - Plan Participants that have not conducted testing or production reporting using the new technical specification format will receive support from CAT as the go-live date approaches.
- The new technical specification will include a summary list of changes as well as a table listing the specific areas of the document where the changes have been made.

1.3. CAT Identifiers

CAT uses a number of identifiers, many of which readily convey their meaning from the context in which they are used. The subsections below include terms associated with the entities that will report data into CAT and their respective roles. As shown in the diagram below, Exchange ID is a subset of Participant ID, which is a subset of Reporter ID.

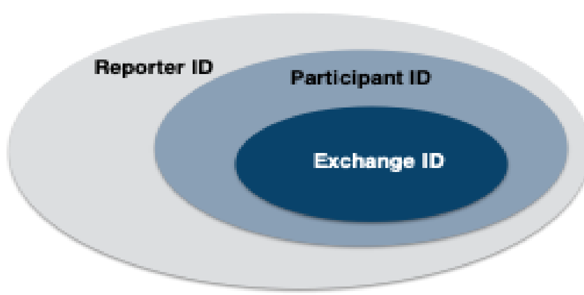

Figure 1: CAT Identifier Hierarchy

1.3.1. CAT Reporter ID

Each entity which reports into CAT will be assigned a unique identifier: a CAT Reporter ID. This ID will uniquely identify each reporter, including plan participants, industry members, and associated reporting facilities. The database of CAT Reporter IDs will be made available both as a downloadable file on the CAT website and through the web portal API.

1.3.2. Participant ID

The Participant ID is an ID assigned by CAT to each plan participant. The value will be the same as the participant's CAT Reporter ID.

1.3.3. Exchange ID

The Exchange ID is an ID assigned by CAT to each stock/options exchange. The actual value will be the same as the exchange Participant ID and Reporter ID, but, as indicated in Figure 1, Exchange ID is a subset of Participant ID, which is a subset of Reporter ID.

1.3.4. Member Alias

Each SRO will assign unique IDs to its industry members. These IDs are aliases for CAT reporters so that reporting firms can use existing identifiers when reporting market events to CAT. It is important that both the member and SRO are aware of the assigned IDs and when they should be used in reports to CAT.

Each SRO has autonomy in assigning their IDs. Note that the same ID could possibly be assigned to different industry members across SROs. Furthermore, a member may have multiple aliases assigned to them by the same SRO. Thus, the alias is only valid in combination with the SRO that assigned the ID. Specifically, when an exchange receives a routed order from one of its members, both the routing member and the exchange must report the same Member Alias in their reports to CAT in order to properly link the reports to the same order lifecycle.

An industry member can have the same alias value assigned by multiple SROs. Note that an alias is used in conjunction with an identifier that links the alias with the SRO that assigned the alias (either by explicit designation, or implicitly by context).

For example, consider three firms (Firm A, Firm B, and Firm C) and three SRO participants (Participant A, Participant B, and Participant C), and the following table of SRO-assigned member IDs.

Table 2: Example of SRO-assigned Member IDs

FIRM	Participant A	Participant B	Participant C
Firm A	FRMA	AAAA	FRMA
Firm B	FRMB		BBBB
Firm C	FRMC	CCCC	FRMB

Note that Member Alias FRMA is assigned to Firm A by both Participant A and Participant C, and Member Alias FRMB is assigned to two different firms by two different participants. While the same alias is used multiple times, these are valid mappings because the same alias is not assigned multiple times within a participant. Also note that Firm B is not a member of Participant B, and so there is no corresponding mapping.

Thus, each firm will have at least one alias for each SRO in which they have membership. The value may or may not be the same across all participants. When Participant A refers to Firm C, it will use the alias FRMC. Likewise, when Firm C refers to itself in relation to Participant A, it will use the alias FRMC.

Note that industry members can have multiple Member Aliases, but they will also be assigned a unique CAT Reporter ID. CAT maps the SRO-assigned Member Alias values to ensure the same unique CAT Reporter ID assigned to the member firm across SRO's. Note that member dictionary entries apply to data uploaded for the same business date as the member dictionary itself (values do not have to be the same from day to day).

1.4. Fundamental Data Types

The fundamental data types used in this document are described below. A complete list of data types is presented in Appendix F. Data Dictionary.

CAT will accept two kinds of text-based files: JSON and CSV. To support both JSON and CSV submissions, CAT will publish a JSON schema file which describes each data type with required representation formats, and a mapping that defines the position in a CSV representation that the data element would assume.

A schema will be provided for each data object that can be reported in both JSON and CSV.

When a data field is marked as either optional or conditional, some records may not provide values for that field. In such a case, the field is simply not reported as part of the JSON record. In a CSV record, it is reported as an empty column. [2]

[2] For each CSV record, all fields up to and including the furthest of the last required field and last provided field for the record must be included. For example, consider the NOTE event in Section 3.7.1 where a number of fields are conditional or optional, including the last four fields. Each field through `noteType` (the last required field) must be provided (either with data or as an empty column). If one of the remaining non-required fields is provided, such as `Note`, then all values through `Note` must be provided.

Table 3: Data Type Descriptions

Data Type	JSON Type	Description
Alphanumeric	STRING	A string, composed only of letters and digits [a-zA-Z0-9]. When an Alphanumeric type is described, it will include a number, indicating the maximum length of the field. For example, Alphanumeric(7) means that the field can contain up to 7 characters
Array of XXX	ARRAY	When represented in JSON, it is an array of the indicated type (XXX is a placeholder). So, Array of Unsigned would be an array of unsigned integers, and would be represented as `[ 0, 42 ]`. When represented in CSV, it is a series of the indicated type, separated by the pipe symbol. So, the aforementioned array of Unsigned would be represented as `0|42`.
Boolean	BOOLEAN	A value with only two choices: true or false
Choice	STRING	A Text field, but with an explicit list of acceptable values.
Date	NUMBER	An 8-digit integer representing the date in YYYYMMDD.
Exchange ID	STRING	A subclass of Participant ID that only applies to exchanges (all participants except FINRA)
Integer	NUMBER	An integer value (positive, negative, or zero), with no decimal fraction component, in the inclusive range from −9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 (the same range as a 64-bit signed integer)
Member Alias	STRING	Text(8) - one of the aliases assigned by an SRO to one of its members
Message Type	STRING	An Alphanumeric(5) field, indicating the type of message being reported
Name Value Pairs	STRING JSON Object	A value of type Text (except the pipe is allowed), composed as described in the Name Value Pairs section below
Numeric	NUMBER	A general numeric type, composed of digits, an optional decimal point, followed by more digits (with an optional leading +/- sign). These values, while looking like floating point numbers, should always be read and processed in a way that represents the exact value as represented by the text. Examples: 1235, -1235, 1235.67, -1235.67 When a numeric type is described in this document, it will include two numbers, the first is the maximum number of digits before the decimal point, and the second is the maximum number of digits after the decimal point. For example, Numeric(6,4) means that the number can have up to 6 digits before the decimal point and up to 4 digits after the decimal point (visual format would be ######.####). Note that these are maximum limits - the lengths can be smaller. Valid examples which comply with Numeric(6,4) would be -999999.9999, -0.1, 0, 0.0001, and 999999.99. All numeric values must have a whole number portion before the decimal point (*e.g.*, 0.25 can't be represented as .25). The fractional portion is optional. Do not use leading zeros in numeric values. A zero should only appear as the first digit if it is the only digit before the decimal point (*e.g.*, 0.75)

Data Type	JSON Type	Description
Participant ID	STRING	A subclass of Reporter ID that applies only to participants
Price	NUMBER	A Price is shorthand for Numeric(10,8), which can support prices in the inclusive range [-9999999999.99999999, 9999999999.99999999]
Reporter ID	STRING	Alphanumeric(7) - a CAT Reporter ID
Symbol	STRING	Text (20)
Symbol Alias	STRING	Text (20) - an alias that can be assigned to a symbol
Text	STRING	A string, composed of any printable character, except comma (ASCII decimal 44, hex 2C), pipe (ASCII decimal 124, hex 7C), and double quote (ASCII decimal 34, hex 22). When a Text type is described, it will include a number, indicating the maximum length of the field. For example, Text(7) means that the field can contain up to 7 characters.
Time	NUMBER	A numeric field, with a specific format conforming to what the ISO 8601 standard calls the *basic format*, with a few extra specifications. All 24-hour time components are mandatory (*i.e.*, hour, minute, and second as HHMMSS). The decimal-fraction part must be separated from the whole part with a period (ASCII decimal 46, hex 2E), and can contain up to 9 digits (to represent nanosecond component). The time zone is always Eastern Time. For example, 09:30:00.123456789 would be reported as `093000.123456789`.
Timestamp	STRING NUMBER	A timestamp represents a moment in time, and contains both Date and Time, separated by the letter T (ASCII decimal 84, hex 54) or a space (ASCII decimal 32, hex 20). All time must be in Eastern Time. For example, January 7, 2017 21:30:00.123456789 in New York would be represented as the string `20170107T213000.123456789`. As an alternative format, the timestamp can be submitted as a value of type Unsigned, representing the number of nanoseconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday,1 January 1970, not counting leap seconds. This is also commonly known as POSIX time or UNIX time. The same point in time from the above example would be represented as the number 1483842600123456789. Note that the data type is different between the two formats. In JSON, the first representation requires it to be surrounded by double quotes, while the second does not
Unsigned	NUMBER	An unsigned value, greater than or equal to zero, with no decimal fraction component, in the inclusive range from 0 to 18,446,744,073,709,551,615 (the same range as a 64-bit unsigned integer)

1.4.1. Data Validation

All data submitted to CAT will be validated based on the defined data type of each item, including proper formatting and range checking. All File Names, Field Names, and Field Values are case sensitive. During validations, if the case does not match, an error will occur. Examples of accepted values are detailed in the table above. Valid values for Choice fields are defined in the Data Dictionary for each data element. Valid data values, ranges, and formats will be specified in the record schema files, which will be used to validate submitted data element values. Records and values which fail validation will be marked as a failure and will be reported as feedback to the Submitting Member as detailed in Section 10.

1.4.2. Name Value Pairs

Some fields are described as containing name/value pairs. Name Value Pairs is a list of zero or more attributes, where each attribute is either a name with no value, or a name with an accompanying value such that the name and value are separated by a single equal sign (ASCII decimal 61, hex 3D). Multiple attributes are separated by the pipe symbol (ASCII decimal 124, hex 7C). If an attribute is Boolean in nature, it can optionally be represented as a name alone, where its value is implied by its presence (true) or absence (false).

The name part is the string up to the first pipe symbol or equal sign. Names must not contain commas (ASCII 44, hex 2C), pipes, equal-signs, or double-quotes (ASCII decimal 34, hex 22).

If the name terminates with a pipe, it is a Boolean value, and its presence indicates true. If the name terminates with an equal sign, the value must follow.

The value part is the string starting with the character just after the equal sign, up to either a pipe symbol or the end of the string. Values may contain an equal sign, but must not contain commas, pipes or double-quotes.

In some cases, the names are free-format (i.e., undefined). Both the name and any value are left up to the discretion of the reporter and the contents are not validated by CAT.

For example, the following JSON represents a hypothetical name/value pair field, with a Boolean attribute and a price attribute: `{ "data": "XYZ|ABC=12.55" }`

The above format works for both JSON and CSV data entry. However, when submitting data in JSON, a more native JSON style can optionally be used by assigning a JSON object as the value for a Name Value Pair attribute. Note, however, that Boolean values must be explicitly set. The above example can alternatively be submitted as: `{ "data": { "XYZ": true, "ABC": 12.55 } }`

2. Reference Data

This section describes the reference or supplemental data required to be reported by each participant.

2.1. Member Information

Each SRO must submit to CAT a directory of information that lists each industry member with which it has a reporting relationship. Each dictionary entry identifies a specific industry member, and assigns one or more IDs to that member. These IDs may be used by the SRO and/or the member when reporting order events to CAT. The industry members listed in the dictionary will also be participant members of the SRO, although this is not always the case. For example, each industry member that submits an order to an exchange must be a registered member of that exchange. However, the exchange may route orders to an industry member that is not a member of that exchange. In either case, the exchange must give at least one Member Alias to each industry member that appears in any of the order events reported to CAT.

Each member may have multiple aliases, but a specific Member Alias may only be assigned once per SRO. Note that the member dictionary is loaded each day, and the values only apply to that trading day. Thus, Member Aliases could be reassigned on subsequent trading days.

The Member Dictionary will be uploaded as a file of newline-delimited JSON objects, one object per member entry. The member dictionary is necessary to process other file uploads, and must be uploaded to CAT no later than T+1 at 4:00 a.m. ET, with entries sufficient to support all reports submitted on that trading day. Note that this is a same-day upload requirement whereas order events are required to be reported by 8:00 a.m. ET the following trading day.

CAT must be able to identify each IMID that operates as an ATS and must know the name under which the ATS operates. The Member Dictionary Entry does not support the provision of this information. When reporting a Member Dictionary Entry (MDE) that includes an ATS, a corresponding Member Alias Detail Entry (MADE) must also be submitted. Additionally, when reporting a Member Dictionary Entry (MDE) where the status is ‘Other’, a Member Alias Detail Entry (MADE) must also be submitted to provide the entity name. Note that, while any participant can submit a MADE record, it is intended for use only by participants that report ATSs to CAT, such as FINRA.

2.1.1.Member Dictionary Entry

Table 4: Member Dictionary Entry

Member Dictionary Entry (MDE)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	MDE	R
2	reporter	Reporter ID	The unique identifier assigned to the reporter by CAT	R
3	ID	Text (20)	The CRD number of the firm, if the status field directly below is set to Active, Inactive, or NonMember. Otherwise (Internal, Other), this must be an ID for the entity generated by the reporter.	R
4	status	Choice	The status of the member for the reporting date. If the status is 'Other', a corresponding MADE record must be reported for each member alias. See Data Dictionary: `status`	R
5	memberAliases	Array of Member Alias	A list of Member Alias values for the member, as assigned by this SRO, for use in association with this SRO. A corresponding MADE record must be reported for each member alias representing an ATS.	R

Member Alias values that are used as a Routing Party for intervenue linkage must be seven (7) or fewer characters.

The following example shows a potential member dictionary for exchange Exch1 where the first entry represents an industry member that is also a member of the reporting SRO, the second entry represents an industry member that is not a member of the reporting SRO, and the third entry represents the SRO itself, with various facilities that have been given Member Alias values.


```
{
  "type": "MDE",
  "reporter": "Exch1",
  "ID": "1234567",
  "status": "Active",
  "memberAliases": [ "FRMA", "FRMA1", "FRMA:U1", "FRMA:U2" ]
}
{
  "type": "MDE",
  "reporter": "Exch1",
  "ID": "7654321",
  "status": "NonMember",
  "memberAliases": [ "FRMB" ]
}
```

```
{
  "type": "MDE",
  "reporter": "Exch1",
  "ID": "123xyz",
  "status": "Internal",
  "memberAliases": [ "XXX" ]
}
{
  "type": "MDE",
  "reporter": "Exch1",
  "ID": "123abc",
  "status": "Internal",
  "memberAliases": [ "ZZZ" ]
}
```

The next example shows a potential member dictionary for exchange Exch2. Note how the same entities are members of both Exch1 and Exch2, but they may or may not have different Member Alias values with each SRO.

```
{
  "type": "MDE",
  "reporter": "Exch2",
  "ID": "1234567",
  "memberAliases": [ "FRMZ", "FRMZ:U1", "FRMZ:U2" ],
  "status": "Active"
}
{
  "type": "MDE",
  "reporter": "Exch2",
  "ID": "7654321",
  "memberAliases": [ "FRMB" ],
  "status": "Active"
}
```

2.1.2.Member Alias Detail Entry

Table 5: Member Alias Detail Entry

Member Alias Detail Entry (MADE)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	MADE	R
2	reporter	Reporter ID	The unique identifier assigned to the reporter by CAT.	R
3	ID	Text (20)	The CRD number of the firm for which the entry is being provided, or the ID for the entity generated by the reporter. Must also appear in an MDE event.	R
4	memberAlias	Member Alias	Member Alias of the member for the MADE submission. Must also appear in an MDE event for the corresponding ID.	R

Member Alias Detail Entry (MADE)				
#	Field Name	Data Type	Description	Include Key
5	ats	Boolean	Indicates that the memberAlias is an ATS.	C
6	name	Text (128)	The doing-business-as (DBA) name of the ATS; required when ats is true. OR The name of the entity; required when the status of the corresponding MDE is 'Other'.	R

The following example shows a member dictionary for exchange Exch2 where one of the aliases represents an ATS. Note that the ID and memberAlias in the MADE record matches the data provided in the MDE record.

```
{
  "type": "MDE",
  "reporter": "Exch2",
  "ID": "7654321",
  "status": "Active"
  "memberAliases": [ "ATSA", "FRMA", "FRMA1", "FRMA:U01", "FRMA:U02" ]
}
{
  "type": "MADE",
  "reporter": "Exch2",
  "ID": "7654321",
  "memberAlias": "ATSA",
  "ats": true
  "name": "ATS Alpha"
}
```

2.2. Equity Symbols

FINRA CAT maintains a symbol master for CAT reportable equity securities. [3]

2.2.1.CAT Symbol Master

The CAT Equity Securities Symbol Master is published on the CAT NMS website at https://www.catnmsplan.com/reference-data, for use by Industry Members in reporting. FINRA CAT publishes a Start-of-Day (SOD) file by 6:00 a.m. ET daily, an End of Day (EOD) file by 6:00 p.m. ET, and intraday updates approximately every two hours during the business day. Information including file descriptions, file layouts, and retrieval instructions can also be found on that page. Additionally, the data is available to the Plan Participants via the CAT Query Tools.

[3] The symbol master is maintained based on a data feed provided by FINRA independently from FINRA's reporting obligation as a CAT Plan Participant.

2.3. Corporate Actions

FINRA CAT provides details for equity corporate actions impacting equities [4] and options [5].

2.4. Options Dictionary

Naming conventions for options can vary among exchanges and trading firms. To reduce confusion and simplify reporting, CAT allows reporters to submit options reports using a unique ID of type Text(40), as defined by the reporter, for each option. However, each reporter must upload a dictionary every day for which it reports option quote/order events. The dictionary is valid only for events reported on the same business day.

The options dictionary shall include simple option entries and complex option entries, to cover all options utilized in any report submitted to CAT by that reporter on a given date. This file is composed of a series of dictionary entries for each option, with the Option ID that will be used by the reporter for all option reports done on that day.

Each Option ID defined in the dictionary must be unique for that reporter on that day, across all simple and complex options. As for reportable order events, Options Dictionary entries can be uploaded throughout the day. When uploaded files are processed, option dictionary files are processed before any order event files for the same uploaded timeframe. Entries can be added dynamically throughout the day.

Note that this is not the product definition, but a universal way to reference an options product for the purposes of reporting order events to CAT.

While the Options Dictionary is necessary to process plan participant data, the Processor uses Options Clearing Corporation (OCC) files to generate the Start of Day and End of Day Options Symbol Master files available to support reporting by Industry Members. In the event an exchange supports options products that are not available in the OCC data set (e.g., FLEXPCT products), CAT requests that the exchange submit those products to CAT by 6:00 p.m. ET on T+0, for inclusion in the End of Day Options files for industry member reporting.

The options dictionary is uploaded as a file of newline delimited JSON objects.

---

[4] Corporate Actions for listed equities are received via a data feed provided by FINRA independently from FINRA’s reporting obligation as a CAT Plan Participant.

[5] Corporate Actions for listed options are retrieved from the Options Clearing Corporation (OCC).

2.4.1.Option Series Dictionary Entry

The dictionary mapping for an option series (i.e., flex or simple) will contain the information provided in Table 6: Simple Option Series Dictionary Entry, which allows options events to be reported using the Option ID reported in the dictionary entry.

Table 6: Simple Option Series Dictionary Entry

Simple Option Series Dictionary Entry (OSDE)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OSDE	R
2	reporter	Reporter ID	The unique identifier assigned to the reporter by CAT	R
3	optionID	Text (40)	The unique ID assigned to this option by this reporter. No other simple/complex/flex option should receive the same ID. All reports from this reporter will use this ID to reference a particular option product	R
4	kind	Choice	Specifies if an option is a simple, complex, flex, or percentage denominated flex option. For the value FLEXPCT, the strike price and order prices of the option are in percentages. See Data Dictionary: `kind`	R
5	optionsSymbol	Text (14)	The option class or symbol for the series (as known by OCC)	R
6	primaryDeliverable	Symbol	The symbol for the primary deliverable component of the option, provided in the symbology of the listing exchange or a valid alias.	R
7	underlyingType	Choice	This field specifies whether a simple option series has an equity or index as its underlying. The underlying type mapping is consistent with the same mapping used at OCC (e.g., ETF is treated as Equity and WCO is treated as Index). See Data Dictionary: `underlyingType`	R
8	expirationDate	Date	The date that the contract will expire	R
9	strikePrice	Numeric (10,8)	The dollar and decimal value of the strike price. If option kind = FLEXPCT, this will be the percentage	R
10	putCall	Choice	Specifies if this simple option or option leg is a put or call. See Data Dictionary: `putCall`	R
11	exerciseStyle	Choice	Specifies the exercise style of the Option Series See Data Dictionary: `exerciseStyle`	R

Simple Option Series Dictionary Entry (OSDE)				
#	Field Name	Data Type	Description	Include Key
12	settlement	Choice	Specifies the settlement of the option See Data Dictionary: `settlement`	R
13	testSeriesFlag	Boolean	Indicates that the entry represents a test symbol. Events submitted for a test symbol are excluded from linkage processing.	C

For example, the following dictionary entry would be for the January 19, 2018 150.0 Put for BRK class B. Note that the primary deliverable is reported in NYSE symbology because BRK.B is listed on NYSE.

```
{
  "type": "OSDE",
  "reporter": "MYID",
  "optionID": "12345",
  "kind": "Standard",
  "optionsSymbol": "BRKB",
  "primaryDeliverable": "BRK.B",
  "underlyingType": "Equity",
  "expirationDate": 20180119,
  "strikePrice": 150.00,
  "putCall": "Put",
  "exerciseStyle": "American",
  "settlement": "PM"
}
```

2.4.2.Option Symbol Changes

Changes to symbols stemming from corporate actions can be handled by reporters using Dictionary Entries. Each options exchange should ensure that on the effective date for a corporate action, its Dictionary Entries accurately reflect option symbols with the appropriate numerical suffix when applicable, and it includes any new option symbols created as the result of the corporate action. A detailed corporate action example follows:

> Stock ABCD undergoes a 2 for 1 stock split on June 1, 2018. All strike prices are halved, the deliverable remains 100 and the symbol is unchanged. On August 1, 2018 stock ABCD spins off company EFGH, 10 shares per 100 ABCD owned. On the market opening at ex-date all open interest in ABCD corp. is moved to symbol ABCD1 delivering 100 shares of ABCD and 10 shares of EFGH. Option symbol ABCD1 = 100 ABCD + 10 EFGH. Subsequently, ABCD and EFGH shares are each listed in the underlying cash market and their prices are used in the valuation of options ABCD1 respectively. The options exchanges list new option contracts for each underlying that deliver 100 shares using symbols ABCD and EFGH (assuming listing

criteria is met). Options symbols ABCD and EFGH begin trading (independently) and each delivers 100 shares of the corresponding stock upon exercise. On November 1, 2018 ABCD undergoes a 3 for 2 stock split. Option contracts in ABCD and ABCD1 are affected. Contracts in ABCD become ABCD2 delivering 150 shares of underlying stock ABCD. Option symbol ABCD2 = 150 ABCD. Contracts in ABCD1 remain ABCD1 and deliver 150 shares ABCD and 10 shares EFGH. Option symbol ABCD1 = 150 ABCD + 10 EFGH. The exchange will again list a new ABCD delivering 100 shares of ABCD stock upon exercise.

Considering the example above, the two entries below demonstrate the values before and after the first corporate action event:

Stock ABCD undergoes a 2 for 1 stock split on June 1, 2018. All strike prices are halved, the deliverable remains 100 and the symbol is unchanged.

Before 2:1 Stock Split on June 1, 2018

```
"type": "OSDE",
"reporter": "MYID",
"optionID": "4322",
"kind": "Standard",
"optionsSymbol": "ABCD",
"primaryDeliverable": "ABCD",
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 45.00,
"putCall": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
```

After 2:1 Stock Split on June 1, 2018

```
{
"type": "OSDE",
"reporter": "MYID",
"optionID": "4322",
"kind": "Standard",
"optionsSymbol": "ABCD",
"primaryDeliverable": "ABCD",
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 22.50,
"putCall": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
```

The next entries demonstrate the impact of the second corporate action event – the spinoff on August 1, 2018.

On August 1, 2018 stock ABCD spins off company EFGH, 10 shares per 100 ABCD owned. On the market opening at ex-date all open interest in ABCD corp. is moved to symbol ABCD1 delivering 100 shares of ABCD and 10 shares of EFGH. Option symbol ABCD1 = 100 ABCD + 10 EFGH. Subsequently, ABCD and EFGH shares are each listed in the underlying cash market and their prices are used in the valuation of options ABCD1 respectively. The options exchanges list new option contracts for each underlying that deliver 100 shares using symbols ABCD and EFGH (assuming listing criteria is met). Options symbols ABCD and EFGH begin trading (independently) and each delivers 100 shares of the corresponding stock upon exercise.

Before Spinoff - Note that at this time, EFGH is still part of ABCD.

```
{
"type": "OSDE",
"reporter": "MYID",
"optionID": "4322",
"kind": "Standard",
"optionsSymbol": "ABCD",
"primaryDeliverable": "ABCD",
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 45.00,
"putCall": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
```

After Spinoff – three Dictionary Entries would now be reported as the result of this corporate action:

```
{
"type": "OSDE",
"reporter": "MYID",
"optionID": "4322",
"kind": "Non-Standard",
"optionsSymbol": "ABCD1",
"primaryDeliverable": "ABCD",
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 22.50,
"putCall": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
{
"type": "OSDE",
"reporter": "MYID",
"optionID": "99123",
"kind": "Standard",
"optionsSymbol": "EFGH",
"primaryDeliverable": "EFGH",
"underlyingType": "Equity",
```

```
"expirationDate": 20181221,
"strikePrice": 5.00,
"type": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
{
"type": "OSDE",
"reporter": "MYID",
"optionID": 99124,
"kind": "Standard",
"optionsSymbol": "ABCD",
"primaryDeliverable": "ABCD",
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 17.50,
"putCall": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
```

The pre- and post-Spinoff JSON Dictionary Entries shown above are also shown in table format below.

Table 7: Pre- and Post-Spinoff JSON Dictionary Entries

Field Name	Pre-Spinoff Value	Post-Spinoff		
		Entry #1 Value	Entry #2 Value	Entry #3 Value
Exchange ID	CBOE	CBOE	CBOE	CBOE
Option ID	4322	4322	99123 (new unique id)	99124 (new unique id)
Option Kind	Standard	Non-standard	Standard	Standard
Underlying Type	Equity	Equity	Equity	Equity
Primary Deliverable	ABCD	ABCD	EFGH	ABCD
Option Symbol	ABCD or ABCD181221C00022500 *Note: EFGH is still part of parent company ABCD*	ABCD1 or ABCD181221C00022500 *Note: Delivery components of ABCD1 include 10 shares of EFGH. CAT will know this since ABCD1 is the*	EFGH or EFGH81221C00005000 *Note: This a new standard option as of Aug 1, 2018 which delivers 100 shares of the new standalone*	ABCD or ABCD181221C00017000 *Note: This is a new standard option as of Aug 1 2018, which delivers 100 shares of the parent company*

Field Name	Pre-Spinoff Value	Post-Spinoff		
		Entry #1 Value	Entry #2 Value	Entry #3 Value
		symbol used by OCC.	*company EFGH. Investors will price the underlying and the options accordingly.*	*ABCD that remains after EFGH was spun off. Investors will price the underlying and the options accordingly.*
Expiration Date	20181221	20181221	20181221	20181221
Option Put/Call Code	C	C	C	C
Strike Price	22.50	22.50	5.00	17.50
Exercise Style	American	American	American	American
Settlement	PM	PM	PM	PM

A final example demonstrates the impact of the third corporate action event – the stock split on November 1, 2018.

> On November 1, 2018 ABCD undergoes a 3 for 2 stock split. Option contracts in ABCD and ABCD1 are affected. Contracts in ABCD become ABCD2 delivering 150 shares of underlying stock ABCD. Option symbol ABCD2 = 150 ABCD. Contracts in ABCD1 remain ABCD1 and deliver 150 shares ABCD and 10 shares EFGH. Option symbol ABCD1 = 150 ABCD + 10 EFGH. The exchange will again list a new ABCD delivering 100 shares of ABCD stock upon exercise.

Before 3:2 Stock Split -- ABCD delivers 100 shares of ABCD. ABCD1 options deliver 100 shares of ABCD + 10 shares EFGH.

```
{
"type": "OSDE",
"reporter": "MYID",
"optionID": "4322",
"kind": "Non-Standard",
"optionsSymbol": "ABCD1",
"primaryDeliverable": "ABCD",
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 22.50,
"putCall": "Call",
```

```
"exerciseStyle": "American",
"settlement": "PM"
}
{
"type": "OSDE",
"reporter": "MYID",
"optionID": "99124",
"kind": "Standard",
"optionsSymbol": "ABCD",
"primaryDeliverable": "ABCD",
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 22.50,
"putCall": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
```


After 3:2 Stock Split - ABCD becomes ABCD2 and delivers 150 shares of ABCD. Symbol ABCD1 remains, though now delivers 150 shares ABCD and 10 shares EFGH. The exchange lists new, standard ABCD options that deliver 100 shares of ABCD.


```
{
"type": "OSDE",
"reporter": "MYID",
"optionID": "4322",
"kind": "Non-Standard",
"optionsSymbol": "ABCD1",
"primaryDeliverable": "ABCD”,
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 22.50,
"putCall": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
{
"type": "OSDE",
"reporter": "MYID",
"optionID": "99124",
"kind": "Non-Standard",
"optionsSymbol": "ABCD2",
"primaryDeliverable": "ABCD",
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 22.50,
"putCall": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
{
"type": "OSDE",
"reporter": "MYID",
"optionID": 100501,
"kind": "Standard",
```

```
"optionsSymbol": "ABCD",
"primaryDeliverable": "ABCD",
"underlyingType": "Equity",
"expirationDate": 20181221,
"strikePrice": 15.00,
"putCall": "Call",
"exerciseStyle": "American",
"settlement": "PM"
}
```


2.4.3.Complex Option Dictionary Entry

The dictionary mapping for a complex option will contain the information presented in the table below. Each complex option can contain multiple legs, where each leg is either an option leg or a stock leg (stock leg will generically refer to equity/exchange-traded fund "ETF").

Table 8: Complex Option Dictionary Entries

Complex Option Dictionary Entry (CODE)					
#	**Field Name**		**Data Type**	**Description**	**Include Key**
1	type		Message Type	CODE	R
2	reporter		Reporter ID	The unique identifier assigned to the reporter by CAT	R
3	optionID		Text (40)	The unique ID assigned to this option by this reporter. No other simple/complex/flex option should receive the same ID. All reports from this reporter will use this ID to reference a particular option product	R
4	kind		Choice	Specifies if an option is a simple, complex, flex, or percentage denominated flex option. For this message type, the kind will always be “Complex”. See Data Dictionary: `kind`	R
5	groupID		Text (40)	An identifier supplied by the user/reporter, to be associated with this entry. The value of the field is not checked by CAT, but it will be stored, and can be used to search for dictionary entries that have the same value	O
6	legs	legType	Choice	Defines the type of leg. See Data Dictionary: `kind`	R
		side	Choice	The side of the order: See Data Dictionary: `side`	R
		ratio	Unsigned	The ratio quantity for this leg, relative to the other legs. For option legs, the ratios must already be reduced to the smallest units possible	R

Complex Option Dictionary Entry (CODE)					
#	**Field Name**		**Data Type**	**Description**	**Include Key**
		optionID	Text (40)	The ID of the option - **for option legs only**. Note that the Option ID for the leg must have already been uploaded before using it in the definition of a complex option. Furthermore, the combination of Option ID / Side must be unique among all legs	C
		symbol	Symbol	The symbol of the equity, in the symbology of the listing exchange - **for equity legs only**. The same symbol must not appear in more than one leg. Multiple symbol legs are only allowed for index options only	C
7	testSeriesFlag		Boolean	Indicates that the entry represents a test symbol. Events submitted for a test symbol are excluded from linkage processing.	C

The Option ID must be unique. Duplicate dictionary entries are ignored. Entries that have the same Option ID, but different details are rejected. Any entry which defines the opposite side of an existing entry will be rejected. For example, a complex option dictionary entry to Buy one (1) contract of option 1234 and Sell two (2) contracts of option 4321 is considered to be the "opposite side" of an entry to Sell one (1) contract of option 1234 and Buy two (2) contracts of 4321. Thus, if both were submitted the second would be rejected.

JSON Example

```
{
  "type": "CODE",
  "reporter": "MYID",
  "kind": "Complex",
  "optionID": "98765",
  "legs": [
    {
      "legType": "Option",
      "side": "Buy",
      "ratio": 1,
      "optionID": "121345"
    },
    {
      "legType": "Equity",
      "side": "Buy",
      "ratio": 100,
      "symbol": "ABCD"
    }
  ]
}
```

JSON Example of reject

```
{
  "type": "CODE", "reporter": "MYID", "kind": "Complex",
  "optionID": "98765",
  "legs": [
    { "legType": "Option", "side": "Buy",
      "ratio": 1, "optionID": "121345"
    },
    { "legType": "Option", "side": "Sell",
      "ratio": 2, "optionID": "99999"
    }
  ]
}
{
  "type": "CODE", "reporter": "MYID", "kind": "Complex",
  "optionID": "56789",
  "legs": [
    { "legType": "Option", "side": "Sell",
      "ratio": 1, "optionID": "121345"
    },
    { "legType": "Option", "side": "Buy",
      "ratio": 2, "optionID": "99999"
    }
  ]
}
```

2.5. Market Maker Information

Each Equity SRO must submit to CAT a directory of information that lists industry member with which it has a reporting relationship and where the member makes a market in one or more equity symbols. Each Market Maker Dictionary entry identifies a specific industry member, symbol, market maker type, status and status time. The Market Maker may make a market in one or more symbols.

The Market Maker dictionary is loaded each day, and the data values only apply to that trading day. The Market Makers, their assigned Symbols, Market Maker Type, and Status may change on subsequent trading days.

The Market Maker Dictionary will be uploaded as a file of newline-delimited JSON objects, one object per Market Maker entry. The Market Maker dictionary must be uploaded to CAT no later than T+1 at 4:00 a.m. ET.

Table 9: Market Maker Dictionary Entry

Market Maker Dictionary Entry (MMDE)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	MMDE	R
2	reporter	Reporter ID	The unique identifier assigned to the reporter by CAT	R
3	marketMaker	Member Alias	Identifier assigned to a member by the SRO as provided in the Member Dictionary	R
4	symbol	Symbol	Equity Symbol in listing market format, in which the Market Maker is registered on the exchange	R
5	marketMakerType	Market Maker Type	A list of exchange defined values for the Equity Market Maker distinguishing between types or designations of market makers. See Data Dictionary: `marketMakerType`	R
6	marketMakerStatus	Choice	The status of the member/symbol for the reporting date. For details, see the Data Dictionary entry for Status See Data Dictionary: `marketMakerStatus`	R
7	statusTime	Timestamp	Time of change in market maker's status. If one record for a member alias and symbol combination is provided, it is assumed to be active for the entire day. For market making initiations not at the open, provide the start time.	R
8	definedMMDEData	Name/Value Pair	A list of key/value pairs, providing machine parseable exchange specific regulatory context data for the Equity Market Maker. The attributes are not defined in the spec, and can be any values as long as they conform to the format for a list of name/value pairs as defined under the Fundamental Data Types section of this document.	O

The following example shows a Market Maker for exchange Exch1 where the first entry represents an industry member with one active symbol and another inactive.

```
{
  "type": "MMDE",
  "reporter": "Exch1",
  "marketMaker": "ABCD",
  "symbol": "PZ",
  "marketMakerType": "MM",
  "status": "Active",
  "statusTime": "20200714022015.123456789"
```

```
}
{
  "type": "MMDE",
  "reporter": "Exch1",
  "marketMaker": "ABCD",
  "symbol": "PX",
  "marketMakerType": "MM",
  "status": "Inactive",
  "statusTime": "20200714022216.123456789"
}
```

3. Special Data Elements and Common Events

This section describes data elements that are common to most order events, including timestamps, sequence numbers, symbols, material terms of an order, and elements used during the CAT process of creating order lifecycles.

Events that are universal, or common, are also described in this section.

3.1. Timestamps and Sequence Numbers

All timestamps are required to be reported in the greatest granularity in use by the reporter's trading platform, up to nanoseconds. While the timestamp generally allows the system to properly sequence events within the lifecycle of an order event, it is possible for multiple events to have the same timestamp, especially if the granularity of the reported timestamp is insufficient. In these cases, the system cannot confidently sequence the events by timestamp alone. When it is possible for multiple events --- from the same reporter, on the same day [6], in the same symbol --- to have the same timestamp, a sequence number must also be provided for each event.

The sequence number is required to be strictly increasing, and must guarantee proper sequencing of events in the order in which they originally occurred. The sequence number may be globally unique, in which case it provides sequencing unilaterally; however, this is not required. The sequence number does not sequence events across multiple reporters.

The system only uses the sequence number if two or more events have the same timestamp. If the timestamp alone provides the ability to determine the proper order of the events, the sequence number does not need to be reported.

3.1.1. Sequence Number Subsystems

The purpose of the sequence number is to allow regulators to sequence multiple events that have the same timestamp. However, reports for the same reporter/date/symbol may originate from multiple systems, and it may be difficult to coordinate a sequence number that is unique among all subsystems.

In such cases, a sequence number subsystem (`seqNumSub`) can be optionally reported along with the sequence number. This value can be examined to better determine ordering characteristics of the events that have the same timestamp.

[6] For purposes of 24-hour trading, a “day” is considered to be a single cycle date. See the definition of `cycleDate` in Appendix F: Data Dictionary for additional details.

3.1.2. Time of Order Receipt

The time of order receipt is the time at which an exchange Participant assigns an Order-ID to an incoming message.

3.2. Symbology

When reporting events for equities, the symbol must be reported in the symbology of the listing exchange or using an alternate symbology identified in the symbol master as described in Section 2.2.

Any reporter who reports options events must submit an option dictionary to CAT. All options are identified using the Option ID, as provided to CAT in the reporter's option dictionary.

3.3. NBBO

The NBBO is provided with each relevant order event (i.e., when available). This is the NBBO from the perspective of the reporter at the time of the event, but not including the effect that the event would have on the NBBO. For example, if the NBBO were 100@10.10 x 100@10.15, and a new order arrived at the exchange to BUY 100@10.10, the reported NBBO would be 100@10.10 x 100@10.15, even though the immediate effect of the order would be to change the best bid to 200@10.10.

Note that the bid/ask prices are required, but the quantities being bid or offered are optional.

There exist some special cases where the NBBO is unavailable or nonexistent. In those cases, the NBBO values should be reported with a zero price and zero quantity. An entry with both the price and quantity of zero will indicate that the data was either unavailable or not applicable for that particular event. Note that the values can't just be reported as unavailable because it is hard to acquire them. They must truly be unavailable or not applicable to that particular event. NBBO prices are not required for leg-level events of complex orders and zero may be provided in lieu of a quote price.

3.4. Order Linkage and Lifecycle

When all members have submitted their reports to CAT for a given trading day, CAT will link all reportable events to create a complete lifecycle of each order. A key part of being able to connect the orders is recognizing and connecting the daisy chain of orders across all CAT reporters. In order to accomplish this, both the reporter routing an order away and the reporter accepting the order must report the exact same details about the order.

Of particular interest to reporting participants, the data elements important to creating cross-reporter order linkages are: Exchange ID, Date, Symbol/Option, Routing Party, Routed Order ID, and Session ID.

When an order is routed to an exchange, each communication protocol specifies a way to uniquely identify that order (e.g., FIX protocol calls it ClOrdId, OUCH calls it Order Token). However, the uniqueness guarantees differ from protocol to protocol. Some exchanges may assign a unique Member Alias for each account, and require uniqueness based on the account ID and order ID alone. Others may issue special identifiers for each API session that the member uses to connect into the exchange. Since there is no universally accepted method, CAT uses a combination of several different attributes that provide flexibility in ensuring globally unique order IDs across all known supported protocols.

Both the routing firm — once industry member reporting has commenced — and the exchange will submit information to CAT in their Order Route and Order Accepted reports. Note that exchange and industry member Routed Order ID, Routing Party, and Session ID must exactly match between in order for CAT to accomplish the linkage process.

The Routed Order ID is the unique order identifier sent in the API message going from the routing entity to the destination entity.

The Routing Party is a text string that the exchange has assigned to the firm routing the order. Complexity arises when a member is assigned multiple values by the exchange. The determination as to which value is used by both parties depends on protocol-specific information. The text string must be a Member Alias of seven (7) or fewer characters. It can be any string, so long as both the sender of the order and the exchange agree on using the same string for their orders.

The Session ID is also exchange-assigned, usually a unique login account, an actual protocol session name, IP/port combination, or some other means of identifying a particular API session. The Session ID identifies the specific session used to route the order. Even in cases where there is only one session in use between reporters, the same non-empty value must be reported in the session field by both parties.

CAT, in cooperation with each exchange, shall determine how the Routing Party, Routed Order ID, and Session ID are derived for each API supported by the exchange. This guidance will be documented and published on the CAT website.

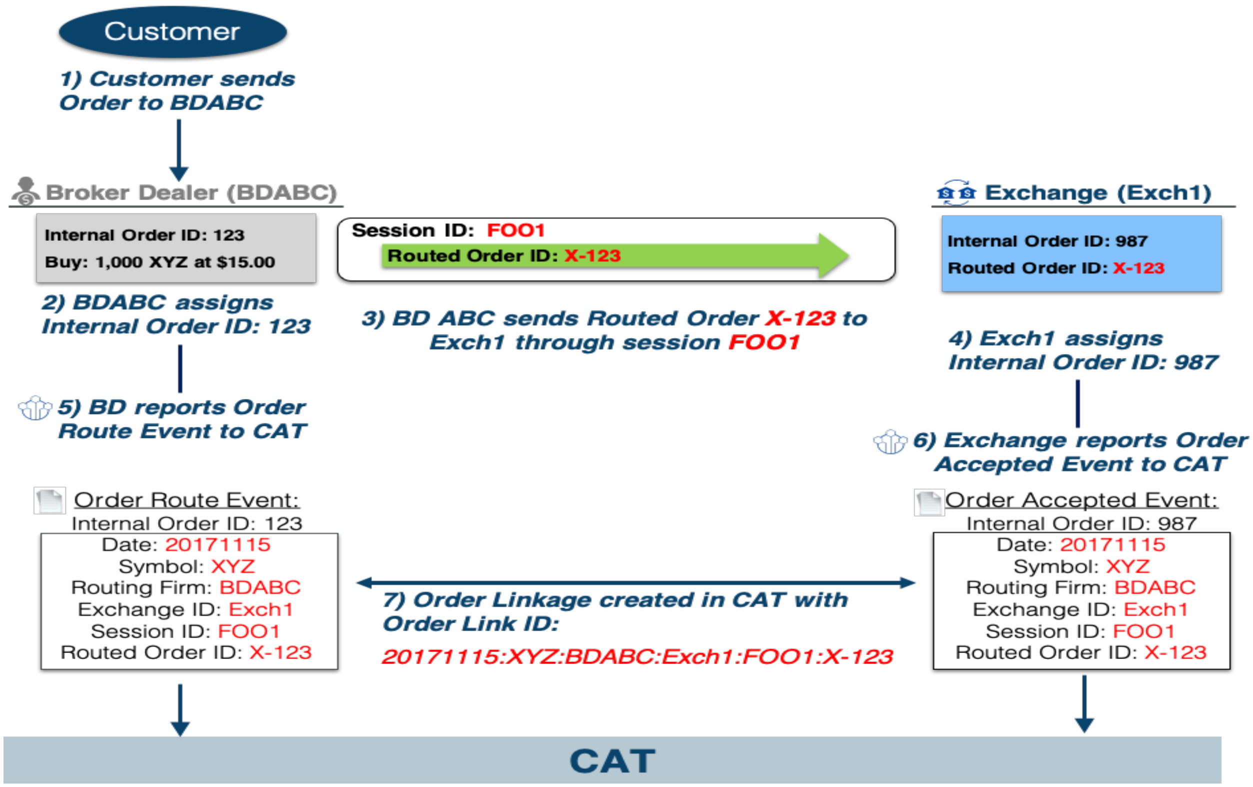

Figure 2: Order Linkage and Lifecycle

3.5. Material Terms of an Order

The material terms of an order include but are not limited to price, quantity, side, order type, open/close indicator (for options), time in force, and special handling instructions. Each order event includes fields for each of these.

However, each exchange offers significant distinguishing features and instructions to describe how orders are to be handled. These differences are mainly captured in the possible values for the order type and any special handling instructions. The CAT system is generally agnostic to these values, and their primary utility is in how they are interpreted and used in surveillance activities.

In order to provide utility in using the reported data for surveillance purposes, both the reporters and the users must have well known definitions of the data being reported. In addition, without specific definitions, the submitted data cannot be checked for integrity in those fields that comprise the material terms of an order. Thus, every possible value for each field must be explicitly defined both in this specification and the separate specification document for industry members[7]. Every value that could possibly be reported must

[7] Industry Members must also report the material terms of the order on their route reports

be well-defined in the technical specifications. CAT maintains the technical specifications for both the participants and industry members to reflect changes to order types and/or handling instructions over time. Each exchange must provide guidance to CAT on how these values are determined for each of their system interfaces, with lead time sufficient to allow CAT to update the specifications for both participants and industry members.

3.5.1. Order Types

The Order Type for each order must be assigned with exactly one value from a predefined set of choices. These choices are documented in the data dictionary entry for Order Type (see Appendix F). CAT, in cooperation with each exchange, has defined a list of acceptable values for this field, however additional order types may be added to accommodate future market needs.

The CAT website contains guidance on how these choices can be determined for each exchange API.

3.5.2. Order Handling Instructions

The Handling Instructions field defines special instructions as to how the order should be handled by the exchange. Neither SEC Rule 613, nor the CAT NMS Plan dictate the special handling instructions that must be supported. Furthermore, each exchange may use different names and values to describe how orders are handled, and there can be numerous customized special handling instructions. While the CAT processor must be able to support any instructions which are required to be reported, mandating specific instructions is beyond the scope of the CAT processor as that information is only known by the exchanges and the appropriate surveillance and regulatory entities. Thus, the allowed values for this field support a wide array of special handling instructions. Order Handling Instructions' values must be documented in the data dictionary of this technical specification, and guidance must be provided to CAT by reporters for how these values can be determined based on each exchange API. Guidance will be subsequently posted on the CAT website.

The Handling Instructions field can specify as many special handling instructions as apply for that order (or be empty if no such instructions apply). Thus, the handling instructions field will be a list of name/value pair.

Note that the full intent of the order is reportable to CAT. At a minimum, every term and/or instruction for an order that is communicated to the exchange must be reported to CAT. It can be reported as part of the standard set of material terms, or via one of the defined name/value pairs as defined in the Handling Instructions section of the Data Dictionary. Reporters cannot choose which order instructions to report: they must report every instruction applicable to each order.

Note that the Order Handling Instructions field is marked as 'conditionally required' in the event definitions, because its existence is not enforced by the system. If the order does not have any characteristics that are reportable to CAT, then the field does not have to be provided. However, if there are any explicit or implied handling instructions for the order, then this effectively becomes a required field, as all instructions must be reported.

For example, assume two hypothetical handling instructions: AON and WDS=<percent>; where AON means all-or-none and WDS means a discretion price is allowed to be less than or equal to some percentage of the spread. If an order were to be placed as all-or-none, with a discretion of up to 50 percent of the spread, then the Order Handling Instructions field would contain "`AON|WDS=50`" as its value.

This approach provides flexibility for exchanges, enabling them to represent a wide array of handling instructions, while also enabling CAT to validate submitted data and providing regulators a defined structure for interpretation of the data.

3.6. Optional, Required, and Conditional Fields

Subsequent sections describe event types and their fields. Each field will be notated with the abbreviation R, O, C, or r to represent whether it is required, optional, conditional, or required conditionally. This codification will be present in the last column of each table describing an event.

Table 10: Optional, Required and Conditional Fields

Type	Abbreviation	Description
Optional	O	Optional for the event, may be included at the discretion of the reporter
Conditional	C	Conditional fields may be required depending on the contents of the event. For example: in the note event, quoteID and orderID are conditional fields. If the note event is on a quote, then quoteID is required, if the note event is on an order, then orderID is required
Required	R	Required for the event, must always be included. For example, the field "type" is always required.
Required Conditionally	r	This is a special category of fields that currently applies to options only. Specifically, fields marked as 'r' are required if the event applies to a simple option order, but they are conditional if the event applies to an option order that is part of a complex order

3.7. Common Events

3.7.1.Note Event

The Note Event is a generic event that accommodates reporting for events that are not defined with explicit events. For example, there could be certain events that occur in the process of handling an order on the floor of an exchange that may be desired to be included in the trail of events for a particular order, but don't fit into an explicitly defined reportable event. In another example, there could be a certain process that the order goes through as part of its handling that does not constitute a change in terms of the order, but may be beneficial as part of the order's audit trail.

The Note event requires either an Order ID or a Quote ID (but not both), so that the notation can be appropriately linked by CAT to the associated order/quote. If the note relates to a stock order, then both orderID and symbol are required. If the note relates to an option order/quote then both optionID and orderID/quoteID are required.

Table 11: Note Event

Note (NOTE)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	NOTE	R
2	reporter	Reporter ID	The identifier for the reporter that generated the note	R
3	eventTimestamp	Timestamp	The date/time of the event being noted	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The symbol of order; for a stock order	C
7	optionID	Text (40)	The ID of the option; for an option order/quote	C
8	quoteID	Text (40)	The ID of the quote on which the note is being placed, only applicable if the note is related to a quote	C
9	orderID	Text (40)	The ID of the order on which the note is being placed, only applicable if the note is related to an order	C

Note (NOTE)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
10	noteType	Choice	One of several predefined types of notation events, providing a way to classify or categorize notations. See Data Dictionary: `noteType`	R
11	definedNoteData	Name Value Pairs	A list of key/value pairs, providing machine parseable data for the notation. See Data Dictionary: `definedNoteData`	O
12	undefinedNoteData	Name Value Pairs	A list of key/value pairs, providing machine parseable data for the notation. The attributes are not defined in the spec, and can be any values as long as they conform to the format for a list of name/value pairs as defined under the Fundamental Data Types section of this document.	O
13	note	Text (255)	A free-form text field to describe the notation for the event	O
14	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

The Note Type and Defined Note Data fields are well-defined and must conform to the permitted values as described in this specification. The Undefined Note Data can accommodate any attributes, as long as the field conforms to the format for a list of name/value pairs.

Thus, Note Events, while generic in nature, can be parsed and evaluated by both humans and computer programs.

Linkage Keys for **NOTE**:

- **Order Key:** date, reporter, symbol, orderID
- **Order Key:** date, reporter, optionID, orderID
- **Quote Key:** date, reporter, optionID, quoteID

3.7.2.Self-Help Declarations

"Self-help" declarations allow market participants to disregard the protected quotations of trading centers that are experiencing systems problems such as failure, material delay, or malfunction.

Participants must report to CAT any self-help declarations they make. If a self-help declaration is carried over to the next day, it must be reported again on that day. The following data is required to be reported for Self-Help declarations:

Table 12: Self-Help Declaration

Self-Help Declaration (SHD)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	SHD	R
2	reporter	Reporter ID	Identifier of reporter declaring self-help	R
3	declaredTimestamp	Timestamp	Date and time self-help was declared	C
4	revokedTimestamp	Timestamp	Date and time self-help was revoked. Self-help declarations must be reported each day. If self-help is not revoked by the end of the day, this field may be left unreported or can be set to the closing time. However, another self-help event must be reported for the next day	C
5	awayExchange	Exchange ID	Exchange affected by self-help event	R
6	comments	Text (255)	Comments related to self-help event	O
7	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date	C

Self-Help Declaration (SHD)				
#	Field Name	Data Type	Description	Include Key
			including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	

Both the declared and revoked timestamps can be reported in one single event by including both declaredTimestamp and revokedTimestamp. Alternatively, the declaration and revocation can be reported independently by just including the relevant timestamp in separate events.

3.7.3.Supplemental Trade Event

Each trade event (stock and option) contains some information which may not be readily available when generating the trade event. Thus, an independent event can be submitted to augment the information in the trade event. These events can be submitted in the same file as other events or in a separate file.

These events will not be recorded as separate events in CAT. Rather, the information in these events will be merged with the appropriate trade event to provide data that may have been missing in the original trade event. Currently, only the saleCondition can be reported in this way. Supplemental Trade Events must be received within the four day processing window (e.g., by T+4 at 8:00 a.m. ET) in order to update the trade event.

This event is used for stock and option trades. If the trade references a stock, then the symbol field must be provided. If the trade references an option, then the optionID field must be provided.

The description uses "trade" in a general manner. If the event references a trade, the tradeID field is required. If the event references a fill, the fillID and side are required.

Table 13: Supplemental Trade Event

Supplemental Trade Event (STE)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	STE	R
2	exchange	Exchange ID	The ID of the exchange where the trade took place	R
3	tradeID	Text (40)	The tradeID from the original trade event	C
4	fillID	Text (40)	The fillID from the original fill event	C
5	optionID	Text (40)	The ID of the option being traded	C
6	symbol	Symbol	The symbol for the stock being traded	C
7	side	Choice	Side of the executed trade (required when fillID is used) See Data Dictionary: `side`	C
8	saleCondition	Text (8)	Conditions under which trade was executed	R
9	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **STE**:

- **Trade Key:** date, exchange, symbol, tradeID
- **Trade Key:** date, exchange, optionID, tradeID
- **Fill Key:** date, exchange, symbol, fillID

- **Fill Key:** date, exchange, optionID, fillID

3.7.4.Reject Message Event

This event may be submitted in the Test Environment beginning on November 14, 2022 and in the Production Environment beginning on December 5, 2022.

The Reject Message Event is reported when an exchange rejects a message, for example, when an exchange received a message that could not be accepted by the receiving system. This event is not intended for malformed messages. It is intended for well-formed messages that fail validation and are assigned a reject **reason by the receiver.** See JSON example below:


```
{
  "type":"RME",
  "exchange":"EXCH1",
  "symbol":"SYM",
  "eventTimestamp":"20221110T120000.000000",
  "sequenceNumber":58,
  "seqNumSub":"random",
  "exchangeInternalID":"123456",
  "routingParty":"ABCD",
  "routedOrderID":"orderID",
  "session":"ssn1",
  "routedOriginalOrderID":"12345678",
  "rejectReason":52,
  "rejectContext":[{"exampleContext":"random"}],
  "member":"Mem01",
  "cycleDate":"20221110"
}
```


A single event structure is defined for both equities and options, and it applies to both order-related and quote-related messages.

Table 14: Reject Message Event

Reject Message Event (RME)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	RME	R
2	exchange	Exchange ID	The ID for the exchange which received the rejected message.	R
3	symbol	Symbol	Symbol for the stock being traded. Required for equity messages, including equity orders; stock legs of multi-leg orders; and stock	C

Reject Message Event (RME)				
#	Field Name	Data Type	Description	Include Key
			legs for cross orders. Must match the symbol on the Industry Member's route event. Required for all stock-related events unless the submitted value was missing or unreadable. For simple option messages with a single underlying equity, the stock symbol should be provided unless the submitted value was missing or unreadable.	
4	optionID	Text (40)	The ID of the option being traded. Required for all rejections related to a simple option unless the submitted value was missing or unreadable. Required for all rejections related to a complex option unless the submitted value was not yet defined for the strategy, missing or unreadable.	C
5	eventTimestamp	Timestamp	The date/time of reject event.	R
6	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps.	C
7	seqNumSub	Text (10)	A sequence number subsystem identifier assigned to the system that rejected the message. Required if the process or processes rejecting messages are different from the process assigning eventTimestamp values and sequenceNumber values to other events.	C
8	exchangeInternalID	Text (40)	The internal ID assigned to the order or quote by the exchange. Required if the message is related to an existing order (such as a quantity reduction), and the nature of the rejection allows the exchange to match the rejected message to the existing order.	C
9	routingParty	Text (20)	The ID string used to identify the entity that routed the rejected message to the exchange. Must match the senderIMID on the Industry Member's route event. Required unless the submitted value was missing or unreadable.	C
10	routedOrderID	Text (40)	The order ID that the firm used in the API message when they sent the rejected message to the exchange (e.g., in FIX it would be ClOrdId, in OUCH it would be Order Token). Must match the routedOrderID value from the Industry Member's route event.	C

Reject Message Event (RME)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			Required unless the submitted value was missing or unreadable.	
11	session	Text (40)	The ID assigned to the specific session that the routing member used to route the rejected message to the exchange. Must match the session value from the Industry Member's route event. Required unless the submitted value was missing or unreadable.	C
12	routedOriginalOrderID	Text (40)	The ID for the order being modified, as sent by the routing broker in the original route message, or the most recent modify message (in FIX OrigClOrdID, in OUCH Existing Order Token). Required if the message can be connected to an order that exists within the system.	C
13	rejectReason	Choice	Code representing the reason why the order was rejected. Codes are exchange-specific. See Data Dictionary for the list of allowed values.	R
14	rejectContext	Name/Value Pairs	A list of key/value pairs, providing machine parseable data for the notation. The attributes are not defined in the spec, and can be any values as long as they conform to the format for a list of name/value pairs. Any additional information can be provided in this field. For example, the entire rejected message in Tag=Value format.	O
15	member	Member Alias	The identifier for the member firm that is responsible for the order.	C
16	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin	C

Reject Message Event (RME)				
#	Field Name	Data Type	Description	Include Key
			date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	

The Processor will attempt linkage for all Reject Message events. If all Lifecycles Keys are present (as defined below), a link will be made. If all Lifecycle Keys are not present, no link will be made. Reject Messages events that fail to link will not generate feedback; nor will they be included on report card statistics.

Lifecycle Keys for this event:

- Equities
  - **Order Key:** date, exchange, symbol, exchangeInternalID
  - **Quote Key:** date, exchange, symbol, exchangeInternalID
  - **Route Link Key:** date, symbol, routingParty, routedOrderID, session, exchange
- Simple Options Orders and Quotes:
  - **Order Key:** date, exchange, optionID, exchangeInternalID
  - **Quote Key:** date, exchange, optionID, exchangeInternalID
  - **Route Link Key:** date, optionID, routingParty, routedOrderID, session, exchange
- Complex Options Orders:
  - **Complex Order Key:** date, exchange, [OptionID,] exchangeInternalID
  - **Route Link Key:** date, routingParty, routedOrderID, session, exchange

The Linkage Keys above have been grayed out as the Reject Message Event will not be linked to accepted data in the initial phase of rejection reporting. Linkage may be performed at a later time, pending the results of a data study and submission and approval of a corresponding Change Request.

4. Events for Stock Exchanges

Within this Technical Specification, events for stock exchanges, options exchanges, and the trade reporting facilities are documented in separate sections. This section describes reportable events for stock exchanges.

Table 15: Events for Stock Exchanges

Sec	Event	Message Type	Description
4.1	Order Accepted	EOA	An Exchange receives and accepts a routed order
4.2	Order Route	EOR	An Exchange routes an order through a routing broker dealer
4.3	Internal Order Route	EIR	An exchange routes an order to another internal subsystem
4.4	Order Modified	EOM	The material terms of an order have been changed
4.5	Order Adjusted	EOJ	A select set of material terms of an order have been changed
4.6	Order Canceled	EOC	An Exchange cancels an order in part or in whole
4.7	Order Trade	EOT	All trades are reported to CAT as two-sided transactions with a single event
4.8	Order Fill	EOF	When a routed order executes, the Exchange reports the fill with the order and the routing firm
4.9	Order Cancel Route	ECR	An exchange initiates a cancel request on an order that it previously routed away.
4.10	Order Modify Route	EMR	An exchange initiates a modify or cancel/replace request on an order it previously routed away
4.11	Order Restatement	EORS	An order that persists across multiple business days is restated each day before any other activity is reported for that symbol
4.12	Trade Break	ETB	A trade is broken
4.13	Trade Correction	ETC	A trade is corrected

4.1. Order Accepted Event

When an exchange receives and accepts a routed order, an Order Accepted event is reported to CAT. If the order is rejected (i.e., not received and successfully processed by the matching engine), then an event is not reported to CAT.

Some systems will outright reject messages if they are malformed or contain a duplicate order ID. Other systems will silently ignore certain malformed messages (e.g., the OUCH protocol specifically states that new orders containing duplicate order tokens are silently ignored). However, all current systems will send some sort of positive acknowledgement when an order has been finally accepted into the system. Some systems will send an acknowledgement from the gateway upon receipt of the request, but the order could still possibly be rejected instead of accepted by the matching engine. Such protocols have a prescribed way of notifying the sender whether or not their order was actually accepted.

The basic rule is that orders rejected by the gateway are not reportable, but any order reaching the matching engine is reportable.

Note that for the order accepted event, the firm that sends the order to the exchange will be referred to as the routing firm. In the next event, order route event (section 4.2), the routing broker dealer will also be referred to as the routing firm.

The Order ID that is used in orders must be globally unique when combined with the date, exchange, symbol and general side, where the general side is either Buy or Sell.

Table 16: Order Accepted

Equity Order Accepted (EOA)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EOA	R
2	exchange	Exchange ID	The ID for the exchange which has accepted this order	R
3	eventTimestamp	Timestamp	The date/time of order receipt	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C

Equity Order Accepted (EOA)				
#	Field Name	Data Type	Description	Include Key
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	routingParty	Text (8)	The ID string used to identify the entity that routed this order to the exchange	R
9	routedOrderID	Text (40)	The order ID that the firm used in the API message when they sent the order to the exchange (e.g., in FIX it would be ClOrdId, in OUCH it would be Order Token)	R
10	session	Text (40)	The ID assigned to the specific session that the routing member used to route the order to the exchange	R
11	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values	R
12	price	Price	The limit price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
13	quantity	Unsigned	The order quantity	R
14	displayQty	Unsigned	The displayed quantity for this order	R
15	displayPrice	Price	The displayed price for this order. This must be provided when displayQty is greater than zero.	C
16	workingPrice	Price	The working price of the order at the time it was accepted. Note that Modified events must be reported to CAT anytime the working price changes.	C
17	orderType	Choice	The type of order being submitted (e.g., market, limit). See the corresponding entry in the Data Dictionary for more details about order types	R
18	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
19	capacity	Choice	See entry for "capacity" in the Data Dictionary for acceptable values	R
20	handlingInstructions	Name/Value Pairs	Defines the handling instructions, as described in Data Dictionary for Handling Instructions	C
21	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order, that are not necessarily handling instructions	C

Equity Order Accepted (EOA)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
22	member	Member Alias	The identifier for the member firm that is responsible for the order	R
23	nbbPrice	Price	The NBBO at the moment the order was accepted. Prices are required. Quantities are optional	R
24	nbbQty	Unsigned		O
25	nboPrice	Price		R
26	nboQty	Unsigned		O

Linkage Keys for **EOA**:

- **Order Key:** date, exchange, symbol, orderID
- **Route Link Key:** date, symbol, routingParty, routedOrderID, session, exchange
- **Cross Order Key:** date, exchange, orderID, pairedOrderId (if populated in order attributes name value pair)

4.2. Order Route Event

The following Order Route event is used to report when an exchange routes an order through a routing broker dealer.

When an order is routed, some exchanges create a derived order (with a different order ID), to represent the order being routed away. Others just route the order (or part of the order) straight to the routing broker without changing the Order ID. In either case, CAT must be able to link the internal order on the exchange with the internal order at the routing BD. Thus, both the report from the exchange and the report from the routing BD must have the same identifiers for the routed order. This is very similar to the process described earlier related to the Accepted event.

Note that for an order route event, the routing broker is referred to as the routing firm.

The Order Route event reported by the exchange needs three key pieces of information: the Routing Firm receiving the routed order, the Session ID through which the order is being routed, and the Routed Order ID, which is the order ID sent to the routing firm.

The Routing Firm must be represented by an entry in the exchange's member dictionary (though not necessarily a member of the exchange). Furthermore, as explained in the linkage section, both the exchange and the Routing Firm must know which Member Alias is to be reported to CAT because both will have to report the same Member Alias (the exchange in their Route event, and the firm in their

Accepted event). Either both sides must use a constant value, or there must be some way to derive the value being used (via session configurations or in the message itself).

If the exchange creates a derived order, and passes that order ID to the firm via its API, then the Routed Order ID will be the order ID of the derived order. If, however, there is no derived order and the exchange passes its own internal order ID to the routing broker, then the internal order ID will also be assigned as the Routed Order ID. In this case, both the order ID and the routed order ID are populated with the same value.

Table 17: Order Route

Equity Order Route (EOR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EOR	R
2	exchange	Exchange ID	The ID for the exchange which is routing this order	R
3	eventTimestamp	Timestamp	The date/time at which the order was routed	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	routingParty	Text (8)	The ID string used to identify the entity receiving this routed order. This value must match the value reported by the routing broker in their Order Accepted report	R
9	routedOrderID	Text (40)	The ID assigned to this order by the exchange when submitting the order to the routing firm. This value must match the value reported by the routing broker in their Order Accepted report	R
10	session	Text (40)	The ID assigned to the specific session used when sending the order from the exchange to the routing firm.	R
11	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values	R

Equity Order Route (EOR)				
#	Field Name	Data Type	Description	Include Key
12	price	Price	The limit price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
13	quantity	Unsigned	The order quantity	R
14	displayQty	Unsigned	The displayed quantity for this order	R
15	orderType	Choice	The type of order being submitted (e.g., market, limit). See the corresponding entry in the Data Dictionary for more details about order types	R
16	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
17	capacity	Choice	See entry for "capacity" in the Data Dictionary for acceptable values	R
18	handlingInstructions	Name/Value Pairs	Defines the handling instructions, as described in Data Dictionary for Handling Instructions	C
19	result	Choice	The result of the route request (e.g. acknowledged, rejected, or no response). See the Data Dictionary for the list of allowed values	O
20	resultTimestamp	Timestamp	The date/time the result of the request was received, required if the result is ACK (acknowledged) or REJ (rejected)	O
21	member	Member Alias	The identifier for the member firm that is responsible for the order	R
22	nbbPrice	Price	The NBBO at the moment the order was routed. Prices are required. Quantities are optional	R
23	nbbQty	Unsigned		O
24	nboPrice	Price		R
25	nboQty	Unsigned		O

Linkage Keys for **EOR**:

- **Order Key:** date, exchange, symbol, orderID
- **Route Link Key:** date, symbol, exchange, routedOrderID, routingParty

4.3. Internal Order Route Event

In some cases, an exchange may have multiple internal subsystems involved in handling orders. In such cases, and order may be accepted by one internal system, and then routed to one or more internal systems for processing. Routes within an exchange are not required to be reported to CAT. However, there are cases where it is difficult for an exchange to report the entire status of an order to CAT when its internal processing is handled on multiple systems. Specifically, ensuring that the events contain the same order identifiers would require substantial post processing.

Thus, an internal route event may be reported to CAT, indicating that an order is being passed from one internal system to another. This will allow CAT to link events that are related to the same order within an exchange, even if the exchange has changed the identifiers on the order as it moves between internal systems.

Table 18: Internal Order Route

Equity Internal Order Route (EIR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EIR	R
2	exchange	Exchange ID	The ID for the exchange which is routing this order.	R
3	eventTimestamp	Timestamp	The date/time at which the order was routed.	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps.	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	routingParty	Text (8)	The ID string used to identify the internal subsystem that is receiving this routed order. This value must match the value reported by the receiving subsystem in the `routingParty` field of their Order Accepted report	R
9	routedOrderID	Text (40)	The ID assigned to this order by the exchange when submitting the order to the subsystem. This value must match the value reported by the receiving subsystem in the `routedOrderID` field of their Order Accepted report	R
10	session	Text (40)	The ID assigned to the specific session used when sending the order from the sending subsystem to the receiving subsystem. This value must match the value reported by the receiving subsystem in the `session` field of their Order Accepted report	R

Equity Internal Order Route (EIR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
11	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values	R
12	price	Price	The limit price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
13	quantity	Unsigned	The order quantity	R
14	displayQty	Unsigned	The displayed quantity for this order	R
15	orderType	Choice	The type of order being submitted (e.g., market, limit). See the corresponding entry in the Data Dictionary for more details about order types	R
16	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
17	capacity	Choice	See entry for "capacity" in the Data Dictionary for acceptable values	R
18	handlingInstructions	Name/Value Pairs	Defines the handling instructions, as described in Data Dictionary for Handling Instructions	C
19	result	Choice	The result of the route request (e.g. acknowledged, rejected, or no response). See the Data Dictionary for the list of allowed values	O
20	resultTimestamp	Timestamp	The date/time the result of the request was received, required if the result is ACK (acknowledged) or REJ (rejected)	O
21	member	Member Alias	The identifier for the member firm that is responsible for the order	R

Linkage Keys for **EIR**:

- **Order Key:** date, exchange, symbol, orderID
- **Route Link Key:** date, symbol, exchange, routedOrderID, session, routingParty

4.4. Order Modified Event

An event must be sent to CAT to report any customer modification to the order. Additionally, an event must be sent to CAT to report any changes to the order due to an exchange action, including updates related to changes in market conditions.

Events that should be reported include, but are not limited to:

- Any customer update that passes validation and is successfully processed by the trading system
- Changes to the available quantity of the order, such as liquidity returning from an away market unexecuted
- Changes to the working price, display price, or display quantity
- Changes to the executability of an order, such as when a regular-hours order arrived prior to the opening time and it is now the opening time or when an order expires and no explicit cancellation is provided

This event supports all possible modifications to an equity order. The full state of the order should be reported, including fields that did not change as a result of the modification.

Table 19: Order Modified

Equity Order Modified (EOM)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EOM	R
2	exchange	Exchange ID	The identifier for the exchange which has modified this order	R
3	eventTimestamp	Timestamp	The date/time at which the modification was received or originated	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	originalOrderID	Text (40)	The internal order ID before the modify / replacement created a new order ID. If the order kept its ID through the modification, then this value need not be included	C
9	initiator	Choice	Indicates who initiated the order modification: See entry for "initiator" in the Data Dictionary for acceptable values	R
10	nbbPrice	Price	The NBBO at the moment the order was modified. Prices are required. Quantities are optional	R
11	nbbQty	Unsigned		O

Equity Order Modified (EOM)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
12	nboPrice	Price		R
13	nboQty	Unsigned		O
14	price	Price	The limit price of the order, if applicable. This must be provided when orderType indicates a limit order. Note that this is only for reporting limit price modifications. Automated changes to prices (e.g., PEG orders) would be tracked by reporting a difference in the working price. See the PEG example in section 7.5 for exact details	C
15	quantity	Unsigned	When the initiator field is set to Firm or Market Maker, the order quantity. When the initiator field is set to Exchange, the total quantity available on the local book at the conclusion of the modification.	R
16	displayQty	Unsigned	The displayed quantity for this order	R
17	displayPrice	Price	The displayed price for this order. This must be provided when displayQty is greater than zero.	C
18	workingPrice	Price	The working price of the order	C
19	leavesQty	Unsigned	The quantity left open after the modification has occurred.	R
20	orderType	Choice	The type of order being submitted (e.g., market, limit). See the corresponding entry in the Data Dictionary for more details about order types.	R
21	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
22	capacity	Choice	See entry for Capacity in the Data Dictionary for acceptable values	R
23	handlingInstructions	Name/Value Pairs	Defines the handling instructions, as described in Data Dictionary for Handling Instructions.	C
24	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order, that are not necessarily handling instructions.	C
25	member	Member Alias	The identifier for the member firm that is responsible for the order	R

Equity Order Modified (EOM)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
26	routedOrderID	Text (40)	For customer-driven changes to the order, the ID assigned to this order by the routing firm when submitting the modification to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID used to route the order away. Except as noted above, not required for exchange-driven modifications. This must be provided when initiator is ‘Firm’ or ‘MarketMaker’.	C
27	routingParty	Text(8)	For customer-driven changes to the order, the ID string used to identify the entity that routed this order modification to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID string used to route the order away. Should match the value of the EOR event routingParty with the same routedOrderID. Except as noted above, not required for exchange-driven modifications. This must be provided when initiator is ‘Firm’ or ‘MarketMaker’.	C
28	session	Text(40)	For customer-driven changes to the order, the ID assigned to the specific session that the routing member used to route the order to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID used to route the order away. Should match the value of the EOR event session with the same routedOrderID. Except as noted above, not required for exchange-driven modifications. This must be provided when initiator is ‘Firm’ or ‘MarketMaker’.	C
29	side	Choice	The side of the order: See entry for “side” in the Data Dictionary for acceptable values. Should be provided for firm or market maker updates to an order. Should be reported even if it has not changed from the prior version of the order. This must be provided when initiator is ‘Firm’ or ‘MarketMaker’.	C

Linkage Keys for **EOM**:

- **Order Key:** date, exchange, symbol, orderID

- **Previous Order Key:** date, exchange, symbol, originalOrderID
- **Route Link Key:** date, symbol, exchange, routedOrderID, routingParty, session
- **Cross Order Key:** date, exchange, orderID, pairedOrderId (if pairedOrderId is populated in order attributes name value pair)

4.5. Order Adjusted Event

An event must be sent to CAT to report any customer modification to the order. Additionally, an event must be sent to CAT to report any changes to the order due to an exchange action, including updates related to changes in market conditions.

Unlike the EOM, which supports changes to any reportable attribute, the EOJ event supports only changes to the side, price, quantity, working price, display price, and display quantity. Side adjustments are only allowed for same-side changes (e.g., changes between short and long sell).

All other order instructions are assumed to be unchanged.

Table 20: Order Adjusted

Equity Order Adjusted (EOJ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EOJ	R
2	exchange	Exchange ID	The identifier for the exchange which has modified this order	R
3	eventTimestamp	Timestamp	The date/time at which the modification was received or originated	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	originalOrderID	Text (40)	The internal order ID before the modify / replacement created a new order ID. If the order kept its ID through the modification, then this value need not be included	C

Equity Order Adjusted (EOJ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
9	initiator	Choice	Indicates who initiated the order modification: See entry for "initiator" in the Data Dictionary for acceptable values	R
10	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values. Should be provided for firm or market maker updates to an order. Should be provided for all firm updates to the order even if it hasn’t changed from the previous version of the order. This must be provided when initiator is ‘Firm’ or ‘MarketMaker’.	C
11	price	Price	The limit price of the order, if it changed. This must be provided when orderType indicates a limit order.	C
12	displayPrice	Price	The displayed price for this order. This must be provided when displayQty is greater than zero.	C
13	workingPrice	Price	The working price of the order	C
14	quantity	Unsigned	When the initiator field is set to Firm or Market Maker, the order quantity. When the initiator field is set to Exchange, the total quantity available on the local book at the conclusion of the modification. This must be provided when initiator is ‘Firm’ or ‘MarketMaker’.	C
15	displayQty	Unsigned	The displayed quantity for this order. This must be provided when displayPrice is provided.	C
16	leavesQty	Unsigned	The quantity left open after the modification has occurred.	C
17	member	Member Alias	The identifier for the member firm that is responsible for the order	R
18	nbbPrice	Price	The NBBO at the moment the order was modified. Prices are required. Quantities are optional.	R
19	nbbQty	Unsigned		O
20	nboPrice	Price		R
21	nboQty	Unsigned		O

Equity Order Adjusted (EOJ)				
#	Field Name	Data Type	Description	Include Key
22	routedOrderID	Text (40)	For customer-driven changes to the order, the ID assigned to this order by the routing firm when submitting the modification to the exchange For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID used to route the order away Except as noted above, not required for exchange-driven modifications This must be provided when initiator is 'Firm' or 'MarketMaker'.	C
23	routingParty	Text(8)	For customer-driven changes to the order, the ID string used to identify the entity that routed this order modification to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID string used to route the order away. Should match the value of the EOR event routingParty with the same routedOrderID. Except as noted above, not required for exchange-driven modifications. This must be provided when initiator is 'Firm' or 'MarketMaker'.	C
24	session	Text(40)	For customer-driven changes to the order, the ID assigned to the specific session that the routing member used to route the order to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID used to route the order away. Should match the value of the EOR event session with the same routedOrderID. Except as noted above, not required for exchange-driven modifications. This must be provided when initiator is 'Firm' or 'MarketMaker'.	C

Linkage Keys for **EOJ**:

- **Order Key:** date, exchange, symbol, orderID
- **Previous Order Key:** date, exchange, symbol, originalOrderID
- **Route Link Key:** date, symbol, exchange, routedOrderID, routingParty, session
- **Cross Order Key:** date, exchange, orderID, pairedOrderId (if pairedOrderId is populated in order attributes name value pair)

4.6. Order Canceled Event

When an exchange cancels an order, in part or in whole, the event must be reported to CAT. Note that an explicit Canceled Event is required for every order that is canceled, even orders that have implicit "execute or cancel" instructions like IOC orders.

A Canceled event should be used anytime any part of an order is canceled. For example, an order can be partially reduced either with a cancel message or a modify (cancel/replace) message. If an actual cancel is processed by the exchange, a Canceled event would be reported. If a modify and/or cancel/replace was sent to the exchange, a Modified event would be reported. This keeps the reported event in line with the original intent.

Some protocols only allow full cancels; partial cancels must be accomplished via a cancel/replace. In such cases, partial cancels would always be reported as Modified events.

Table 21: Order Canceled

Equity Order Canceled (EOC)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EOC	R
2	exchange	Exchange ID	The ID for the exchange which has canceled this order.	R
3	eventTimestamp	Timestamp	The date/time at which the cancellation was received or originated.	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier.	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	cancelQty	Unsigned	The quantity being canceled.	R
9	leavesQty	Unsigned	The quantity left open after the cancel event (zero for a full cancel)	R
10	initiator	Choice	Indicates who initiated the order cancellation: See entry for "initiator" in the Data Dictionary for acceptable values	R

Equity Order Canceled (EOC)				
#	Field Name	Data Type	Description	Include Key
11	cancelReason	Choice	Code representing the reason why the order was canceled. The actual value of the code is exchange specific. See Data Dictionary for the list of allowed values	O
12	member	Member Alias	The identifier for the member firm that is responsible for the order	R

Linkage Keys for **EOC**:

- **Order Key:** date, exchange, symbol, orderID

4.7. Order Trade Event

All trade events are reported to CAT as two-sided transactions, with a single event.

Each order trade event is represented with the following details. The details in the table Order Trade Side Details must be populated for each side of the trade.

Table 22: Order Trade Events

Equity Order Trade (EOT)				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	EOT	R
2	exchange	Exchange ID	The ID for the exchange on which the trade took place	R
3	eventTimestamp	Timestamp	The date/time of execution	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	tradeID	Text (40)	This ID will be used when a specific trade needs to be identified, for example in trade break and correction reports. The combination of date, exchange, symbol, and tradeID must be globally unique	R
8	quantity	Unsigned	Quantity of the trade	R

Equity Order Trade (EOT)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
9	price	Price	Price of the trade	R
10	saleCondition	Text (8)	Conditions under which trade was executed	C
11	executionCodes	Name/Value Pairs	Describes any execution codes, acceptable values are described in Data Dictionary. These codes apply to both sides of the trade	C
12	buyDetails	Order Trade Side Details	See Order Trade Side Details table	R
13	sellDetails	Order Trade Side Details	See Order Trade Side Details table	R
14	nbbPrice	Price	The national best bid price at the moment the trade occurred	R
15	nbbQty	Unsigned	The national best bid quantity at the moment the trade occurred	O
16	nboPrice	Price	The national best offer price at the moment the trade occurred	R
17	nboQty	Unsigned	The national best offer quantity at the moment the trade occurred	O

Table 23: Order Trade Side Details

Equity Order Trade: Side Details				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
12.*n*.1 / 13.*n*.1	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values. Not required if there is not order for the side as indicated by the NOBUYID/NOSELLID instruction. This must be provided if orderID is provided.	C
12.*n*.2 / 13.*n*.2	leavesQty	Unsigned	The quantity remaining unfilled after this trade event. Not required if there is no order for the side as indicated by the NOBUYID/NOSELLID instruction. Not required when used in a trade correction	C

Equity Order Trade: Side Details				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
12.*n*.3 / 13.*n*.3	orderID	Text (40)	The internal order ID for this side of the trade. This must be provided when, and only when, there is not a NOBUYID/NOSELLID instruction. This must be blank if the NOBUYID/NOSELLID instruction exists.	C
12.*n*.4 / 13.*n*.4	capacity	Choice	See entry for Capacity in the Data Dictionary for acceptable values. Not required if there is no order for the side as indicated by the NOBUYID/NOSELLID instruction. This must be provided if orderID is provided.	C
12.*n*.5/ 13.*n*.5	clearingNumber	Text (20)	DTCC clearing number for this side of the trade. Not required if there is no order for the side as indicated by the NOBUYID/NOSELLID instruction. This must be provided if orderID is provided.	C
12.*n*.6 / 13.*n*.6	executionCodes	Name/Value Pairs	Describes any execution codes, as described in Data Dictionary for Execution Codes. These codes would only apply only to this side of the trade	C
12.*n*.7 / 13.*n*.7	liquidityCode	Choice	Specifies if this side of the trade was adding or removing liquidity. See entry for liquidityCode in the Data Dictionary for permitted values. Not required if there is no order for the side as indicated by the NOBUYID/NOSELLID instruction.	C
12.*n*.8 / 13.*n*.8	member	Member Alias	The identifier for the member firm that is responsible for the order on this side of the trade. Not required if there is no order for the side as indicated by the NOBUYID/NOSELLID instruction. This must be provided if orderID is provided.	C
12.*n*.9 / 13.*n*.9	routedOrderID	Text (40)	For events representing an away trade, the exchange-assigned ID used to route the order away.	O

Linkage Keys for **EOT**:

- **Order Key:** date, exchange, symbol, buyDetails.orderID
- **Order Key:** date, exchange, symbol, sellDetails.orderID
- **Trade Key:** date, exchange, symbol, tradeID
- **Route Link Key:** date, symbol, exchange, buyDetails.routedOrderID
- **Route Link Key:** date, symbol, exchange, sellDetails.routedOrderID

4.8. Order Fill Event

When a routed order executes, the routing firm acquires the position. The exchange will report the fill with the order on one side, and the routing firm on the other side.

Table 24: Order Fill Event

Equity Order Fill (EOF)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EOF	R
2	exchange	Exchange ID	The ID of the exchange reporting the fill to CAT	R
3	eventTimestamp	Timestamp	The date/time when the fill was processed by the exchange	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	fillID	Text (40)	A unique identifier for the transaction. The combination of reporter, date, symbol, side, and fillID should be unique	R
7	symbol	Symbol	The symbol of the stock being filled	R
8	quantity	Unsigned	Quantity of the fill	R
9	price	Price	Price of the fill	R
10	leavesQty	Unsigned	The quantity remaining unfilled after this fill event	R
11	saleCondition	Text (8)	Conditions under which trade was executed	C
12	orderID	Text (40)	The internal ID of the order	R
13	side	Choice	Side of the executed trade: for example Buy, Sell or Short. See the entry 'side' in data dictionary for the list of accepted values	R
14	clearingNumber	Text (20)	DTCC clearing number for this side of the trade	R
15	contraClearingNumber	Text (20)	DTCC clearing number for contra side of the trade	O
16	executionCodes	Name / Value Pairs	Optional. Can include zero or more execution codes, as described in Data Dictionary for Execution Codes. These codes would only apply only to this side of the trade	C

Equity Order Fill (EOF)				
#	Field Name	Data Type	Description	Include Key
17	routingParty	Text (8)	The ID string used to identify the entity that received this routed order. This value will be the same as in the Order Route event for the order being filled	R
18	routedOrderID	Text (40)	The same Order ID that was used when the order was routed away - and will be on the execution report from the routing BD	R
19	session	Text (40)	The Session ID of the session on which the order was routed to the BD, and will be the same session on which the execution came back from the BD	R
20	capacity	Choice	See entry for Capacity in the Data Dictionary for acceptable values	R
21	member	Member Alias	The identifier for the member firm that is responsible for the order being filled	R

Linkage Keys for **EOF**:

- **Order Key:** date, exchange, symbol, orderID
- **Fill Key:** date, exchange, symbol, fillID
- **Route Link Key:** date, symbol, exchange, routedOrderID, session, routingParty

4.9. Order Cancel Route Event

When an exchange initiates a cancel request on an order it has previously routed away, it must report its intent to cancel, using a Cancel Route Event.

Table 25: Order Cancel Route

Equity Order Cancel Route (ECR)				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	ECR	R
2	exchange	Exchange ID	The ID for the exchange canceling the routed order	R
3	eventTimestamp	Timestamp	The date/time when the cancel request was sent to the routing firm	R

Equity Order Cancel Route (ECR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	routingParty	Text (8)	The ID string used to identify the entity that received this routed order. This value will be the same as in the Order Route event for the order being canceled	R
9	routedOrderID	Text (40)	The routed ID for the order being canceled - must also match the routedOrderID in the original Order Route message for this order	R
10	session	Text (40)	The session ID on which the cancel request is being made - must also match the session in the original Order Route message for this order	R
11	desiredLeavesQty	Unsigned	The desired number of shares remaining in the order after the cancel request has been issued. A value of zero indicates a full cancel	R
12	result	Choice	The result of the cancel request (e.g. acknowledged, rejected, or no response). See the Data Dictionary for the list of allowed values	O
13	resultTimestamp	Timestamp	The date/time the result of cancel request was received, required if the result is ACK (acknowledged) or REJ (rejected)	O
14	member	Member Alias	The identifier for the member firm that is responsible for the order	R

Linkage Keys for **ECR**:

- **Order Key:** date, exchange, symbol, orderID
- **Route Link Key:** date, symbol, exchange, routedOrderID, routingParty

4.10. Order Modify Route Event

When an exchange initiates a modify or cancel/replace request on an order it has previously routed away, it must report its intent to modify the order, using a Modify Route Event.

If the request does not change the routed order ID, then both routedOrderID and routedOriginalOrderID must be the same.

Table 26: Order Modify Route

Equity Order Modify Route (EMR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EMR	R
2	exchange	Exchange ID	The ID for the exchange modifying the routed order	R
3	eventTimestamp	Timestamp	The date/time when the exchange made the modify request	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	routingParty	Text (8)	The ID string used to identify the entity that received this routed order. This value will be the same as in the Order Route event for the order being modified	R
9	routedOrderID	Text (40)	The new routed ID for the order, which will be used to refer to the routed order after the modification (in FIX, ClOrdID - in OUCH, Replacement Order Token)	R
10	routedOriginalOrderID	Text (40)	The ID for the order being modified, as sent to the routing broker in the original route message, or the most recent modify message (in FIX OrigClOrdID, in OUCH Existing Order Token)	R
11	session	Text (40)	The ID assigned to the session used to send the modify request from the routing broker to the exchange - must also match the session in the original Order Route message for this order	R
12	price	Price	The limit price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
13	quantity	Unsigned	The order quantity	R
14	displayQty	Unsigned	The displayed quantity for this order	R

Equity Order Modify Route (EMR)				
#	Field Name	Data Type	Description	Include Key
15	orderType	Choice	The type of order being submitted (e.g., market, limit). See the corresponding entry in the Data Dictionary for more details about order types.	R
16	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values.	R
17	capacity	Choice	See entry for Capacity in the Data Dictionary for the full list of acceptable values	R
18	handlingInstructions	Name/Value Pairs	Can include zero or more handling instructions, as described in Data Dictionary for Handling Instructions	C
19	result	Choice	The result of the modify request (e.g. acknowledged, rejected, or no response). See the Data Dictionary for the list of allowed values	O
20	resultTimestamp	Timestamp	The date/time the result of modify request was received, required if the result is ACK (acknowledged) or REJ (rejected)	O
21	member	Member Alias	The identifier for the member firm that is responsible for the order	R
22	nbbPrice	Price	The national best bid price at the moment the trade occurred	R
23	nbbQty	Unsigned	The national best bid quantity at the moment the trade occurred	O
24	nboPrice	Price	The national best offer price at the moment the trade occurred	R
25	nboQty	Unsigned	The national best offer quantity at the moment the trade occurred	O

Linkage Keys for **EMR**:

- **Order Key:** date, exchange, symbol, orderID
- **Route Link Key:** date, symbol, exchange, routedOrderID, routingParty
- **Previous Route Link Key:** date, symbol, exchange, routedOriginalOrderID, routingParty

4.11. Order Restatement Event

Orders that persist across business days (e.g., GTC orders) must be restated each day before any other activity is reported for that symbol. The restatement is an explicit confirmation that the order is still active in the reporter's order book, and also provides an opportunity to use per-day unique order IDs for all orders.

The attributes of the order will be restated in terms of the order's current state, after any corporate actions have been processed (e.g., if a 2:1 split occurred, the quantity and price would reflect the resulting change).

Table 27: Order Restatement

Equity Order Restatement (EORS)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EORS	R
2	exchange	Exchange ID	The ID for the exchange which is restating this order	R
3	eventTimestamp	Timestamp	The date/time when the order was restated by the exchange	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	originalOrderDate	Date	The most recent trading day for which the order was active. Note that this may not be the date when the order was originally accepted. If the order has been active for multiple trading days, this field must reference the previous trading day when the order was active	R
9	originalOrderID	Text (40)	The most recent internal order ID that was assigned to the order before this restatement event. If the order ID has not changed, then orderID and originalOrderID must be equivalent. Note this requirement is different from modification events	R
10	side	Choice	The side of the order (e.g., Buy, Sell, Short, etc.). See entry for "side" in the Data Dictionary for acceptable values	R
11	price	Price	The limit price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
12	quantity	Unsigned	The order quantity, as adjusted for a corporate action, if applicable	R

Equity Order Restatement (EORS)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
13	displayQty	Unsigned	The displayed quantity for this order	R
14	displayPrice	Price	The displayed price for this order. This must be provided when displayQty is greater than zero.	C
15	workingPrice	Price	The working price of the order	C
16	leavesQty	Unsigned	The quantity of the order that remains open	R
17	orderType	Choice	The type of order being submitted (e.g., market, limit). See the corresponding entry in the Data Dictionary for more details about order types	R
18	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values.	R
19	capacity	Choice	See entry for Capacity in the Data Dictionary for acceptable values	R
20	handlingInstructions	Name/Value Pairs	Defines the handling instructions, as described in Data Dictionary for Handling Instructions	C
21	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order that are not necessarily handling instructions	C
22	member	Member Alias	The identifier for the member firm that is responsible for the order	R

Linkage Keys for **EORS**:

- **Order Key:** date, exchange, symbol, orderID
- **Previous Order Key:** originalOrderDate, exchange, symbol, originalOrderID

4.12. Trade Break Event

When a trade is broken, an event is reported to CAT with the appropriate information. Note that CAT adds the event to the history of the order. The broken trade is not removed from the history, as it is something that actually happened and should be recorded.

Table 28: Order Trade Break

Equity Order Trade Break (ETB)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	ETB	R
2	exchange	Exchange ID	The ID for the exchange on which the trade took place	R
3	eventTimestamp	Timestamp	The date/time of the break event	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, as reported on the original trade that is being broken	R
7	tradeDate	Date	The date on which the trade being broken occurred	R
8	tradeID	Text (40)	The ID for the trade that is being broken. This must match a previously reported trade	R
9	quantity	Unsigned	If the full quantity is being broken, then this field can be omitted. Otherwise, this represents the quantity of the original trade that is being broken	O
10	reason	Text (255)	Free format text field, with the reason for the break	O

Linkage Keys for **ETB**:

- **Trade Key:** tradeDate, exchange, symbol, tradeID

4.13. Trade Correction Event

If a trade is corrected in any way, a correction event must be reported to CAT with all details of the trade, after having been corrected.

As with trade breaks, CAT will still keep the original trade, adding the correction to the audit trail of the trade being corrected.

Table 29: Order Trade Correction

Equity Order Trade Correction (ETC)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	ETC	R
2	exchange	Exchange ID	The ID for the exchange on which the trade took place.	R
3	eventTimestamp	Timestamp	The date/time of correction	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	tradeID	Text (40)	This ID for the trade being corrected	R
8	refTradeID	Text (40)	The trade being referenced. Used to link corrections if trade corrections can assign new identifiers to trades. If included, refTradeID must reference a previously reported trade, or a previously reported trade correction that has a matching tradeID	C
9	quantity	Unsigned	Quantity of the trade.	R
10	price	Price	Price of the trade	R
11	saleCondition	Text (8)	Conditions under which trade was executed	C
12	executionCodes	Name/Value Pairs	Describes any execution codes, acceptable values are described in Data Dictionary. These codes apply to both sides of the trade	C
13	executionTimestamp	Timestamp	The date/time of the execution, applicable only when the execution time was corrected	O
14	buyDetails	Order Trade Side Details	See Order Trade Side Details table 26	O
15	sellDetails	Order Trade Side Details	See Order Trade Side Details table 26	O
16	reason	Text (255)	Free format text field, with the reason for the correction	O

Linkage Keys for **ETC**:

- **Order Key:** date, exchange, symbol, buyDetails.orderID
- **Order Key:** date, exchange, symbol, sellDetails.orderID

- **Route Link Key:** date, symbol, exchange, buyDetails.routedOrderID
- **Route Link Key:** date, symbol, exchange, sellDetails.routedOrderID
- **Trade Key:** date, exchange, symbol, tradeID
- **Previous Trade Key:** date, exchange, symbol, refTradeID

4.14. Lifecycle Keys

The lifecycle keys for each event are summarized in the following table.

Table 30: Equity Event Lifecycle Keys

Section	Event	Lifecycle Keys
4.1	Order Accepted	**Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, symbol, routingParty, routedOrderID, session, exchange **Cross Order Key**: date, exchange, orderID, pairedOrderId (if pairedOrderId is populated in order attributes name value pair)
4.2	Order Route	**Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, symbol, exchange, routedOrderID, routingParty
4.3	Internal Order Route	**Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, symbol, exchange, routedOrderID, session, routingParty
4.4	Order Modified	**Order Key**: date, exchange, symbol, orderID **Previous Order Key**: date, exchange, symbol, originalOrderID **Route Link Key:** date, symbol, exchange, routedOrderID, session, routingParty **Cross Order Key**: date, exchange, orderID, pairedOrderId (if pairedOrderId is populated in order attributes name value pair)
4.5	Order Adjusted	**Order Key**: date, exchange, symbol, orderID **Previous Order Key**: date, exchange, symbol, originalOrderID **Route Link Key:** date, symbol, exchange, routedOrderID, session, routingParty **Cross Order Key**: date, exchange, orderID, pairedOrderId (if pairedOrderId is populated in order attributes name value pair)
4.6	Order Canceled	**Order Key**: date, exchange, symbol, orderID

Section	Event	Lifecycle Keys
4.7	Order Trade	**Order Key**: date, exchange, symbol, buyDetails.orderID **Order Key**: date, exchange, symbol, sellDetails.orderID **Trade Key**: date, exchange, symbol, tradeID **Route Link Key:** date, symbol, exchange, buyDetails.orderID **Route Link Key:** date, symbol, exchange, sellDetails.orderID
4.8	Order Fill	**Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, symbol, exchange, routedOrderID, session, routingParty **Fill Key**: date, exchange, symbol, fillID
4.9	Order Cancel Route	**Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, symbol, exchange, routedOrderID, routingParty
4.10	Order Modify Route	**Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, symbol, exchange, routedOrderID, routingParty **Previous Route Link Key**: date, symbol, exchange, routedOriginalOrderID,, routingParty
4.11	Order Restatement	**Order Key**: date, exchange, symbol, orderID **Previous Order Key**: originalOrderDate, exchange, symbol, originalOrderID
4.12	Trade Break	**Trade Key**: tradeDate, exchange, symbol, tradeID
4.13	Trade Correction	**Order Key**: date, exchange, symbol, buyDetails.orderID **Order Key**: date, exchange, symbol, sellDetails.orderID **Route Link Key:** date, symbol, exchange, buyDetails.routedOrderID **Route Link Key:** date, symbol, exchange, sellDetails.routedOrderID **Trade Key**: date, exchange, symbol, tradeID

5. Events for Options Exchanges

These events are specific for options exchanges.

Table 31: Events for Options Exchanges

Section	Event	Message Type	Description
5.1.1	Quote	OQ	A new quote or a quote replacement
5.1.2	Quote Cancel	OQC	Report when a quote is canceled
5.2.1.1	Simple Option Order Accepted	OOA	Represents either a stand-alone option series order, or one leg of a complex parent order accepted by an exchange
5.2.1.2	Complex Option Order Accepted	OCOA	Represents the complex option order accepted by an exchange
5.2.1.3	Stock Leg Order	OSL	Stock legs are reported individually, with a link to the parent complex order
5.2.2.1	Option Order Modified	OOM	Modification of a simple option order or an option leg order
5.2.2.2	Complex Option Order Modified	OCOM	Modification of a complex option order
5.2.2.3	Stock Leg Modified	OSLM	Modification of a stock leg of a complex option order
5.2.2.4	Option Order Adjusted	OOJ	Adjustment of a simple option order or an option leg order
5.2.2.5	Complex Option Order Adjusted	OCOJ	Adjustment of a complex option order
5.2.2.6	Stock Leg Adjusted	OSLJ	Adjustment of a stock leg of a complex option order
5.2.3	Option Order Canceled	OOC	Cancellation of a simple option order or a complex option order
5.2.4.2	Option Route	OOR	Routing all or part of a simple option order, routing two stock legs to be crossed, or routing a stock leg for execution
5.2.4.3	Complex Option Route	OCOR	Routing of a complex order to an external destination.
5.2.4.4	Internal Option Route	OIR	Internal route of an option or a leg of a complex option
5.2.4.5	Internal Complex Option Route	OCIR	Internal route of a complex option
5.2.4.6	Modify Option Route	OOMR	Modification or cancel/replace request on an option or stock leg order previously routed away,

Section	Event	Message Type	Description
5.2.4.7	Option Cancel Route	OOCR	Cancel request on an order that has been previously routed away
5.2.5.1	Simple Option Trade	OT	Two-sided trade report for simple options and option legs
5.2.5.2	Stock Leg Fill	OSLF	One-sided fill of a routed stock leg order
5.2.6	Post Trade Allocation	OPTA	In the event of a modified, canceled, or replaced post trade Allocation, the final allocation is reported to CAT.
5.3	Option Order Restatement	OORS	Restatement for options orders that persist across business days (e.g., GTC orders)
5.4	Option Trade Break	OTB	When a trade is broken
5.5	Option Trade Correction	OTC	When a trade is corrected in any way
5.6.1	Option Floor Participant	OFP	A floor participant routes a simple option order to a matching engine.
5.6.2	Complex Option Floor Participant	OCFP	A floor participant routes a complex option order to a matching engine.
5.6.3	Option Return to Floor Participant	ORFP	The matching engine returns an order to a Floor Participant.

5.1. Market Maker Quotes

Quotes issued by market makers (MMs) to options exchanges must be reported to CAT. This section will describe the types of attributes that are used to model quote events, and the types of quote events that should be reported to CAT. CAT supports both one-sided and two-sided quotes.

While some exchanges create quotes and orders the same way, CAT considers them distinct from a reporting perspective, and they must be reported distinctly. First, MMs are exempt from reporting their quotes to CAT (Section 6.4(d)(iii) of the CAT NMS Plan). Instead, the exchange is fully responsible for submitting the quotes they receive from MMs. Second, the MMs must inform the exchange of the time that they sent each quote, so the exchange can report it to CAT along with the quote. Third, quotes require fewer data elements than orders.

Each quote must have a unique Quote ID. Specifically, when a trade occurs with a MM quote on one side, the Quote ID in the trade will identify the exact quote. The combination of Exchange ID, Date, Option ID, and Quote ID should be globally unique.

Furthermore, each quote update must also have a unique Quote ID which is different from the Quote ID for the quote being updated. If the exchange only supports a single quote per MM, the event can be so noted, and the Quote ID for the quote that is being replaced is not necessary. Otherwise, the update must also include the Quote ID for the quote that is being updated/replaced by the new quote.

The exchange must guarantee uniqueness of quote IDs throughout the day.

There are two types of quote events in CAT:

- **Quote Event:** Used to report a new quote or a quote replacement. When a quote is replaced, the Original Quote ID will identify the quote being replaced, and the Quote ID will provide the new ID for the updated and replaced quote (or note in the event that the market maker can only have one quote active at any given time).
- **Quote Cancel:** Reported when a quote is canceled.

For block quotes, each quote in the block would be reported to CAT as a separate quote, with a separate unique Quote ID. In such a case, the quote Sent Timestamp would be the same for each quote from the same block because they were all sent at the same time by the MM. However, the combination of Event Timestamp and Event Sequence Number must be unique for each quote.

Similarly, when a bulk cancel is requested, a separate quote cancel event is required for each quote that is canceled by such a request.

On some exchanges, quotes are allowed to be sent before the trading system is ready to process them. For example, there may be an established protocol where the API documents that quotes sent before a particular time are ignored. Or, a protocol may send a "Now Accepting Quotes" message to market makers, and any quotes sent before that time are ignored. In such cases, those ignored quotes are not processed, so they should not be reported to CAT.

Note that all pre-open quotes are still reportable to CAT. This exception is explicitly for those cases where the exchange allows quotes to be sent before they are officially accepted - but those quotes are neither processed, nor entered into the book, nor accepted for participating in the opening nor any other trading session.

Once the system has started accepting quotes (either because a set time has arrived, or it has sent out a message indicating that quotes are now being accepted), then all quotes must be reported. CAT does not have rules in place for when exchanges start accepting quotes, but it seems that all exchanges start accepting quotes at least five minutes before the start of trading.

For example, in the following diagram, an exchange ignores quotes until they send their "Now Accepting Quotes" message. Thereafter all quotes are processed and reported to CAT.

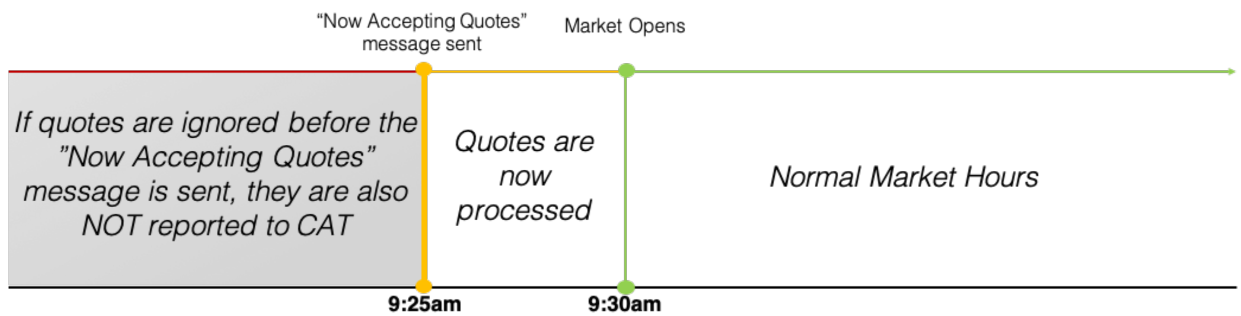

Figure 3: Accepted Quotes Processing

Similarly, if a quote is rejected and neither accepted nor booked, then the quote should not be reported to CAT.

5.1.1.Quote Event

The following data elements are to be reported with all quote events. For two-sided quotes, all bid/ask/price/qty values are required. For one-sided quotes, both the price and quantity fields are required, but only for one side.

Table 32: Quote Events

Option Quote (OQ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OQ	R
2	exchange	Exchange ID	The identifier for the exchange that received this quote	R
3	eventTimestamp	Timestamp	The date/time when the quote was received by the exchange	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	marketMaker	Member Alias	The Member Alias assigned by the SRO to identify the market maker issuing the quote. In the case where a market maker has multiple users (e.g., acronyms used to differentiate users within the	R

Option Quote (OQ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			same MM), there would be a separate Member Alias given to each user or sub-account	
7	sentTimestamp	Timestamp	The date/time when the market maker sent the quote to the exchange. Must be populated unless quote record was generated by an exchange system and was not received systematically from a Market Maker.	C
8	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R
9	quoteID	Text (40)	When onlyOneQuote=True, the unique identifier assigned to this quote by the exchange. If an askQuoteID value is also provided, then this value will only be applied to the bid side of the quote. When onlyOneQuote=False, the unique identifier assigned to the bid. To provide a unique identifier for an ask, use the askQuoteID field. When onlyOneQuote=False, this field must be populated when bidPrice is populated.	C
10	onlyOneQuote	Boolean	True if the system allows only one quote per OptionID for this market maker; false otherwise	R
11	originalQuoteID	Text (40)	When onlyOneQuote=False, this field must be populated when the bid from this record replaces a previously submitted bid. This field must not be provided for a bid that does not replace a previous bid, and it should never be populated for an ask. When onlyOneQuote=True no value is necessary for this field.	C
12	bidPrice	Price	The price being bid for the option (can be zero in two-sided quote which supports spread quotes in low prices names). When onlyOneQuote=False, this field must be populated when quoteID is populated. At least one of bidPrice and askPrice must be provided.	C
13	bidQty	Unsigned	The quantity being bid for the option (can be zero in two-sided quote which supports spread quotes in low prices names)	C
14	askPrice	Price	The price being asked for the option. When onlyOneQuote=False, this field must be populated when askQuoteID is populated. At least one of bidPrice and askPrice must be provided.	C

Option Quote (OQ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
15	askQty	Unsigned	The quantity being asked for the option	C
16	bidDisplayPrice	Price	The display price being bid for the option (can be zero in two-sided quote which supports spread quotes in low prices names)	C
17	bidDisplayQty	Unsigned	The display quantity being bid for the option (can be zero in two-sided quote which supports spread quotes in low prices names)	C
18	askDisplayPrice	Price	The display price being asked for the option	C
19	askDisplayQty	Unsigned	The display quantity being asked for the option	C
20	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C
21	askQuoteID	Text (40)	When onlyOneQuote=True, the unique identifier assigned to the ask by the exchange. If this field is populated, then the quoteID value will only be applied to the bid. When onlyOneQuote=False, the unique identifier assigned to the ask. To provide a unique identifier for a bid, use the quoteID field. If this field is populated, then the askPrice must also be populated.	C
22	originalAskQuoteID	Text (40)	When onlyOneQuote=False, this field must be populated when the ask from this record replaces a previously submitted ask. This field must not be provided for an ask that does not replace a previous	C

Option Quote (OQ)				
#	Field Name	Data Type	Description	Include Key
			ask, and it should never be populated for a bid. When onlyOneQuote=True no value is necessary for this field.	
23	initiator	Choice	Specifies who initiated the quote: the market maker or exchange	O

Linkage Keys for **OQ**:

- **Quote Key:** date, exchange, optionID, quoteID, askQuoteID
- **Previous Quote Key:** date, exchange, optionID, originalQuoteID, originalAskQuoteID

5.1.2.Quote Cancel Event

The following data elements are required for cancel quote events.

Table 33: Quote Cancel Events

Option Quote Cancel (OQC)				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	OQC	R
2	exchange	Exchange ID	The identifier for the exchange processing the quote cancel	R
3	eventTimestamp	Timestamp	The date/time when the quote cancel occurred	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	marketMaker	Member Alias	The Member Alias assigned by the SRO to identify the market maker issuing the quote cancel. In the case where a market maker has multiple users (e.g., acronyms used to differentiate users within the same MM), there would be a separate Member Alias given to each user or sub-account	R
7	sentTimestamp	Timestamp	The date/time when the market maker sent the quote cancel to the exchange. This field is only required if the cancel initiator is the market maker	C
8	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R

Option Quote Cancel (OQC)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
9	quoteID	Text (40)	The unique identifier assigned to this quote to be canceled by the exchange. This field can be omitted if onlyOneQuote is true If onlyOneQuote=False, then this field will only be used to cancel bids. To cancel an ask, provide the relevant identifier in the askQuoteID field.	C
10	onlyOneQuote	Boolean	True if the system allows only one quote for this market maker; false otherwise	R
11	initiator	Choice	Specifies who initiated the cancel: the market maker or exchange	R
12	cancelReason	Choice	This code represents the reason why the quote was canceled. The actual value of the code is exchange specific. See Data Dictionary for the list of allowed values	O
13	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C
14	askQuoteID	Text (40)	The unique identifier assigned to cancel the ask to be canceled by the exchange. This field can be omitted if onlyOneQuote is true. If onlyOneQuote=False, then this field will only be used to cancel asks. To cancel a bid, provide the relevant identifier in the quoteID field.	C

Linkage Keys for **OQC**:

- **Quote Key:** date, exchange, optionID, quoteID, askQuoteID

5.2. Options Orders

Order events for options are reported as either simple or complex. Simple option orders are orders for a single option series (including flex options). Complex option orders contain two or more simple option orders, or at least one each of a simple option order and equity order.

For CAT, an order for a complex option will be reported at the parent complex level, and additional orders will be reported if/when orders are created for each leg. Some exchanges create leg order reporting events as soon as the parent is created, and other exchanges create leg order reporting events only when an execution is created. CAT supports both reporting scenarios. In the latter case, when no leg executions occur, it is possible that no leg-level order events are generated.

Each options order routed to (and then accepted by) an exchange must be reported to CAT. Options orders that are routed to an exchange and then rejected by the exchange are not reportable by the exchange. When an exchange accepts an options order, it must report either a single Option Order Accepted event, or a single Complex Option Order Accepted event followed by one Accepted event for each leg of the complex option.

For manual/floor trades, this will be the identifier for the physical broker. For quotes, it will be an alias for the market maker behind the quote. For system trades, it will be an alias for the system handling that order.

5.2.1. Order Accepted Events

5.2.1.1. Simple Option Order Accepted Event

A simple option order can represent either a stand-alone option series, or one leg of a complex parent order. If the order represents a leg of a complex order, then the field Complex Order ID will be set to the Order ID of the parent complex order. If necessary, the event timestamp and sequence number could be the same as those in the parent complex order.

Fields marked with a lower-case 'r' are required if the event represents a normal option order, and they are conditional if the event represents a leg of a complex order.

Table 34: Simple Option Order Accepted Event

Simple Option Order Accepted (OOA)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OOA	R

Simple Option Order Accepted (OOA)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
2	exchange	Exchange ID	The identifier for the exchange which has received this order	R
3	eventTimestamp	Timestamp	The date/time of order receipt	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	routingParty	Text (8)	The ID string used to identify the entity that sent this routed order. This must be provided for simple options (i.e. complexOrderID is null). Leave unset if the option is a leg of a complex order.	C
9	routedOrderID	Text (40)	The ID assigned to this order by the client when submitting the order to the exchange. This must be provided for simple options (i.e. complexOrderID is null). Leave unset if the option is a leg of a complex order.	C
10	session	Text (40)	The name of the session used to send the order from the routing member firm to the exchange. This must be provided for simple options (i.e. complexOrderID is null). Leave unset if the option is a leg of a complex order	C
11	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values	R
12	price	Price	The limit price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
13	quantity	Unsigned	The order quantity	r
14	displayQty	Unsigned	The displayed quantity for this order	r
15	displayPrice	Price	The displayed price for this order. This must be provided on simple option orders (i.e. complexOrderID is null) when displayQty is greater than zero.	C
16	workingPrice	Price	The working price of the order at the time it was accepted. Note that Modified events must be	C

Simple Option Order Accepted (OOA)

#	Field Name	Data Type	Description	Include Key
			reported to CAT anytime the working price changes	
17	openCloseIndicator	Choice	the position of the order: either Open, Close, or Unspecified	R
18	orderType	Choice	The order type is one of several possible pre-defined order types. There are a few general order type codes, and several codes unique for each exchange. See the corresponding entry in the Data Dictionary for more details about order types.	R
19	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	r
20	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details	C
21	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order, that are not necessarily handling instructions	C
22	exchOriginCode	Choice	Exchange defined code designating the origin of the order, see data dictionary for list of acceptable values	r
23	coverage	Choice	Specifies whether the order is covered or uncovered. This field may also be filled in as unspecified. See the data dictionary for a list of acceptable values	r
24	executingFirm	Alphanumeric(8)	The OCC number of the executing/give-up firm	r
25	cmtaFirm	Alphanumeric(8)	The OCC number of the CMTA firm (only valid for CMTA trades)	C
26	member	Member Alias	The identifier for the member firm that is responsible for the order	R
27	mktMkrSubAccount	Text (20)	The sub-account for the market maker, only valid when Origin Code is Market Maker	C
28	nbbPrice	Price	The NBBO at the moment just before accepting this order.	R
29	nbbQty	Unsigned		O
30	nboPrice	Price		R
31	nboQty	Unsigned		O
32	complexOrderID	Text (40)	The Order ID for the parent complex order, if this order represents a leg of a complex order. This must be provided if the order represents a leg	C

Simple Option Order Accepted (OOA)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			of a complex order.	
33	complexOptionID	Text (40)	The optionID for the parent complex order, if this order represents a leg of a complex order. Not reported if the complex order's orderID is globally unique	C
34	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OOA**:

- **Order Key:** date, exchange, optionID, orderID
- **Cross Order Key:** date, exchange, orderID, pairedOrderId (if populated in order attributes name value pair)
- **Route Link Key:** date, optionID, routingParty, routedOrderID, session, exchange
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderID

5.2.1.2. Complex Option Order Accepted Event

Each complex option order routed to (and accepted by) an exchange must be reported to CAT. CAT allows each leg of a complex order to be reported separately, thus the parent order is relatively small with most order details reported on behalf of each leg. If possible, exchanges should report leg events for all complex orders whether or not the complex order executes. Leg events must be reported for all legs for all executed complex orders.

The number of legs, and description of each leg is encapsulated in the dictionary entry for the Option ID. In addition to the Complex Order Accepted event, at least one Option Order Accepted event must be submitted for each leg of a complex order (Stock Leg Order Accepted for non-option legs).

Some systems allow individual legs to carry specific instructions. Thus, order type information is relevant on a per-leg basis, and not reported for the complex parent itself. Furthermore, some exchange don't ever create leg orders within the trading system. Instead they create synthetic leg order events for CAT reporting purposes at the time of execution. For these exchanges, the execution triggers the leg level order accept events, so for complex orders that are canceled without executions there will not be any leg order events. Thus, the model supports both processes, where leg orders can be created upon initial acceptance and at the point of execution.

When a leg order is created, each leg must have a unique internal Order ID. Some reporters already create such derived order representations, so these IDs are easy to acquire. Others do not assign identifiers to legs. However, all reporters will be expected to report individual order events for each leg that is executed. One suggested method for creating unique leg Order IDs is to use the Order ID of the parent complex order, combined with the leg number (its ordering in the complex option definition). Another is to combine the Complex Order ID with the Option ID and Side of that leg.

Qualified Contingent Cross orders are not exposed to other market participants, and require special handling on some exchanges. In certain cases, these orders may only be processed as individual leg orders and no Complex Option Order Accepted event is reported.

Note that the following fields are conditional in this event. If they are present, then they do not have to appear in the individual order events for option legs, unless the value for a leg would be different from the value in the complex order. In other words, these field values apply to all option legs, unless the option leg contains a different value. If these fields are missing, then the data must be present in each option leg.

`coverage, exchOriginCode, executingFirm, cmtaFirm, mktMkrSubAccount`

Table 35: Complex Option Order Accepted Event

Complex Option Order Accepted (OCOA)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OCOA	R
2	exchange	Exchange ID	The identifier for the exchange which has received this order	R

Complex Option Order Accepted (OCOA)				
#	Field Name	Data Type	Description	Include Key
3	eventTimestamp	Timestamp	The date/time of order receipt	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R
7	orderID	Text (40)	The internal order ID assigned to the complex order by the exchange	R
8	side	Choice	The side of the order, for a complex order the values for side can be either "AsDirected" or "Opposite", see entry for "Side" in the Data Dictionary for acceptable values	R
9	routingParty	Text (8)	The ID string used to identify the entity that sent this routed order	R
10	routedOrderID	Text (40)	The ID assigned to this order by the routing firm when submitting the order to the exchange	R
11	session	Text (40)	The name of the session used to send the order from the routing member firm to the exchange	R
12	price	Price	the net price of the order, which may be negative	C
13	quantity	Unsigned	the order quantity	R
14	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
15	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details	C
16	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order, that are not necessarily handling instructions	C
17	isGloballyUnique	Boolean	If reported with a value of true, then the orderID is globally unique across all optionIDs for this exchange/date. This means that no other complex order can have the same orderID. Furthermore, leg events for this complex order must be reported with just the complexOrderID and not the complexOptionID	O
18	member	Member Alias	The identifier for the member firm that is	R

Complex Option Order Accepted (OCOA)				
#	Field Name	Data Type	Description	Include Key
			responsible for the order	
19	exchOriginCode	Choice	Exchange defined code designating the origin of the order, see data dictionary for list of acceptable values	C
20	coverage	Choice	Specifies whether the order is covered or uncovered. This field may also be filled in as unspecified. See the data dictionary for a list of acceptable values	C
21	executingFirm	Alphanumeric(8)	The OCC number of the executing/give-up firm	C
22	cmtaFirm	Alphanumeric(8)	The OCC number of the CMTA firm (only valid for CMTA trades)	C
23	mktMkrSubAccount	Text (20)	The sub-account for the market maker, only valid when Origin Code is Market Maker	C
24	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OCOA**:

- **Order Key:** date, exchange, optionID, orderID (if isGloballyUnique is false)
- **Order Key:** date, exchange, orderID (if isGloballyUnique is true)
- **Cross Order Key:** date, exchange, optionID, orderID, pairedOrderId (if pairedOrderId is populated in order attributes name value pair and isGloballyUnique is false)
- **Cross Order Key:** date, exchange, orderID, pairedOrderId (if pariedOrderId is populated in order attributes name value pair and isGloballyUnique is true)

- **Route Link Key:** date, optionID, routingParty, routedOrderID, session, exchange

5.2.1.3. Stock Leg Order Event

Similar to option legs, stock legs are reported individually, with a link to the parent complex order. If necessary, the event timestamp and sequence number could be the same as those in the parent complex order.

See the explanation about leg Order IDs in the section on complex orders. The same process applies to Order IDs for stock legs.

Table 36: Stock Leg Event

Option Stock Leg (OSL)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OSL	R
2	exchange	Exchange ID	The identifier for the exchange which has accepted this order	R
3	eventTimestamp	Timestamp	The date/time of order receipt	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	side	Choice	The side of the order: See entry for "Side" in the Data Dictionary for acceptable values	R
9	price	Price	the limit price of the order, if applicable	C
10	quantity	Unsigned	the order quantity	R
11	displayQty	Unsigned	the displayed quantity for this order	R
12	orderType	Choice	The order type is one of several possible pre-defined order types. There are a few general order type codes, and several codes unique for each exchange. See the corresponding entry in the Data Dictionary for more details about order types	R
13	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC,	R

Option Stock Leg (OSL)				
#	Field Name	Data Type	Description	Include Key
			GTC). See the Data Dictionary for a complex list of acceptable values	
14	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details.	C
15	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order that are not necessarily handling instructions	C
16	clearingFirm	Text (10)	Firm receiving the stock execution	O
17	nbbPrice	Price	The NBBO at the moment the order was accepted	R
18	nbbQty	Unsigned		O
19	nboPrice	Price		R
20	nboQty	Unsigned		O
21	complexOrderID	Text (40)	The Order ID for the parent complex order. This must be provided if the order represents a leg of a complex order.	R
22	complexOptionID	Text (40)	The optionID for the parent complex order. Not reported if the complex order's orderID is globally unique	C
23	member	Member Alias	The identifier for the member firm that is responsible for the order. This is the same member as in the complex order	R
24	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the	C

Option Stock Leg (OSL)				
#	Field Name	Data Type	Description	Include Key
			next Trade Date.)	

Linkage Keys for **OSL**:

- **Order Key:** date, exchange, symbol, orderID
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderID

5.2.2.Order Modified Events

An event must be sent to CAT to report any customer modification to the order. Additionally, an event must be sent to CAT to report any changes to the order due to an exchange action, including updates related to changes in market conditions.

Events that should be reported include, but are not limited to:

- Any customer update that passes validation and is successfully processed by the trading system
- Changes to the available quantity of the order, such as liquidity returning from an away market unexecuted
- Changes to the working price, display price, or display quantity
- Changes to the executability of an order, such as when a regular-hours order arrived prior to the opening time and it is now the opening time or when an order expires but no explicit cancellation is provided

5.2.2.1. Option Order Modified Event

This event supports all possible modifications to a simple option order and the legs of multi-leg orders. The full state of the order should be reported, including fields that did not change as a result of the modification.

Table 37: Option Order Modified Event

Option Order Modified (OOM)				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	OOM	R
2	exchange	Exchange ID	The identifier for the exchange which has received this order	R

Option Order Modified (OOM)				
#	Field Name	Data Type	Description	Include Key
3	eventTimestamp	Timestamp	The date/time at which the modification was received or originated	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R
7	orderID	Text (40)	The internal order ID used by the exchange to refer to this order from this point forward	R
8	coverage	Choice	Specifies whether the order is covered or uncovered. This field may also be filled in as unspecified. See the data dictionary for a list of acceptable values	R
9	originalOrderID	Text (40)	The internal order ID that used to be assigned to this order until this modification happened. If the order kept its ID through the modification, then this value need not be included	C
10	initiator	Choice	Indicates who initiated the order modification: See entry for "initiator" in the Data Dictionary for acceptable values	R
11	nbbPrice	Price	The NBBO at the moment the modification took place	R
12	nbbQty	Unsigned		O
13	nboPrice	Price		R
14	nboQty	Unsigned		O
15	price	Price	The limit price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
16	Quantity	Unsigned	When the initiator field is set to Firm or Market Maker, the order quantity. Conditional if the order represents a leg of a complex order; otherwise Required. When the initiator field is set to Exchange, the total quantity available on the local book at the conclusion of the modification. This must be provided for simple option orders (i.e. complexOrderID is null) when initiator is 'Firm' or 'MarketMaker'.	C
17	leavesQty	Unsigned	The quantity left open after the modification has	R

Option Order Modified (OOM)				
#	Field Name	Data Type	Description	Include Key
			occurred	
18	displayQty	Unsigned	The displayed quantity for this order. This must be provided for simple option orders (i.e. complexOrderID is null). Conditional if the order represents a leg of a complex order.	C
19	displayPrice	Price	The displayed price for this order. This must be provided on simple option orders (i.e. complexOrderID is null) when displayQty is greater than zero.	C
20	workingPrice	Price	The working price of the order	C
21	openCloseIndicator	Choice	the position of the order: either Open, Close, or Unspecified	R
22	orderType	Choice	The order type is one of several possible pre-defined order types. There are a few general order type codes, and several codes unique for each exchange. See the corresponding entry in the Data Dictionary for more details about order types.	R
23	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values. This must be provided for simple option orders (i.e. complexOrderID is null). Conditional if the order represents a leg of a complex order.	C
24	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details	C
25	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order that are not necessarily handling instructions	C
26	exchOriginCode	Choice	Exchange defined code designating the origin of the order, see data dictionary for list of acceptable values	R
27	executingFirm	Alphanumeric (8)	The OCC number of the executing/give-up firm	R
28	cmtaFirm	Alphanumeric (8)	The OCC number of the CMTA firm (only valid for CMTA trades)	C
29	member	Member Alias	The identifier for the member firm that is responsible for the order	R
30	mktMkrSubAccount	Text (20)	The sub-account for the market maker, only valid when Origin Code is Market Maker	C

Option Order Modified (OOM)				
#	Field Name	Data Type	Description	Include Key
31	complexOrderID	Text (40)	The Order ID for the parent complex order, if this order represents a leg of a complex order. If the ID for the complex order also changed, then this would be the new Order ID for the complex order. This must be provided if the order represents a leg of a complex order.	C
32	complexOptionID	Text (40)	The optionID for the parent complex order, if this order represents a leg of a complex order. Not reported if the complex order's orderID is globally unique	C
33	routedOrderID	Text (40)	For customer-driven changes to the order, the ID assigned to this order by the routing firm when submitting the modification to the exchange. For the return of unexecuted liquidity previously routed away, the exchange-assigned ID used to route the order away. Except as noted above, not required for exchange-driven modifications. This must be provided for simple option orders (i.e. complexOrderID is null).	C
34	side	Choice	The side of the order. See entry for “side” in the Data Dictionary for acceptable values. Should be provided for firm or market maker updates to an order. Should be reported even if it has not changed from the prior version of the order. This must be provided when initiator is ‘Firm’ or ‘MarketMaker’.	C
35	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The	C

Option Order Modified (OOM)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	
36	routingParty	Text(8)	For customer-driven changes to the order, the ID string used to identify the entity that routed this order modification to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID string used to route the order away. Should match the value of the OOR event routingParty with the same routedOrderID. Except as noted above, not required for exchange-driven modifications.	C
37	session	Text(40)	For customer-driven changes to the order, the ID assigned to the specific session that the routing member used to route the order to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID used to route the order away. Should match the value of the OOR event session with the same routedOrderID. Except as noted above, not required for exchange-driven modifications.	C

Linkage Keys for **OOM**:

- **Order Key:** date, exchange, optionID, orderID
- **Cross Order Key:** date, exchange, orderID, pairedOrderId (if pairedOrderId is populated in order attributes name value pair)
- **Previous Order Key:** date, exchange, optionID, originalOrderID
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderID

5.2.2.2. Complex Option Order Modified Event

If the price or quantity changes on a complex order, a complex option order modified event needs to be submitted to CAT. If a change to the parent complex order causes attributes in the leg orders to change, then Order Modified events must be reported for each affected leg. Note that this only applies if a leg order actually exists at the time of the modification to the complex order. For exchanges that create leg orders at execution, only the complex order needs to be modified. However, if a change in net price to the complex order causes the price of the leg orders to change, changes to the leg order prices are not reportable to CAT.

If the internal order ID of the complex order changes, then modified reports must be generated for every leg that exists at the time of the modification, referencing the new order ID of the parent complex order.

The full state of the modified order must be reported, including fields that did not change value as a result of the modification.

Table 38: Complex Option Order Modified Event

Complex Option Order Modified (OCOM)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OCOM	R
2	exchange	Exchange ID	The identifier for the exchange which has received this order	R
3	eventTimestamp	Timestamp	The date/time at which the modification was received or originated	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R
7	orderID	Text (40)	The internal order ID assigned to the complex order by the exchange	R
8	originalOrderID	Text (40)	The internal order ID that used to be assigned to this order until this modification happened. If the order kept its ID through the modification, then this value need not be included	C
9	initiator	Choice	Indicates who initiated the order modification: See entry for "initiator" in the Data Dictionary for acceptable values	R
10	price	Price	The net price of the order, which may be negative	C
11	quantity	Unsigned	The order quantity	R
12	leavesQty	Unsigned	The quantity left open after the modification has occurred	R
13	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
14	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation	C

Complex Option Order Modified (OCOM)				
#	Field Name	Data Type	Description	Include Key
			in the Data Dictionary for more details	
15	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order that are not necessarily handling instructions	C
16	member	Member Alias	The identifier for the member firm that is responsible for the order	R
17	routedOrderID	Text (40)	For customer-driven changes to the order, the ID assigned to this order by the routing firm when submitting the modification to the exchange. For the return of unexecuted liquidity previously routed away, the exchange-assigned ID used to route the order away. Except as noted above, not required for exchange-driven modifications	C
18	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C
19	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values. Should be provided for firm or market maker updates to an order. Should be reported even if it has not changed from the prior version of the order.	C
20	routingParty	Text(8)	For customer-driven changes to the order, the ID string used to identify the entity that routed this order modification to the exchange.	C

Complex Option Order Modified (OCOM)				
#	Field Name	Data Type	Description	Include Key
			For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID string used to route the order away. Should match the value of the OOR event routingParty with the same routedOrderID. Except as noted above, not required for exchange-driven modifications.	
21	session	Text(40)	For customer-driven changes to the order, the ID assigned to the specific session that the routing member used to route the order to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID used to route the order away. Should match the value of the OOR event session with the same routedOrderID. Except as noted above, not required for exchange-driven modifications.	C

Linkage Keys for **OCOM**:

- **Order Key:** date, exchange, optionID, orderID
- **Cross Order Key:** date, exchange, orderID, pairedOrderId (if pariedOrderId is populated in order attributes name value pair)
- **Route Link Key:** date, optionID, routedOrderID, exchange, routingParty, session
- **Previous Order Key:** date, exchange, optionID, originalOrderID

5.2.2.3. Stock Leg Modified Event

When a stock leg is modified, an event must be reported to CAT with the modified data elements. The full state of the modified order must be reported, including fields that did not change value as a result of the modification.

Table 39: Stock Leg Modified Event

Option Stock Leg Modified				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	OSLM	R
2	exchange	Exchange ID	The identifier for the exchange which has accepted this order	R
3	eventTimestamp	Timestamp	The date/time at which the modification was received	R

Option Stock Leg Modified				
#	Field Name	Data Type	Description	Include Key
			or originated	
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text(10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	originalOrderID	Text (40)	The internal order ID that used to be assigned to this order until this modification happened. If the order kept its ID through the modification, then this value need not be included	C
9	initiator	Choice	Indicates who initiated the order modification: See entry for "initiator" in the Data Dictionary for acceptable values	R
10	nbbPrice	Price	The NBBO at the moment the stock leg was modified.	R
11	nbbQty	Unsigned		O
12	nboPrice	Price		R
13	nboQty	Unsigned		O
14	complexOrderID	Text (40)	The Order ID for the parent complex order. If the ID for the complex order also changed, then this would be the new Order ID for the complex order. This must be provided if the order represents a leg of a complex order.	R
15	complexOptionID	Text (40)	The optionID for the parent complex order. Not reported if the complex order's orderID is globally unique	C
16	price	Price	the limit price of the order, if applicable	C
17	displayPrice	Price	The displayed price for this order (required if displayQty is nonzero)	C
18	quantity	Unsigned	The order quantity	R
19	leavesQty	Unsigned	The number of shares left open after the modification has occurred	R
20	displayQty	Unsigned	The displayed quantity for this order	R

Option Stock Leg Modified				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
21	orderType	Choice	The order type is one of several possible pre-defined order types. There are a few general order type codes, and several codes unique for each exchange. See the corresponding entry in the Data Dictionary for more details about order types	R
22	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
23	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details	C
24	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order that are not necessarily handling instructions	C
25	clearingFirm	Text (10)	Firm receiving the stock execution	O
26	member	Member Alias	The identifier for the member firm that is responsible for the order. This is the same member as in the complex order	R
27	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OSLM**:

- **Order Key:** date, exchange, symbol, orderID
- **Previous Order Key:** date, exchange, symbol, originalOrderID
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderID

5.2.2.4. Option Order Adjusted Event

This event supports changes to the price, quantity, working price, display price, and display quantity. All other order instructions are assumed to be unchanged.

Table 40: Option Order Adjusted Event

Option Order Adjusted (OOJ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OOJ	R
2	exchange	Exchange ID	The identifier for the exchange which has received this order	R
3	eventTimestamp	Timestamp	The date/time at which the modification was received or originated	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R
7	orderID	Text (40)	The internal order ID used by the exchange to refer to this order from this point forward	R
8	originalOrderID	Text (40)	The internal order ID that used to be assigned to this order until this modification happened. If the order kept its ID through the modification, then this value need not be included	C
9	initiator	Choice	Indicates who initiated the order modification: See entry for "initiator" in the Data Dictionary for acceptable values	R
10	price	Price	The limit price of the order	C
11	displayPrice	Price	The displayed price for this order. This must be provided on simple option orders (i.e. complexOrderID is null) when displayQty is greater than zero.	C
12	workingPrice	Price	The working price of the order	C
13	Quantity	Unsigned	When the initiator field is set to Firm or Market Maker, the order quantity. Conditional if the order represents a leg of a complex order; otherwise Required. When the initiator field is set to Exchange, the total	C

Option Order Adjusted (OOJ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			quantity available on the local book at the conclusion of the modification. This must be provided for simple option orders (i.e. complexOrderID is null) when initiator is 'Firm' or 'MarketMaker'.	
14	displayQty	Unsigned	The displayed quantity for this order. This must be provided for simple option orders (i.e. complexOrderID is null).	C
15	leavesQty	Unsigned	The quantity left open after the modification has occurred	C
16	nbbPrice	Price	The NBBO at the moment the stock leg was modified	R
17	nbbQty	Unsigned		O
18	nboPrice	Price		R
19	nboQty	Unsigned		O
20	complexOrderID	Text (40)	The Order ID for the parent complex order, if this order represents a leg of a complex order. If the ID for the complex order also changed, then this would be the new Order ID for the complex order. This must be provided if the order represents a leg of a complex order.	C
21	complexOptionID	Text (40)	The optionID for the parent complex order, if this order represents a leg of a complex order. Not reported if the complex order's orderID is globally unique	C
22	member	Member Alias	The identifier for the member firm that is responsible for the order	R
23	routedOrderID	Text (40)	For customer-driven changes to the order, the ID assigned to this order by the routing firm when submitting the modification to the exchange. For the return of unexecuted liquidity previously routed away, the exchange-assigned ID used to route the order away. Except as noted above, not required for exchange-driven modifications. This must be provided for simple option orders (i.e. complexOrderID is null).	C
24	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values. Should be provided for firm or market maker updates to an order. Should be reported even if it has not changed from the prior version of the order.	C

Option Order Adjusted (OOJ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			This must be provided when initiator is 'Firm' or 'MarketMaker'.	
25	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C
26	routingParty	Text(8)	For customer-driven changes to the order, the ID string used to identify the entity that routed this order modification to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID string used to route the order away. Should match the value of the OOR event routingParty with the same routedOrderID. Except as noted above, not required for exchange-driven modifications.	C
27	session	Text(40)	For customer-driven changes to the order, the ID assigned to the specific session that the routing member used to route the order to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID used to route the order away. Should match the value of the OOR event session with the same routedOrderID. Except as noted above, not required for exchange-driven modifications.	C

Linkage Keys for **OOJ**:

- **Order Key:** date, exchange, optionID, orderID

- **Cross Order Key:** date, exchange, orderID, pairedOrderId (if pariedOrderId is populated in order attributes name value pair)
- **Route Link Key:** date, optionID, routedOrderID, exchange, routingParty, session
- **Previous Order Key:** date, exchange, optionID, originalOrderID
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderID

5.2.2.5. Complex Option Order Adjusted Event

When a complex option is modified in such a way that only impacts the price and/or quantity, an instance of this event can be reported in place of the Complex Option Order Modified event.

The only types of modifications that are allowed to be reported with this event are changes to the price or quantity of the order.

For changes in quantity, both quantity and leavesQty are required (i.e., either both are reported or neither are reported).

Table 41: Complex Option Order Adjusted Event

Complex Option Order Adjusted (OCOJ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OCOJ	R
2	exchange	Exchange ID	The identifier for the exchange which has received this order	R
3	eventTimestamp	Timestamp	The date/time at which the modification was received or originated	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R
7	orderID	Text (40)	The internal order ID assigned to the complex order by the exchange	R
8	originalOrderID	Text (40)	The internal order ID that used to be assigned to this order until this modification happened. If the order kept its ID through the modification, then this value need not be included	C
9	initiator	Choice	Indicates who initiated the order modification: See	R

Complex Option Order Adjusted (OCOJ)				
#	Field Name	Data Type	Description	Include Key
			entry for "initiator" in the Data Dictionary for acceptable values	
10	price	Price	The net price of the order, which may be negative	C
11	quantity	Unsigned	The order quantity	C
12	leavesQty	Unsigned	The quantity left open after the modification has occurred	C
13	member	Member Alias	The identifier for the member firm that is responsible for the order	R
14	routedOrderID	Text (40)	For customer-driven changes to the order, the ID assigned to this order by the routing firm when submitting the modification to the exchange. For the return of unexecuted liquidity previously routed away, the exchange-assigned ID used to route the order away. Except as noted above, not required for exchange-driven modifications	C
15	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C
16	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values. Should be provided for firm or market maker updates to an order. Should be reported even if it has not changed from the prior version of the order.	C

Complex Option Order Adjusted (OCOJ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
17	routingParty	Text(8)	For customer-driven changes to the order, the ID string used to identify the entity that routed this order modification to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID string used to route the order away. Should match the value of the OOR event routingParty with the same routedOrderID. Except as noted above, not required for exchange-driven modifications.	C
18	session	Text(40)	For customer-driven changes to the order, the ID assigned to the specific session that the routing member used to route the order to the exchange. For the return of unexecuted liquidity previously routed away, optionally provide the exchange-assigned ID used to route the order away. Should match the value of the OOR event session with the same routedOrderID. Except as noted above, not required for exchange-driven modifications.	C

Linkage Keys for **OCOJ**:

- **Order Key:** date, exchange, optionID, orderID
- **Cross Order Key:** date, exchange, optionID, orderID, pairedOrderId (if pairedOrderId is populated in order attributes name value pair)
- **Route Link Key:** date, optionID, routedOrderID, exchange, routingParty, session
- **Previous Order Key:** date, exchange, optionID, originalOrderID,

5.2.2.6. Stock Leg Adjusted Event

When a stock leg is modified where it only impacts the price and/or quantity, an instance of this event can be reported in place of the Stock Leg Modified event.

For changes in quantity, both quantity and leavesQty are required (i.e., either both are reported or neither are reported).

Table 42: Stock Leg Adjusted Event

Option Stock Leg Adjusted (OSLJ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OSLJ	R
2	exchange	Exchange ID	The identifier for the exchange which has accepted this order	R
3	eventTimestamp	Timestamp	The date/time at which the modification was received or originated	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	originalOrderID	Text (40)	The internal order ID that used to be assigned to this order until this modification happened. If the order kept its ID through the modification, then this value need not be included	C
9	initiator	Choice	Indicates who initiated the order modification: See entry for "initiator" in the Data Dictionary for acceptable values	R
10	price	Price	The limit price of the order	C
11	displayPrice	Price	The displayed price for this order	C
12	quantity	Unsigned	The order quantity	C
13	leavesQty	Unsigned	The quantity left open after the modification has occurred.	C
14	displayQty	Unsigned	The displayed quantity for this order	C
15	nbbPrice	Price	The NBBO at the moment the stock leg was modified.	R
16	nbbQty	Unsigned		O
17	nboPrice	Price		R
18	nboQty	Unsigned		O
19	complexOrderID	Text (40)	The Order ID for the parent complex order. If the ID for the complex order also changed, then this would	R

Option Stock Leg Adjusted (OSLJ)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			be the new Order ID for the complex order. This must be provided if the order represents a leg of a complex order.	
20	complexOptionID	Text (40)	The optionID for the parent complex order. Not reported if the complex order's orderID is globally unique	C
21	member	Member Alias	The identifier for the member firm that is responsible for the order. This is the same member as in the complex order	R
22	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OSLJ**:

- **Order Key:** date, exchange, symbol, orderID
- **Previous Order Key:** date, exchange, symbol, originalOrderID
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderIDOptions

5.2.3.Order Canceled Event

An order canceled event is used to report a cancelation of a simple option order or a complex option order. For complex options orders, if leg-level orders have been opened before a canceled event, then canceled events must be reported for each of the leg orders as well.

CAT also supports partial cancels. Partial canceled events for complex orders follow the same rule, if there are open leg-level orders before a canceled event, partial canceled events must also be reported for each of the legs.

Note that the order canceled events contains both the fields optionID and symbol. Both of these fields are conditional. If the order canceled event is for a stock leg order corresponding to a complex option order, then the symbol field is mandatory. If the order canceled event is for a simple option order, a complex option order, or an option leg order of a complex order, then the field optionID is mandatory.

Table 43: Option Order Canceled

Option Order Canceled (OOC)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OOC	R
2	exchange	Exchange ID	The ID for the exchange reporting the order canceled	R
3	eventTimestamp	Timestamp	The date/time at which the cancellation was received or originated	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option dictionary. Used if this cancel is for a simple option order or complex option order. Either optionID or symbol, but not both, must be provided.	C
7	symbol	Symbol	The stock symbol in the symbology of the listing exchange, or the reporter's symbology mapping as appropriate. Used only if this cancel is for the stock leg of a complex option order. Either optionID or symbol, but not both, must be provided.	C
8	orderID	Text (40)	The internal order ID assigned to the order by the exchange. If a leg is being canceled, the orderID will represent the leg order being canceled	R
9	cancelQty	Unsigned	The quantity being canceled	R
10	leavesQty	Unsigned	The quantity left open after the cancel event (zero for a full cancel)	R
11	initiator	Choice	Indicates who initiated the order cancellation: See entry for "initiator" in the Data Dictionary for	R

Option Order Canceled (OOC)				
#	Field Name	Data Type	Description	Include Key
			acceptable values	
12	cancelReason	Choice	Code representing the reason why the order was canceled. The actual value of the code is exchange specific. See Data Dictionary for the list of allowed values	O
13	member	Member Alias	The identifier for the member firm that is responsible for the order	R
14	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OOC**:

- **Order Key:** date, exchange, optionID, orderID
- **Order Key:** date, exchange, symbol, orderID

5.2.4.Routing Orders

5.2.4.1. Internal Routing and Floor Activity

Internal routes on the exchange are different from internal routes in a Broker Dealer. In particular, internal routes at a broker dealer are required to be reported to CAT, but internal routes at an exchange are not.

However, there are cases where knowing the system or process of where an order executed is useful, for example when orders are routed through various internal systems on the floor. These processes differ between exchanges and the use cases are incredibly diverse. Furthermore, there is no guidance in the

CAT requirements as to what is or is not supposed to be reported in these cases, so we need to be flexible in allowing a diverse set of items to be reported. These somewhat reportable data elements arrive in two forms.

First, an order may be executed with some additional information that was not available when it was placed (e.g., as part of an auction, or through some floor trading workstation). Thus, there is an element available on Trade Events (Execution Codes), which provides a way to add special exchange specific codes to an execution. The Execution Codes is a name/value pair field (like order Handling Instructions) and can provide additional execution information, like where a trade may have been executed on the floor, or supplemental execution/clearing information.

Additionally, the Note Event (reference Section 3.7.1), which contains either an Order ID or a Quote ID to link the note to a specific order or quote can be used to add specific instructions related to the order.

Some systems are composed of multiple subsystems, each having their own reporting and order identification requirements. In such cases, it may be extremely difficult or time consuming to coerce events into a single set of unique order IDs and reporting. Thus, an internal route event is also provided for reporting an order as it progresses between internal subsystems, and possibly changes internal order ID.

5.2.4.2. Option Route Event

External routes from an options exchange come in three basic forms: routing all or part of a simple option series order to an away market, routing two stock legs to be crossed, and routing a stock leg for execution. All of these events require certain pieces of information to enable linkage creation that can track the entire order lifecycle.

The following Option Route Event is used to report when an exchange routes a simple option order, or any leg of a complex option order.

Table 44: Option Route Event

Option Order Route (OOR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OOR	R
2	exchange	Exchange ID	The identifier for the exchange which is routing the order away	R
3	eventTimestamp	Timestamp	The date/time at which the order was routed	R

Option Order Route (OOR)				
#	Field Name	Data Type	Description	Include Key
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol of the stock leg being routed away - only for routing stock legs. Either optionID or symbol, but not both, must be provided.	C
7	optionID	Text (40)	The ID of the option being routed away. Either optionID or symbol, but not both, must be provided.	C
8	orderID	Text (40)	The internal order ID of the order being routed away	R
9	routingParty	Text (8)	The ID string used to identify the entity that is receiving this routed order	R
10	routedOrderID	Text (40)	The ID of the routed order, as represented in the order message sent to the routing broker	R
11	session	Text (40)	The ID of the session used to send the order to the routing broker	R
12	side	Choice	The side of the order: See entry for "Side" in the Data Dictionary for acceptable values	R
13	price	Price	The price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
14	quantity	Unsigned	The order quantity	R
15	displayQty	Unsigned	The displayed quantity for this order	R
16	orderType	Choice	The order type is one of several possible pre-defined order types. There are a few general order type codes, and several codes unique for each exchange. See the corresponding entry in the Data Dictionary for more details about order types	R
17	coverage	Choice	Specifies whether the order is covered or uncovered. This field may also be filled in as unspecified. See the data dictionary for a list of acceptable values	R
18	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
19	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the	C

Option Order Route (OOR)				
#	Field Name	Data Type	Description	Include Key
			Data Dictionary for more details	
20	result	Choice	The result of the route request. (A request can be ACK - Acknowledged, REJ - Rejected, or NR - No Response) See the data dictionary for a list of permissible values	O
21	resultTimestamp	Timestamp	The date/time when the exchange received the result of the route request. This timestamp is not required if the value for the result field is No Response	O
22	nbbPrice	Price	The NBBO at the moment just before routing this order	R
23	nbbQty	Unsigned		O
24	nboPrice	Price		R
25	nboQty	Unsigned		O
26	complexOrderID	Text (40)	The Order ID for the parent complex order, if this order represents a leg of a complex order. This must be provided if the order represents a leg of a complex order.	C
27	complexOptionID	Text (40)	The optionID for the parent complex order, if this order represents a leg of a complex order. Not reported if the complex order's orderID is globally unique	C
28	member	Member Alias	The identifier for the member firm that is responsible for the order	R
29	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OOR**:

- **Order Key:** date, exchange, optionID, orderID
- **Order Key:** date, exchange, symbol, orderID
- **Route Link Key:** date, optionID, routingParty, routedOrderID, exchange
- **Route Link Key:** date, symbol, routingParty, routedOrderID, exchange
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderID

5.2.4.3. Complex Option Route Event

This is used to report when an exchange routes a complex option order to an external destination (e.g. an exchange receives a complex order from a firm and forwards it to a Floor Broker). Note that most exchanges do not route complex orders externally; this event only applies for exchanges that do.

Table 45: Complex Option Route Event

Complex Option Order Route (OCOR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OCOR	R
2	exchange	Exchange ID	The ID for the exchange which is routing this order.	R
3	eventTimestamp	Timestamp	The date/time at which the order was routed.	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps.	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier.	C
6	optionID	Text (40)	The ID of the option being routed away.	R
7	orderID	Text (40)	The internal order ID of the order being routed away.	R
8	routingParty	Text (8)	The ID string used to identify the entity that is receiving this routed order.	R
9	routedOrderID	Text (40)	The ID of the routed order, as represented in the order message sent to the routing broker.	R
10	session	Text (40)	The ID of the session used to send the order to the destination.	R
11	side	Choice	The side of the order, for a complex order the values for side can be either "AsDirected" or "Opposite", see entry for "Side" in the Data	R

Complex Option Order Route (OCOR)				
#	Field Name	Data Type	Description	Include Key
			Dictionary for acceptable values	
12	price	Price	The net price of the order, which may be negative.	C
13	quantity	Unsigned	The order quantity.	R
14	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values.	R
15	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details.	C
16	result	Choice	The result of the route request. (A request can be ACK - Acknowledged, REJ - Rejected, or NR - No Response) See the data dictionary for a list of permissible values.	O
17	resultTimestamp	Timestamp	The date/time when the exchange received the result of the route request. This timestamp is not required if the value for the result field is No Response.	O
18	isGloballyUnique	Boolean	If reported with a value of true, then the orderID is globally unique across all optionIDs for this exchange/date. This means that no other complex order can have the same orderID. Furthermore, leg events for this complex order must be reported with just the complexOrderID and not the complexOptionID.	O
19	coverage	Choice	Specifies whether the order is covered or uncovered. This field may also be filled in as unspecified. See the data dictionary for a list of acceptable values.	C
20	member	Member Alias	The identifier for the member firm that is responsible for the order.	R
21	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T)	C

Complex Option Order Route (OCOR)				
#	Field Name	Data Type	Description	Include Key
			where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	

Linkage Keys for **OCOR**:

- **Order Key:** date, exchange, optionID, orderID (if isGloballyUnique is false)
- **Route Link Key:** date, exchange, routingParty, routedOrderID

5.2.4.4. Internal Option Route Event

This event provides a means by which options (and legs of complex options) can be routed between internal systems.

Table 46: Internal Option Route Event

Internal Option Route (OIR)				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	OIR	R
2	exchange	Exchange ID	The ID for the exchange which is routing this order	R
3	eventTimestamp	Timestamp	The date/time at which the order was routed	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol of the stock leg being routed away - only for routing stock legs. Either optionID or symbol, but not both, must be provided.	C
7	optionID	Text (40)	The ID of the option being routed away. Either optionID or symbol, but not both, must be provided.	C

Internal Option Route (OIR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
8	orderID	Text (40)	The internal order ID of the order being routed away	R
9	routingParty	Text (8)	The ID string used to identify the internal subsystem that is receiving this routed order. This value must match the value reported by the receiving subsystem in the `routingParty` field of their Order Accepted report	R
10	routedOrderID	Text (40)	The ID assigned to this order by the exchange when submitting the order to the subsystem. This value must match the value reported by the receiving subsystem in the `routedOrderID` field of their Order Accepted report	R
11	session	Text (40)	The ID assigned to the specific session used when sending the order from the sending subsystem to the receiving subsystem. This value must match the value reported by the receiving subsystem in the `session` field of their Order Accepted report	R
12	side	Choice	The side of the order: See entry for "Side" in the Data Dictionary for acceptable values	R
13	price	Price	The price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
14	quantity	Unsigned	The order quantity	R
15	displayQty	Unsigned	The displayed quantity for this order	R
16	orderType	Choice	The order type is one of several possible pre-defined order types. There are a few general order type codes, and several codes unique for each exchange. See the corresponding entry in the Data Dictionary for more details about order types	R
17	coverage	Choice	Specifies whether the order is covered or uncovered. This field may also be filled in as unspecified. See the data dictionary for a list of acceptable values	R
18	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
19	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details.	C
20	result	Choice	The result of the route request. (A request can be ACK - Acknowledged, REJ - Rejected, or NR - No Response) See the data dictionary for a list of	O

Internal Option Route (OIR)				
#	Field Name	Data Type	Description	Include Key
			permissible values	
21	resultTimestamp	Timestamp	The date/time when the exchange received the result of the route request. This timestamp is not required if the value for the result field is No Response	O
22	complexOrderID	Text (40)	The Order ID for the parent complex order, if this order represents a leg of a complex order. This must be provided if the order represents a leg of a complex order.	C
23	complexOptionID	Text (40)	The optionID for the parent complex order, if this order represents a leg of a complex order. Not reported if the complex order's orderID is globally unique	C
24	member	Member Alias	The identifier for the member firm that is responsible for the order	R
25	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OIR**:

- **Order Key:** date, exchange, optionID, orderID
- **Order Key:** date, exchange, symbol, orderID
- **Route Link Key:** date, optionID, routingParty, routedOrderID, session, exchange
- **Route Link Key:** date, symbol, routingParty, routedOrderID, session, exchange
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderID

5.2.4.5. Internal Complex Option Route Event

While complex orders are not routed between exchanges, they may be routed internally. This event provides a means by which complex options can be routed between internal systems.

Table 47: Internal Complex Option Route Event

Internal Complex Option Route (OCIR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OCIR	R
2	exchange	Exchange ID	The ID for the exchange which is routing this order	R
3	eventTimestamp	Timestamp	The date/time at which the order was routed	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID of the option being routed away	R
7	orderID	Text (40)	The internal order ID of the order being routed away	R
8	routingParty	Text (8)	The ID string used to identify the internal subsystem that is receiving this routed order. This value must match the value reported by the receiving subsystem in the `routingParty` field of their Order Accepted report	R
9	routedOrderID	Text (40)	The ID assigned to this order by the exchange when submitting the order to the subsystem. This value must match the value reported by the receiving subsystem in the `routedOrderID` field of their Order Accepted report	R
10	session	Text (40)	The ID assigned to the specific session used when sending the order from the sending subsystem to the receiving subsystem. This value must match the value reported by the receiving subsystem in the `session` field of their Order Accepted report	R
11	side	Choice	The side of the order, for a complex order the values for side can be either "AsDirected" or "Opposite", see entry for "Side" in the Data Dictionary for acceptable values	R
12	price	Price	The net price of the order, which may be negative.	C
13	quantity	Unsigned	The order quantity	R

Internal Complex Option Route (OCIR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
14	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
15	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details	C
16	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order that are not necessarily handling instructions	C
17	isGloballyUnique	Boolean	If reported with a value of true, then the orderID is globally unique across all optionIDs for this exchange/date. This means that no other complex order can have the same orderID. Furthermore, leg events for this complex order must be reported with just the complexOrderID and not the complexOptionID	O
18	exchOriginCode	Choice	Exchange defined code designating the origin of the order, see data dictionary for list of acceptable values	C
19	coverage	Choice	Specifies whether the order is covered or uncovered. This field may also be filled in as unspecified. See the data dictionary for a list of acceptable values	C
20	executingFirm	Alphanumeric(8)	The OCC number of the executing/give-up firm	C
21	cmtaFirm	Alphanumeric(8)	The OCC number of the CMTA firm (only valid for CMTA trades)	C
22	mktMkrSubAccount	Text (20)	The sub-account for the market maker, only valid when Origin Code is Market Maker	C
23	member	Member Alias	The identifier for the member firm that is responsible for the order	R
24	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T)	C

Internal Complex Option Route (OCIR)				
#	Field Name	Data Type	Description	Include Key
			where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	

Linkage Keys for **OCIR**:

- **Order Key:** date, exchange, optionID, orderID
- **Route Link Key:** date, optionID, routingParty, routedOrderID, session, exchange

5.2.4.6. Modify Option Route Event

When an exchange initiates a modify or cancel/replace request on an option or stock leg order it has previously routed away, it must report its intent to modify the order, using a Modify Option Route Event.

If the request does not change the routed order ID, then both routedOrderID and routedOriginalOrderID must be the same.

Note that the Modify Option Route event contains both the fields optionID and symbol. Both of these fields are conditional. If the Modify Option Route event is for a stock leg order, then the symbol field is mandatory and optionID field is not necessary. If the Modify Option Route event is for a simple option order, or an option leg order of a complex order, then the field optionID is mandatory.

Table 48: Modify Option Route Event

Modify Option Route (OOMR)				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	OOMR	R
2	exchange	Exchange ID	The ID for the exchange modifying the routed order	R
3	eventTimestamp	Timestamp	The date/time when the exchange made the modify request	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C

Modify Option Route (OOMR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias. Either optionID or symbol, but not both, must be provided.	C
7	optionID	Text (40)	The ID of the option being routed away. Either optionID or symbol, but not both, must be provided.	C
8	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
9	routingParty	Text (8)	The ID string used to identify the entity that received this routed order	R
10	routedOrderID	Text (40)	The new routed ID for the order, which will be used to refer to the routed order after the modification (in FIX, ClOrdID - in OUCH, Replacement Order Token)	R
11	routedOriginalOrderID	Text (40)	The routed ID for the order being modified, as sent to the routing broker in the original route message, or the most recent modify message (in FIX OrigClOrdID, in OUCH Existing Order Token)	R
12	session	Text (40)	The ID assigned to the session used to send the modify request from the exchange to the routing broker- must also match the session in the original Order Route message for this order	R
13	price	Price	The limit price of the order, if applicable. This must be provided when orderType indicates a limit order.	C
14	quantity	Unsigned	The order quantity	R
15	displayQty	Unsigned	The displayed quantity for this order	R
16	orderType	Choice	The type of order being submitted (e.g., market, limit). See the corresponding entry in the Data Dictionary for more details about order types	R
17	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
18	handlingInstructions	Name/Value Pairs	Can include zero or more handling instructions, as described in Data Dictionary for Handling Instructions	C

Modify Option Route (OOMR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
19	coverage	Choice	Specifies whether the order is covered or uncovered. This field may also be filled in as unspecified. See the data dictionary for a list of acceptable values	R
20	result	Choice	The result of the modify request. (A request can be ACK - Acknowledged, REJ - Rejected, or NR - No Response) See the data dictionary for a list of permissible values	O
21	resultTimestamp	Timestamp	The date/time when the exchange received the result of the modify request. This timestamp is not required if the value for the result field is No Response	O
22	nbbPrice	Price	The national best bid price at the moment the trade occurred	R
23	nbbQty	Unsigned	The national best bid quantity at the moment the trade occurred	O
24	nboPrice	Price	The national best offer price at the moment the trade occurred	R
25	nboQty	Unsigned	The national best offer quantity at the moment the trade occurred	O
26	member	Member Alias	The identifier for the member firm that is responsible for the order	R
27	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OOMR**:

- **Order Key:** date, exchange, optionID, orderID
- **Order Key:** date, exchange, symbol, orderID
- **Route Link Key:** date, optionID, routingParty, routedOrderID, exchange
- **Route Link Key:** date, symbol, routingParty, routedOrderID, exchange
- **Previous Route Link Key:** date, optionID, routingParty, routedOriginalOrderID, exchange
- **Previous Route Link Key:** date, symbol, routingParty, routedOriginalOrderID, exchange

5.2.4.7. Option Cancel Route Event

When an exchange initiates a cancel request on an order that has been previously routed away, it must report the intent to cancel, using an Option Cancel Route Event.

Note that the Option Cancel Route event contains both the fields `optionID` and symbol. Both of these fields are conditional. If the Option Cancel Route event is for a stock leg order, then the symbol field is mandatory and optionID field is not necessary. If the Option Cancel Route event is for a simple option order, or an option leg order of a complex order, then the field `optionID` is mandatory.

Table 49: Option Cancel Route Event

Option Cancel Route (OOCR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OOCR	R
2	exchange	Exchange ID	The ID for the exchange canceling the routed order	R
3	eventTimestamp	Timestamp	The date/time when the cancel request was sent to the routing firm	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias. Either optionID or symbol, but not both, must be provided.	C
7	optionID	Text (40)	The ID of the option being routed away. Either optionID or symbol, but not both, must be provided.	C

Option Cancel Route (OOCR)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
8	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
9	routingParty	Text (8)	The ID string used to identify the entity that received this routed order. This value will match the value on the Route event for the order being canceled	R
10	routedOrderID	Text (40)	The routed ID for the order being canceled - must also match the routedOrderID in the original Order Route message for this order	R
11	session	Text (40)	The session ID on which the cancel request is being made - must also match the session in the original Order Route message for this order	R
12	desiredLeavesQty	Unsigned	The desired number of shares remaining in the order after the cancel request has been issued. A value of zero indicates a full cancel	R
13	result	Choice	The result of the cancel request. (A request can be ACK - Acknowledged, REJ - Rejected, or NR - No Response) See the data dictionary for a list of permissible values	O
14	resultTimestamp	Timestamp	The date/time when the exchange received the result of the cancel request. This timestamp is not required if the value for the result field is No Response	O
15	member	Member Alias	The identifier for the member firm that is responsible for the order	R
16	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OOCR**:

- **Order Key:** date, exchange, optionID, orderID
- **Order Key:** date, exchange, symbol, orderID
- **Route Link Key:** date, optionID, routingParty, routedOrderID, exchange
- **Route Link Key:** date, symbol, routingParty, routedOrderID, exchange

5.2.5.Trades and Fills

All trades on an options exchange involving options are reported as two sided trades, with appropriate clearing information for each side. In the case where an order is routed away, the trade is still reported as a two-sided trade, but without an order on one side (that side will just have clearing information).

Trades off-exchange for non-option legs are reported as one-sided pass through fill events. Note the difference between a trade which the exchange transacted and a fill which the exchange is passing on. Both events are reportable, but they will be reported in different ways. The former as a two-sided trade, and the latter as either a one-sided fill.

5.2.5.1. Simple Option Trade Event

Simple option trade events are two-sided trade reports, providing details about both sides of the trade for an option. The same event is used for both simple options trades and trades for each leg of a complex option.

This section will deal only with simple option trades, the following section will demonstrate how the same event type will be used to report trades at the leg level of complex options.

Option Trade Event

Each option trade contains the following data elements.

Table 50: Option Trade Event

Option Trade (OT)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OT	R
2	exchange	Exchange ID	The ID of the participant reporting the trade event to CAT	R
3	eventTimestamp	Timestamp	The date/time of execution	R

Option Trade (OT)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	tradeID	Text (40)	This ID will be used when a specific trade needs to be identified, for example in trade break and correction reports. The combination of date, exchange, optionID, and tradeID must be globally unique	R
7	optionID	Text (40)	The ID of the option being traded	R
8	quantity	Unsigned	Quantity of the trade	R
9	price	Price	Price of the trade	R
10	nbbPrice	Price	The NBBO for this particular option series at the moment the event takes place	R
11	nbbQty	Unsigned		O
12	nboPrice	Price		R
13	nboQty	Unsigned		O
14	saleCondition	Text (8)	Conditions under which trade was executed	C
15	executionCodes	Name / Value Pairs	Adds special exchange specific codes to an execution. Zero or more codes can be entered to provide additional execution information, like where a trade may have been executed on the floor. These codes apply to both sides of the trade	C
16	buyDetails	Side Trade Details	Information for the buy side of the trade. Format and element definitions for Buy Details are described in Side Trade Details in Table 54	R
17	sellDetails	Side Trade Details	Information for the sell side of the trade. Format and element definitions for Sell Details are described in Side Trade Details in – Table 54	R
18	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the	C

Option Trade (OT)				
#	Field Name	Data Type	Description	Include Key
			Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	

Side Trade Details

Each side of a trade contains information pertinent to the order and/or quote that contributed to the trade. The Side Trade Details captures those data elements.

Table 51: Side Trade Details

Side Trade Details				
#	Field Name	Data Type	Description	Include Key
16.*n*.1 / 17.*n*.1	side	Choice	The side of the executed trade: See entry for "Side" in the Data Dictionary for acceptable values	R
16.*n*.2 / 17.*n*.2	leavesQty	Unsigned	The quantity remaining unfilled after this trade event. Not required when used in a trade correction	C
16.*n*.3 / 17.*n*.3	openCloseIndicator	Choice	Indicates the position of the trade, applicable only when this side is an order	C
16.*n*.4 / 17.*n*.4	quoteID	Text (40)	The ID of the quote, only applicable only when this side of the execution is a market maker quote	C
16.*n*.5/ 17.*n*.5	orderID	Text (40)	The ID of the order, only applicable only when this side of the execution is an order	C
16.*n*.6 / 17.*n*.6	executingFirm	Alphanumeric (8)	The OCC number of the executing firm	R
16.*n*.7 / 17.*n*.7	floorBroker	Member Alias	The Member Alias of the floor broker handling the trade, if the trade is handled on the floor	C

Side Trade Details				
#	Field Name	Data Type	Description	Include Key
16.*n*.8 / 17.*n*.8	cmtaFirm	Alphanumeric (8)	The OCC number of the CMTA firm (only valid for CMTA trades)	C
16.*n*.9 / 17.*n*.9	mktMkrSubAccount	Text (20)	The sub-account for the market maker, only valid when Origin Code is Market Maker	C
16.*n*.10 / 17.*n*.10	exchOriginCode	Choice	Exchange defined code designating the origin of the order, see data dictionary for list of acceptable values	R
16.*n*.11 / 17.*n*.11	liquidityCode	Choice	Specifies if this side of the trade was adding or removing liquidity. See entry for liquidityCode in the Data Dictionary for permitted values	O
16.*n*.12 / 17.*n*.12	executionCodes	Name/Value Pairs	Describes any execution codes, as described in Data Dictionary for Execution Codes. These codes would only apply only to this side of the trade	C
16.*n*.13 / 17.*n*.13	member	Member Alias	The identifier for the member firm that is responsible for the order	R
16.*n*.14 / 17.*n*.14	routedOrderID	Text (40)	For events representing an away trade, the exchange-assigned ID used to route the order away.	O

In some cases, an option trade may occur with neither a quoteID nor an orderID for one or both sides of the trade. In these cases, the quoteID/orderID can be omitted. However, the executionCodes must include NOBUYID and/or NOSELLID as appropriate.

Linkage Keys for **OT**:

- **Order Key:** date, exchange, optionID, buyDetails.orderID
- **Order Key:** date, exchange, optionID, sellDetails.orderID
- **Quote Key:** date, exchange, optionID, buyDetails.quoteID
- **Quote Key:** date, exchange, optionID, sellDetails.quoteID
- **Route Link Key:** date, symbol, exchange, buyDetails.routedOrderID
- **Route Link Key:** date, symbol, exchange, sellDetails.routedOrderID
- **Trade Key:** date, exchange, optionID, tradeID
- **Exchange/Firm Trade Key:** date, exchange, optionID, MOOTLINK, side

5.2.5.2. Stock Leg Fill Event

When a stock leg executes, it always executes at an away venue, which will report both sides of the trade. The options exchange, while possibly knowing both orders that crossed, did not actually perform the transaction. Thus, all transactions involving stock legs are reported as one-sided pass-along fills of the order, and contain the following data elements.

Table 52: Stock Leg Fill Event

Option Stock Leg Fill (OSLF)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OSLF	R
2	exchange	Exchange ID	The ID of the exchange reporting the fill to CAT	R
3	eventTimestamp	Timestamp	The date/time when the fill was processed by the exchange	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	fillID	Text (40)	An identifier for the fill, unique per reporter/trade date. This ID should uniquely identify any fill for the given exchange, date, and symbol	R
7	symbol	Symbol	The symbol of the stock being filled	R
8	quantity	Unsigned	Quantity of the fill	R
9	price	Price	Price of the fill	R
10	saleCondition	Text (8)	Conditions under which trade was executed	C
11	executionCodes	Name / Value Pairs	Adds special exchange specific codes to an execution. Zero or more codes can be entered to provide additional execution information, like where a trade may have been executed on the floor	C
12	side	Choice	The side of the executed trade: See entry for "Side" in the Data Dictionary for acceptable values	R
13	leavesQty	Unsigned	The quantity remaining unfilled after this fill event	R
14	orderID	Text (40)	The ID of the stock leg order	R
15	clearingFirm	Text (10)	The Member Alias of the clearing firm	O
16	clearingNumber	Text (20)	DTCC clearing number for this side of the trade	O

Option Stock Leg Fill (OSLF)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
17	member	Member Alias	The identifier for the member firm that is responsible for the order. This is the same member as in the complex order	R
18	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OSLF**:

- **Order Key:** date, exchange, symbol, orderID
- **Fill Key:** date, exchange, symbol, fillID

5.2.6.Post Trade Allocation Event

In the event of a modified, canceled, or replaced post trade Allocation, only the final allocation should be reported to CAT.

The fields quoteID and orderID must reference the quote/order from the original trade that is being allocated. If the trade has neither a quoteID nor an orderID, then this event will include neither IDs as well (this implies that the executionCodes field from the original trade message contains either NOBUYID or NOSELLID).

Table 53: Post Trade Allocation Event

Post Trade Allocation				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OPTA	R
2	exchange	Exchange ID	The ID of the exchange reporting the fill to CAT	R
3	eventTimestamp	Timestamp	The date/time when the allocation happened	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID of the option being traded	R
7	tradeID	Text (40)	The ID for the trade that is being reallocated. This must match a previously reported trade	R
8	orderID	Text (40)	Order ID being allocated, only applicable when the allocation is related to an order. Both orderID and quoteID cannot be provided.	C
9	quoteID	Text (40)	The ID of the quote, only applicable when the allocation is related to a market maker quote. Both orderID and quoteID cannot be provided.	C
10	quantity	Unsigned	Quantity being allocated	R
11	price	Price	Price of the allocation	R
12	side	Choice	The side of the executed trade: See entry for "Side" in the Data Dictionary for acceptable values	R
13	receivingFirm	Alphanumeric (8)	The OCC number of the receiving firm	R
14	cmtaFirm	Alphanumeric (8)	The OCC number of the CMTA firm (only valid for CMTA trades)	C
15	openCloseIndicator	Choice	The position of the order: either Open, Close, or Unspecified	O
16	exchOriginCode	Choice	Exchange defined code designating the origin of the order, see data dictionary for list of acceptable values	O
17	mktMkrSubAccount	Text (20)	The sub-account for the market maker, only meaningful if exchOriginCode rolls up to Market Maker	O

Post Trade Allocation				
#	Field Name	Data Type	Description	Include Key
18	reason	Text (255)	Free format text fields, describing why allocation was done	O
19	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OPTA**:

- **Order Key:** date, exchange, optionID, orderID
- **Quote Key:** date, exchange, optionID, quoteID
- **Trade Key:** date, exchange, optionID, tradeID

5.3. Option Order Restatement Event

Options orders that persist across business days (e.g., GTC orders) must be restated each day before any other activity is reported for that symbol. The restatement is an explicit confirmation that the order is still active in the reporter's order book, and also provides an opportunity to use per-day unique order IDs for all orders.

The attributes of the order will be restated in terms of the order's current state, after any corporate actions have been processed. Pursuant to each exchange's rule book, some corporate action types dictate that persisted orders will be canceled or converted. If converted, the order restatement field values should reflect the adjusted values on the effective date (e.g., if a 2:1 split occurred, the quantity and price would reflect the resulting change).

The following fields will not be included if restating a complex option order, but are otherwise required: openCloseIndicator, orderType, exchOriginCode, coverage, executingFirm.

Table 54: Option Order Restatement Event

Option Order Restatement (OORS)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OORS	R
2	exchange	Exchange ID	The identifier for the exchange which has received this order	R
3	eventTimestamp	Timestamp	The date/time when the order was restated	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange	R
8	originalOrderDate	Date	The most recent trading day for which the order was active. Note that this may not be the date when the order was originally accepted. If the order has been active for multiple trading days, this field must reference the previous trading day when the order was active	R
9	originalOrderID	Text (40)	The most recent internal order ID that was assigned to the order before the Restatement Event. If the orderID has not changed, then orderID and originalOrderID must be equivalent. Note this requirement is different from modification events	R
10	side	Choice	The side of the order: See entry for "Side" in the Data Dictionary for acceptable values	R
11	price	Price	The limit price of the order, if applicable. Adjusted following corporate action, if applicable	C
12	quantity	Unsigned	The order quantity, as adjusted for a corporate action, if applicable	R
13	displayQty	Unsigned	The display quantity, as adjusted for a corporate action, if applicable	R
14	displayPrice	Price	The displayed price for this order (required if displayQty is greater than zero)	C

Option Order Restatement (OORS)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
15	workingPrice	Price	The working price of the order	C
16	leavesQty	Unsigned	The quantity of the order that remains open, as adjusted for a corporate action, if applicable	C
17	openCloseIndicator	Choice	the position of the order: either Open, Close, or Unspecified	C
18	orderType	Choice	The order type is one of several possible pre-defined order types. There are a few general order type codes, and several codes unique for each exchange. See the corresponding entry in the Data Dictionary for more details about order types	C
19	timeInForce	Choice	The Time-in-Force for the order (e.g., DAY, IOC, GTC). See the Data Dictionary for a complex list of acceptable values	R
20	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details	C
21	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order, that are not necessarily handling instructions	C
22	exchOriginCode	Choice	Exchange defined code designating the origin of the order, see data dictionary for list of acceptable values	C
23	coverage	Choice	Specifies whether the order is covered or uncovered. This field may also be filled in as unspecified. See the data dictionary for a list of acceptable values	C
24	executingFirm	Alphanumeric(8)	The OCC number of the executing/give-up firm	C
25	cmtaFirm	Alphanumeric(8)	The OCC number of the CMTA firm (only valid for CMTA trades)	C
26	member	Member Alias	The identifier for the member firm that is responsible for the order	R
27	mktMkrSubAccount	Text (20)	The sub-account for the market maker, only valid when Origin Code is Market Maker	C
28	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays.	C

Option Order Restatement (OORS)				
#	Field Name	Data Type	Description	Include Key
			An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	

Linkage Keys for **OORS**:

- **Order Key:** date, exchange, optionID, orderID
- **Previous Order Key:** originalOrderDate, exchange, optionID, originalOrderID

5.4. Option Trade Break Event

When a trade is broken, an event is reported to CAT with the appropriate information. Note that CAT adds the event to the history of the order. The broken trade is not removed from the history, as it is something that actually happened and should be recorded.

Table 55: Option Trade Break Event

Option Trade Break (OTB)				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	OTB	R
2	exchange	Exchange ID	The ID for the exchange on which the trade took place	R
3	eventTimestamp	Timestamp	The date/time of the break event	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory	R

Option Trade Break (OTB)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
7	tradeDate	Date	The date on which the trade being broken occurred	R
8	tradeID	Text (40)	The ID for the trade that is being broken. This must match a previously reported trade	R
9	quantity	Unsigned	If the full quantity is being broken, then this field can be omitted. Otherwise, this represents the quantity of the original trade that is being broken	O
10	reason	Text (255)	Free format text field, with the reason for the break	O
11	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OTB**:

- **Trade Key:** date, exchange, optionID, tradeID
- **Previous Trade Key:** date, exchange, optionID, refTradeID

5.5. Option Trade Correction Event

If a trade is corrected in any way, a correction event must be reported to CAT with all details of the trade, after having been corrected. This event must capture the entire state of the trade after having been corrected.

As with trade breaks, CAT will still keep the original trade, adding the correction to the audit trail of the trade being corrected.

Table 56: Option Trade Correction Event

Option Trade Correction (OTC)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OTC	R
2	exchange	Exchange ID	The ID of the participant reporting the trade event to CAT	R
3	eventTimestamp	Timestamp	The date/time when the trade correction occurred	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	tradeID	Text (40)	An identifier for the trade being corrected	R
7	refTradeID	Text (40)	The trade being referenced. Used to link corrections if trade corrections can assign new identifiers to trades. If included, refTradeID must reference a previously reported trade, or a previously reported trade correction that has a matching tradeID	C
8	optionID	Text (40)	The ID of the option being traded	R
9	quantity	Unsigned	Quantity of the trade	R
10	price	Price	Price of the trade	R
11	saleCondition	Text (8)	Conditions under which trade was executed	C
12	executionCodes	Name / Value Pairs	Adds special exchange specific codes to an execution. Zero or more codes can be entered to provide additional execution information, like where a trade may have been executed on the floor. These codes apply to both sides of the trade	C
13	executionTimestamp	Timestamp	The date/time of the execution, applicable only when the execution time was corrected	O
14	reason	Text (255)	Free format text field, describing the reason why the correction was made	O
15	buyDetails	Side Trade Details	Information for the buy side of the trade. Format and element definitions for Buy Details are described in Table 51: Side Trade Details.	O
16	sellDetails	Side Trade Details	Information for the buy side of the trade. Format and element definitions for Sell Details are described in Table 51: Side Trade Details.	O

Option Trade Correction (OTC)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
17	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **OTC**:

- **Order Key:** date, exchange, optionID, buyDetails.orderID
- **Order Key:** date, exchange, optionID, sellDetails.orderID
- **Route Link Key:** date, optionID, exchange, buyDetails.routedOrderID
- **Route Link Key:** date, optionID, exchange, sellDetails.routedOrderID
- **Quote Key:** date, exchange, optionID, buyDetails.quoteID
- **Quote Key:** date, exchange, optionID, sellDetails.quoteID
- **Trade Key:** date, exchange, optionID, tradeID
- **Trade Key:** date, exchange, optionID, refTradeID
- **Previous Trade Key:** date, exchange, optionID, refTradeID

5.6. Option Floor Broker Events

The options floor participant event captures instances when an order routed by the matching engine to a floor participant is returned to the matching engine. The floor participant has the option to request the return of the order, or to permanently relinquish the order to the matching engine.

5.6.1.Floor Participant Event

Table 57: Floor Participant

Floor Participant (OFP)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	OFP	R
2	exchange	Exchange ID	The ID for the exchange that reported the event.	R
3	eventTimestamp	Timestamp	The date/time the matching engine was checked.	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps.	R
5	seqNumSub	Text (10)	A sequence number subsystem identifier assigned to the system that rejected the message. Required if different systems that reject messages do not share the same message sequencing process.	C
6	optionID	Text (40)	The ID previously assigned to this option in the reporter's option directory.	R
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange.	R
8	routingParty	Text (8)	The ID string used to identify the floor participant who sent this routed order.	R
9	routedOrderID	Text (40)	The ID assigned to this order when the floor participant submits the order to the exchange.	R
10	routedOriginalOrderID	Text (40)	The routedOrderID for the OFP being modified.	C
11	session	Text (40)	The name of the session used to send the order from the floor participant to the matching engine.	R
12	side	Choice	The side of the order: See entry for "side" in the Data Dictionary for acceptable values.	R
13	price	Price	Limit price for the event, which may be different than the limit price for the order. Required unless the Order Type precludes a price.	C
14	quantity	Unsigned	Quantity of the event. May be different from both the order qty and the leaves qty for the order.	R
15	displayQty	Unsigned	The displayed quantity for this event	R

Floor Participant (OFP)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
16	displayPrice	Price	Display price for the event. This must be provided on simple option orders (i.e. complexOrderID is null) when displayQty is greater than zero.	C
17	workingPrice	Price	Working Price of the event.	C
18	orderType	Choice	See the corresponding entry in the Data Dictionary for more details about order types. Required if the event has a different orderType from the order.	C
19	timeInForce	Choice	Time-in-Force for the event. See the Data Dictionary for a complete list of acceptable values.	R
20	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details. Instructions presented here should include instructions added by the Floor Participant, if any.	C
21	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order that are not necessarily handling instructions. Attributes presented here should include instructions added by the Floor Participant, if any.	C
22	member	Member Alias	The identifier for the member firm that is responsible for the order	R
23	nbbPrice	Price	NBBO at the moment just before the event is accepted by the matching engine.	R
24	nbbQty	Unsigned		O
25	nboPrice	Price		R
26	nboQty	Unsigned		O
27	complexOrderID	Text (40)	The Order ID for the parent complex order, if this event represents a leg of a complex order. This must be provided if the order represents a leg of a complex order.	C
28	complexOptionID	Text (40)	The optionID for the parent complex order, if this event represents a leg of a complex order. Not reported if the complex order's orderID is globally unique	C
29	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps	C

Floor Participant (OFP)				
#	Field Name	Data Type	Description	Include Key
			in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	

Linkage Keys for **OFP**:

- **Order Key:** date, exchange, optionID, orderID
- **Cross Order Key:** date, exchange, optionID, orderID, pairedOrderId (if populated in order attributes name value pair)
- **Route Link Key:** date, optionID, routingParty, routedOrderID, session, exchange
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderID

5.6.2.Complex Floor Participant Event

Table 58: Complex Floor Participant Event

Complex Floor Participant (OCFP)				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	OCFP	R
2	exchange	Exchange ID	The ID for the exchange.	R
3	eventTimestamp	Timestamp	The date/time the matching engine was checked.	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps.	R
5	seqNumSub	Text (10)	A sequence number subsystem identifier assigned to the system that rejected the message. Required if different systems that reject messages do not share the same message sequencing process.	C
6	optionID	Text (40)	The ID previously assigned to this option in the	R

Complex Floor Participant (OCFP)				
#	Field Name	Data Type	Description	Include Key
			reporter's option directory	
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange.	R
8	routingParty	Text (8)	The ID string used to identify the entity or individual who sent this routed order.	R
9	routedOrderID	Text (40)	The ID assigned to this order by the routing firm when submitting the order to the exchange	R
10	routedOriginalOrderID	Text (40)	The routed ID for the OCFB being modified.	C
11	session	Text (40)	The name of the session used to send the order from the floor participant to the exchange.	R
12	side	Choice	The side of the order, for a complex order the values for side can be either "AsDirected" or "Opposite", see entry for "Side" in the Data Dictionary for acceptable values.	R
13	price	Price	The net price of the order, which may be negative.	C
14	quantity	Unsigned	Quantity of the event.	R
15	timeInForce	Choice	Time-in-Force for the event. See the Data Dictionary for a complete list of acceptable values.	R
16	handlingInstructions	Name / Value Pairs	The handling instructions field contains one or more instruction codes from the pre-defined list of order handling instructions. See the documentation in the Data Dictionary for more details. Instructions presented here should include instructions added by the Floor Participant, if any.	C
17	orderAttributes	Name/Value Pairs	Defines reportable attributes of an order that are not necessarily handling instructions. Attributes presented here should include instructions added by the Floor Participant, if any.	C
18	member	Member Alias	The identifier for the member firm that is responsible for the order	R
19	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps	C

Complex Floor Participant (OCFP)				
#	Field Name	Data Type	Description	Include Key
			in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	

Linkage Keys for **OCFP**:

- **Order Key:** date, exchange, optionID, orderID
- **Cross Order Key:** date, exchange, optionID, orderID, pairedOrderId (if populated in order attributes name value pair)
- **Route Link Key:** date, optionID, routingParty, routedOrderID, session, exchange
- **Complex Order Key:** date, exchange, [complexOptionID,] complexOrderID

5.6.3.Return to Floor Participant Event

Table 59: Return to Floor Participant

Return to Floor Participant (ORFP)				
#	Field Name	Data Type	Description	Include Key
1	type	Message Type	ORFP	R
2	exchange	Exchange ID	The ID for the exchange.	R
3	eventTimestamp	Timestamp	The date/time the matching engine was checked.	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps.	R
5	seqNumSub	Text (10)	A sequence number subsystem identifier assigned to the system that rejected the message. Required if different systems that reject messages do not share the same message sequencing process.	C
6	optionID	Text (40)	The ID previously assigned to this option in the	R

Return to Floor Participant (ORFP)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			reporter's option directory	
7	orderID	Text (40)	The internal order ID assigned to the order by the exchange. If a leg is being canceled, the orderID will represent the leg order being canceled	R
8	cancelQty	Unsigned	The quantity being canceled	R
9	leavesQty	Unsigned	The quantity left open after the cancel event (zero for a full cancel)	R
10	initiator	Choice	Indicates who initiated the order cancellation: See entry for "initiator" in the Data Dictionary for acceptable values	R
11	cancelReason	Choice	Code representing the reason why the order was returned to the Floor Participant. The actual value of the code is exchange specific. See Data Dictionary for the list of allowed values	O
12	member	Member Alias	The identifier for the member firm that is responsible for the order	R
13	cycleDate	Date	Set equal to the US business date upon which the daily trading cycle of an event ends. A trading cycle may include more than one trading session. In global trading scenarios, the trading cycle of an order may span multiple dates due to the CAT requirement for representing all event timestamps in Eastern Time as well as the occurrence of US Holidays. An event that occurs on a Global Market where the Eastern Time equivalent is a non-US business date including a holiday or weekend must set the Cycle Date equal to the next US business date. And an event that occurs on a Global Market where the Eastern Time equivalent is a US business date (T) where subsequent events for that event may occur on the next Eastern Time equivalent US business date (T+1) must set the Cycle Date equal to T+1. The Cycle Date must be populated for all orders in an options series for which the trading cycle begin date is prior to the trading cycle end date. The Cycle Date must be between the Event Date and T+1, inclusive. (Where T+1 means Trade Date plus the next Trade Date.)	C

Linkage Keys for **ORFP**:

- **Order Key:** date, exchange, optionID, orderID

5.7. Lifecycle Keys

The lifecycle keys for each event are summarized in the following table. The date component of each Lifecycle Key is typically derived from the event timestamp. However, when a cycle date is provided, it will be used as the date component of the Lifecycle Key, allowing events that occur on different calendar dates, but within the same cycle date, to be properly linked.

Table 60: Section 5 Lifecycle Keys

Section	Event	Lifecycle Keys
5.1.1	Quote	**Quote Key**: date, exchange, optionID, quoteID **Previous Quote Key**: date, exchange, optionID, originalQuoteID
5.1.2	Quote Cancel	**Quote Key**: date, exchange, optionID, quoteID
5.2.1.1	Simple Option Order Accepted	**Order Key**: date, exchange, optionID, orderID **Cross Order Key**: date, exchange, orderID, pairedOrderId (if populated in order attributes name value pair) **Route Link Key**: date, optionID, routingParty, routedOrderID, session, exchange **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID
5.2.1.2	Complex Option Order Accepted	**Order Key**: date, exchange, [optionID,] orderID **Cross Order Key**: date, exchange, [optionID], orderID, pairedOrderId (if populated in order attributes name value pair) **Route Link Key**: date, optionID, routingParty, routedOrderID, session, exchange
5.2.1.3	Stock Leg Order	**Order Key**: date, exchange, symbol, orderID **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID
5.2.2.1	Option Order Modified	**Order Key**: date, exchange, optionID, orderID **Cross Order Key**: date, exchange, orderID, pairedOrderId (if populated in order attributes name value pair) **Previous Order Key**: date, exchange, optionID, originalOrderID **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID **Route Link Key**: date, optionID, routedOrderID, exchange, routingParty, session
5.2.2.2	Complex Option Order Modified	**Order Key**: date, exchange, optionID, orderID **Cross Order Key**: date, exchange, optionID, orderID, pairedOrderId (if populated in order attributes name value pair) **Previous Order Key**: date, exchange, optionID, originalOrderID **Route Link Key**: date, optionID, routedOrderID, exchange, routingParty, session
5.2.2.3	Stock Leg Modified	**Order Key**: date, exchange, symbol, orderID

Section	Event	Lifecycle Keys
		Previous Order Key: date, exchange, symbol, originalOrderID **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID
5.2.2.4	Option Order Adjusted	**Order Key**: date, exchange, optionID, orderID **Cross Order Key**: date, exchange, optionID, orderID, pairedOrderId (if populated in order attributes name value pair) **Previous Order Key**: date, exchange, optionID, originalOrderID **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID **Route Link Key**: date, optionID, routedOrderID, exchange, routingParty, session
5.2.2.5	Complex Option Order Adjusted	**Order Key**: date, exchange, optionID, orderID **Cross Order Key**: date, exchange, optionID, orderID, pairedOrderId (if populated in order attributes name value pair) **Previous Order Key**: date, exchange, optionID, originalOrderID **Route Link Key**: date, optionID, routedOrderID, exchange
5.2.2.6	Stock Leg Adjusted	**Order Key**: date, exchange, symbol, orderID **Previous Order Key**: date, exchange, symbol, originalOrderID **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID **Route Link Key**: date, optionID, routedOrderID, exchange, routingParty, session
5.2.3	Option Order Canceled	**Order Key**: date, exchange, optionID, orderID **Order Key**: date, exchange, symbol, orderID
5.2.4.2	Option Route	**Order Key**: date, exchange, optionID, orderID **Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, optionID, routingParty, routedOrderID, exchange **Route Link Key**: date, symbol, routingParty, routedOrderID, exchange **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID
5.2.4.3	Complex Option Route	**Order Key:** date, exchange, optionID, orderID **Route Link Key:** date, exchange, routingParty, routedOrderID
5.2.4.4	Internal Option Route	**Order Key**: date, exchange, optionID, orderID **Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, optionID, routingParty, routedOrderID, session, exchange **Route Link Key**: date, symbol, routingParty, routedOrderID, session, exchange **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID
5.2.4.5	Internal Complex Option Route	**Order Key**: date, exchange, optionID, orderID **Route Link Key**: date, optionID, routingParty, routedOrderID, session,

Section	Event	Lifecycle Keys
		exchange
5.2.4.6	Modify Option Route	**Order Key**: date, exchange, optionID, orderID **Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, optionID, routingParty, routedOrderID, exchange **Route Link Key**: date, symbol, routingParty, routedOrderID, exchange **Previous Route Link Key**: date, optionID, routingParty, routedOriginalOrderID, exchange **Previous Route Link Key**: date, symbol, routingParty, routedOriginalOrderID, exchange
5.2.4.7	Option Cancel Route	**Order Key**: date, exchange, optionID, orderID **Order Key**: date, exchange, symbol, orderID **Route Link Key**: date, optionID, routingParty, routedOrderID, exchange **Route Link Key**: date, symbol, routingParty, routedOrderID, exchange
5.2.5.1	Simple Option Trade	**Order Key**: date, exchange, optionID, buyDetails.orderID **Order Key**: date, exchange, optionID, sellDetails.orderID **Quote Key**: date, exchange, optionID, buyDetails.quoteID **Quote Key**: date, exchange, optionID, sellDetails.quoteID **Trade Key**: date, exchange, optionID, tradeID
5.2.5.2	Stock Leg Fill	**Order Key**: date, exchange, symbol, orderID **Fill Key**: date, exchange, symbol, fillID
5.2.6	Post Trade Allocation	**Order Key**: date, exchange, optionID, orderID **Quote Key**: date, exchange, optionID, quoteID **Trade Key**: date, exchange, optionID, tradeID
5.3	Option Order Restatement	**Order Key**: date, exchange, optionID, orderID **Previous Order Key**: originalOrderDate, exchange, optionID, originalOrderID
5.4	Option Trade Break	**Trade Key**: tradeDate, exchange, optionID, tradeID
5.5	Option Trade Correction	**Order Key**: date, exchange, optionID, buyDetails.orderID **Order Key**: date, exchange, optionID, sellDetails.orderID **Route Link Key:** date, optionID, exchange, buyDetails.routedOrderID,buyDetails **Route Link Key:** date, optionID, exchange, sellDetails.routedOrderID, **Quote Key**: date, exchange, optionID, buyDetails.quoteID **Quote Key**: date, exchange, optionID, sellDetails.quoteID **Trade Key**: date, exchange, optionID, tradeID

Section	Event	Lifecycle Keys
6.2.1	Floor Participant	**Order Key**: date, exchange, optionID, orderID **Cross Order Key**: date, exchange, optionID, orderID, pairedOrderId (if populated in order attributes name value pair) **Route Link Key**: date, optionID, routingParty, routedOrderID, session, exchange **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID
6.2.2	Complex Floor Participant	**Order Key**: date, exchange, optionID, orderID **Cross Order Key**: date, exchange, optionID, orderID, pairedOrderId (if populated in order attributes name value pair) **Route Link Key**: date, optionID, routingParty, routedOrderID, session, exchange **Complex Order Key**: date, exchange, [complexOptionID,] complexOrderID
6.2.3	Return to Floor Participant	**Order Key**: date, exchange, optionID, orderID

6. FINRA Reporting

6.1. TRF/ORF/ADF Transaction Data Event

Transactions in Eligible Securities reported to a FINRA trade reporting facility must be reported to CAT by FINRA.

Table 61: FINRA TRF/ORF ADF Transaction Data Event

FINRA TRF/ORF/ADF Transaction Data Event (TRF)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	TRF	R
2	actionType	Choice	Indicates if this is a new event, a FINRA-initiated correction, or a firm-initiated correction. **This is a pass-through value and is not used for FINRA CAT processing. Any correction event must be submitted using the standard correction process.** See Data Dictionary: `actionType`	R
3	tradeReportDate	Date	Date the trade report was received by the reporting facility.	R
4	tradeReportTimestamp	Timestamp	Date and time the trade report was received by the reporting facility.	R
5	executionDate	Date	Date the execution occurred.	R
6	executionTimestamp	Timestamp	Date and time the execution occurred.	R
7	contraReportDate	Date	Date the contra party reported the trade.	C
8	contraReportTime	Time	Time the contra party reported the trade.	O
9	contraReportTimestamp	Timestamp	Date and time the contra party reported the trade.	C
10	contraExecutionTimestamp	Timestamp	Date and time the contra party reported that the execution took place.	C
11	assumedExecutionTimestamp	Timestamp	Date and time the trade is assumed to have been executed based on available information.	R
12	acceptTime	Time	Time the trade was accepted by the contra party.	O
13	acceptTimestamp	Timestamp	Date and time the trade was accepted by the contra party.	C

FINRA TRF/ORF/ADF Transaction Data Event (TRF)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
14	declineTime	Time	Time the trade was declined by the contra party.	O
15	declineTimestamp	Timestamp	Date and time the trade was declined by the contra party.	C
16	cancellationTimestamp	Timestamp	Date and time the reporting party cancelled the trade.	C
17	lockedInTradeTimestamp	Timestamp	Date and time the locked-in trade report was received by the reporting facility.	C
18	tradeBreakTimestamp	Timestamp	Date and time the reporting party submitted their break request.	C
19	tradeBrokenTimestamp	Timestamp	Date and time the contra party submitted their break confirmation.	C
20	tradeSettlementDate	Date	Date on which the trade will settle.	C
21	issueSymbolId	Symbol	Character symbol of the traded issue.	R
22	marketCenterId	Choice	Reporting facility to which the trade was reported. See Data Dictionary: `marketCenterId`	R
23	relatedMarketCenterId	Choice	For the non-tape "riskless" leg of a riskless principal transaction, the facility or market where the first leg of the transaction was reported. See Data Dictionary: `relatedMarketCenterId`	C
24	reportedSideCode	Choice	Side of the trade (buy/sell/cross) from the perspective of the firm with the reporting obligation. See Data Dictionary: `reportedSideCode`	R
25	reportingSideMpid	Member Alias	MPID of the firm with the reporting obligation.	R
26	reportingExecutingMpid	Member Alias	MPID of the executing party.	R
27	contraSideReportingMpid	Member Alias	MPID of the contra-side firm that reported the trade.	C
28	contraExecutingMpid	Member Alias	MPID of the contra-side executing party.	C
29	reportingSideClearingNumber	Unsigned	Clearing number of the firm that cleared the trade for the reporting-side firm.	R

FINRA TRF/ORF/ADF Transaction Data Event (TRF)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
30	reportingSideBranchSequenceIdentifier	Text (20)	Branch/sequence number of the reporting-side firm.	C
31	reportingSideCapacityCode	Choice	Capacity of the reporting-side firm. See Data Dictionary: `reportingSideCapacityCode`	C
32	reportingSideShortSaleCode	Choice	Identifies a short sale by the executing firm and indicates the type of short. See Data Dictionary: `reportingSideShortSaleCode`	C
33	contraSideClearingNumber	Unsigned	Clearing number of the firm that cleared the trade for the contra-side firm.	C
34	contraSideBranchSequenceIdentifier	Text (20)	Branch/sequence number of the contra-side firm.	C
35	contraSideCapacityCode	Choice	Capacity of the contra-side firm. See Data Dictionary: `contraSideCapacityCode`	C
36	contraSideShortSaleCode	Choice	Identifies a short sale by the contra firm and indicates the type of short. See Data Dictionary: `contraSideShortSaleCode`	C
37	executionQuantity	Unsigned	Number of shares traded.	R
38	executionPrice	Price	Unit price of the trade.	R
39	reportedShareQuantity	Unsigned	Number of shares traded as reported to the SIP.	C
40	reportedUnitPrice	Price	Unit price of the trade as reported to the SIP.	C
41	clearingPrice	Price	Trade price inclusive of commissions. This information is only currently available for reported trades to the Nasdaq TRF.	C
42	publishIndicatorCode	Choice	Identifies if the trade is media reportable or not (could differ from the mediaReportedFlag for odd lot trades). See Data Dictionary: `publishIndicatorCode`	R
43	mediaReportedFlag	Choice	Identifies if the trade was media reported or not (could differ from the publishIndicatorCode for odd lot trades).	R

FINRA TRF/ORF/ADF Transaction Data Event (TRF)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			See Data Dictionary: `mediaReportedFlag`	
44	tradeStatusCode	Choice	Final status of the trade at the time it was reported. See Data Dictionary: `tradeStatusCode`	C
45	tradeSettlementModifier	Choice	Identifies a Reg NMS Settlement Type Sale Condition Code associated with a trade transaction. See Data Dictionary: `tradeSettlementModifier`	C
46	tradeThroughExemptionModifier	Choice	Further classification of the trade with regard to Trade Through Exemption. This is entered by the firm when it reports the trade. See Data Dictionary: `tradeThroughExemptionModifier`	C
47	tradeReportingModifier	Choice	Further classification of the trade with regard to Extended Hours/Sequence. This can either be entered by the firm or appended by the system. See Data Dictionary: `tradeReportingModifier`	C
48	sroRequiredModifier	Choice	Further classification of the trade with regard to SRO required detail. This can either be entered by the firm or appended by the system. See Data Dictionary: `sroRequiredModifier`	C
49	systemAppendedTradeReportingModifierFlag	Choice	Identifies if the Trade Reporting Modifier Code was entered by the reporting firm or appended by the reporting facility. See Data Dictionary: `systemAppendedTradeReportingModifierFlag`	R
50	originalModifierCode	Text (4)	Four-byte trade modifier as entered by the firm.	C
51	reversalFlag	Choice	Indicates that the trade report is reversal transaction. See Data Dictionary: `reversalFlag`	R
52	carryoverFlag	Choice	Indicates that the trade transaction was carried over (not accepted/declined by the contra firm on T+0) for processing. See Data Dictionary: `carryoverFlag`	C
53	tradeThroughExemptFlag	Choice	Indicates that the trade is trade through exempt.	C

FINRA TRF/ORF/ADF Transaction Data Event (TRF)				
#	Field Name	Data Type	Description	Include Key
			See Data Dictionary: `tradeThroughExemptFlag`	
54	contraEntryFlag	Choice	Indicates that the contra party is the only side that reported the trade. See Data Dictionary: `contraEntryFlag`	C
55	explicitFeeFlag	Choice	Indicates if a Clearing Price was entered. See Data Dictionary: `explicitFeeFlag`	C
56	clearingFlag	Choice	Clearing and matching specifications of the trade transaction. See Data Dictionary: `clearingFlag`	R
57	specialTradeCode	Choice	Identifies special and step-out trades. See Data Dictionary: `specialTradeCode`	C
58	supervisoryEntryCode	Choice	Indicates if a Market Operations Supervisor entered the trade message on behalf of the reporting side of the trade transaction. See Data Dictionary: `supervisoryEntryCode`	C
59	controlNumber	Text (30)	Unique identifier for the reporting side of each trade transaction.	R
60	reportingSideMemoText	Text (30)	Provides a link (via Control Number) to the original trade report, when a subsequent report is submitted to reallocate some of the trade volume to a different capacity. This is a free-form text field; participants can enter any information in this field.	C
61	tradeSourceCode	Choice	Trade Sources. See Data Dictionary: `tradeSourceCode`	R
62	contraControlNumber	Text (30)	Control Number for the contra party.	C
63	OEMemoTx	Text (10)	Memo text entered by firm.	C
64	reportTypeCode	Choice	Identifies whether this is a No/Was report. See Data Dictionary: `reportTypeCode`	C
65	noWasLinkNumber	Text (30)	Link to first No transaction.	C
66	intendedMarketCenter	Choice	Intended Market Center. See Data Dictionary: `intendedMarketCenter`	C

FINRA TRF/ORF/ADF Transaction Data Event (TRF)				
#	Field Name	Data Type	Description	Include Key
67	tradeReferenceNumber	Text (20)	Trade Reference Number	C
68	priceOverrideCode	Choice	Identifies if a price validation test was overridden when the trade was entered into ACT. (When trades are entered into ACT, they are validated for reasonableness against a Price Validation Table. The Price Override widens the validation range). See Data Dictionary: `priceOverrideCode`	C
69	asOfFlag	Choice	Indicates as-of trade. See Data Dictionary: `asOfFlag`	R
70	lastUpdateDate	Date	Date the record was last updated.	R
71	lastUpdateTime	Timestamp	Date and time the record was last updated.	C
72	lockedInFlag	Choice	Locked-in flag. See Data Dictionary: `lockedInFlag`	C
73	noLinkControlNumber	Text (30)	Provides a link (via Control Number) to previous No transaction.	C
74	firmTradeModifierSettlementTypeCode	Choice	User Trade Modifier - Settlement Type (Settlement modifiers). See Data Dictionary: `firmTradeModifierSettlementTypeCode`	C
75	firmTradeModifierThroughExemptCode	Choice	Further classification of the trade with regard to Trade Through Exemption. This is entered by the firm when it reports the trade. See Data Dictionary: `firmTradeModifierThroughExemptCode`	C
76	firmTradeModifierLateCode	Choice	System Trade Modifier - Time Modifiers (TradeModifier 3 in the FIX Spec). See Data Dictionary: `firmTradeModifierLateCode`	C
77	finraTradeModifierSroCode	Choice	System Trade Modifier SRO - Updated by MPP System. See Data Dictionary: `finraTradeModifierSroCode`	C
78	trfTradeModifierSroCode	Choice	User Trade Modifier - SRO - Updated by TRF. SRO detail sale condition. Required indicator if a trade falls under one of the following	C

FINRA TRF/ORF/ADF Transaction Data Event (TRF)				
#	Field Name	Data Type	Description	Include Key
			transaction types (otherwise the field must not be set). See Data Dictionary: `trfTradeModifierSroCode`	
79	trfTradeModifierLateCode	Choice	System Trade Modifier - Time Modifiers - Updated by TRF. See Data Dictionary: `trfTradeModifierLateCode`	C
80	finraTradeModifierLateCode	Choice	System Trade Modifier - Time Modifier - Updated by MPP Engine. See Data Dictionary: `finraTradeModifierLateCode`	C
81	reportingObligationFlag	Choice	Identifies if the reporting-side firm had the reporting obligation for the trade under FINRA trade reporting rules. See Data Dictionary: `reportingObligationFlag`	C
82	tradeCorrectionClassCode	Choice	Trade Correction Classification. See Data Dictionary: `tradeCorrectionClassCode`	C
83	contraReportingObligationFlag	Choice	Identifies if the contra-side firm had the reporting obligation for the trade under FINRA trade reporting rules. See Data Dictionary: `contraReportingObligationFlag`	C
84	finraContraControlDate	Date	Control Date corresponding to FINRA Contra Control Number.	C
85	finraContraControlNumber	Text (30)	Control Number used for interaction between TRFs and FINRA; populated only when trade is matched by comparison. Will be unique for a trade report date and market center.	C
86	finraControlDate	Date	Control Date of the current version of the trade.	R
87	finraControlNumber	Text (30)	Control Number of the current version of the trade.	R
88	firstTradeFinraControlDate	Date	Control Date of the first version of the trade.	R
89	firstTradeFinraControlNumber	Text (30)	Control Number of the first version of the trade.	R

FINRA TRF/ORF/ADF Transaction Data Event (TRF)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
90	previousTradeFinraControlDate	Date	FINRA Control Date of the previous version of the trade.	C
91	previousTradeFinraControlNumber	Text (30)	FINRA Control Number of the previous version of the trade.	C
92	positionTransferFlag	Choice	Special processing flag indicating that the transaction is for internal FINRA use only and should not be disseminated. See Data Dictionary: `positionTransferFlag`	C
93	trfContraControlNumber	Text (30)	Control Number used for interaction between TRFs and Firms; populated only when trade is matched by comparison. May not be unique for a given day.	C
94	trfControlNumber	Text (30)	Control Number used for interaction between Firms and TRFs. May not be unique for a given day.	C
95	referenceNumber	Text (20)	User-defined trade reference number.	C
96	firmTradeModifierSroCode	Choice	Further classification of the trade with regard to SRO required detail. This can either be entered by the firm or appended by the system. See Data Dictionary: `firmTradeModifierSroCode`	C
97	finraTradeModifierThroughExemptTime	Time	System Trade Thru Exempt Modifier Date and Time.	C
98	tradeModifierThroughExemptTime	Time	User Trade Thru Exempt Modifier Time.	O
99	tradeModifierSroTime	Time	Time associated with Prior Reference Price or Stopped Stock trade.	O
100	referenceReportingFacility	Text (6)	Reference Reporting Facility.	C
101	trfProcessingDate	Date	Date FINRA received the record from the reporting facility.	R
102	recordUniqueIdentifier	Text (31)	FINRA-assigned unique identifier for each Reported Trade record.	R
103	recordLoadDate	Date	Date the record was created.	R
104	firstTradeFinraContraControlDate	Date	Control Date of the first trade in a chain of corrections on the contra side trade report.	C

FINRA TRF/ORF/ADF Transaction Data Event (TRF)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
105	firstTradeFinraContraControlNumber	Text (30)	Control Number of the first trade in a chain of corrections on the contra side trade report.	C
106	previousTradeFinraContraControlDate	Date	Control Date of the previous trade in a chain of corrections on the contra side trade report.	C
107	previousTradeFinraContraControlNumber	Text (30)	Control Number of the previous trade in a chain of corrections on the contra side trade report.	C
108	firmOriginalTrfControlNumber	Text (30)	Original Control Number provided by the TRF to the firm.	C
109	reportingSubmittingEntityId	Text (4)	Indicates the entity that initiated the submission. For a FINRA-initiated submission on behalf of the firm, this will be 'FNRA'. Otherwise, for a firm-initiated submission, it will be the firm MPID. For NC TRF, NQ TRF and NY TRF, this is always NQTC, NQTR or NYTR. For ADF and ORF it is the MPID of the submitting firm.	R
110	contraSubmittingEntityId	Text (4)	Indicates the entity that initiated the submission. For a FINRA-initiated submission on behalf of the firm, this will be 'FNRA'. Otherwise, for a firm-initiated submission, it will be the firm MPID. For NC TRF, NQ TRF and NY TRF, this is always NQTC, NQTR or NYTR. For ADF and ORF it is the MPID of the submitting firm.	C

6.2. OTC Halt/Resume Data

FINRA will report OTC Halt/Resume data to FINRA CAT with the following fields:

Table 62: FINRA Halt/Resume

<table>
<tr><th></th><th colspan="4">FINRA Halt/Resume (FHR)</th></tr>
<tr><th>#</th><th>Field Name</th><th>Data Type</th><th>Description</th><th>Include Key</th></tr>
<tr><td>1</td><td>type</td><td>Message Type</td><td>FHR</td><td>R</td></tr>
<tr><td>2</td><td>tradeDate</td><td>Date</td><td>Date on which message was disseminated; derived from the date portion of the <code>messageTimestamp</code>.</td><td>R</td></tr>
<tr><td>3</td><td>1 messageCategory</td><td>Choice</td><td>This field, along with the <code>haltMessageType</code>, identifies the message format.
See Data Dictionary: <code>messageCategory</code></td><td>R</td></tr>
<tr><td>4</td><td>haltMessageType</td><td>Choice</td><td>This field, along with the <code>messageCategory</code>, identifies the message format.
See Data Dictionary: <code>haltMessageType</code></td><td>R</td></tr>
<tr><td>5</td><td>sessionIdentifier</td><td>Choice</td><td>Indicates the market session of the message.
See Data Dictionary: <code>sessionIdentifier</code></td><td>R</td></tr>
<tr><td>6</td><td>retransmissionRequester</td><td>Text (2)</td><td>Indicates if the message is an original transmission or retransmission. If the message is a retransmission, this field indicates the two-character retransmission identifier of the intended data recipient. Values may include:
O (space) An original transmission to all recipients
R (space) A retransmission to all recipients
T (space) A test cycle transmission to all recipients
Specific Vendor ID Two-character value to be assigned on vendor-by-vendor basis. Contact FINRA for additional information.
Note: Because the value could be any two-character value, this field will not be validated against a list of allowable values. Additionally, FINRA CAT will strip all leading and trailing spaces when storing the input data.</td><td>R</td></tr>
<tr><td>7</td><td>messageSequenceNumber</td><td>Unsigned</td><td>At the beginning of each operational cycle, this number will be set to ‘00000000’ (for the Start of Day) for each data channel.</td><td>R</td></tr>
<tr><td>8</td><td>marketCenterOriginatorID</td><td>Choice</td><td>Market center or system that originated the action.</td><td>R</td></tr>
</table>

	FINRA Halt/Resume (FHR)			
#	**Field Name**	**Data Type**	**Description**	**Include Key**
			See Data Dictionary: `marketCenterOriginatorID`	
9	messageTimestamp	Timestamp	The date and time of the action (Halt, Quote Resume or Trade Resume). When the event is for a Halt, this will be the same as the haltActionTimestamp.	R
10	symbol	Symbol	Symbol of the issue being halted/resumed.	C
11	issueID	Integer	FINRA-assigned issue ID of the issue being halted/resumed.	C
12	haltActionCode	Choice	The type of action (i.e. halt, quote resume, trade resume). See Data Dictionary: `haltActionCode`	R
13	haltActionTimestamp	Timestamp	The date/time the halt was initiated.	R
14	haltReasonCode	Choice	The reason the security is being halted/resumed. See Data Dictionary: `haltReasonCode`	C

6.3. Equity Best Bid and Offer Event

When an SRO display-only facility accepts a routed quote, an Equity Best Bid and Offer event is reported to CAT. If the quote is rejected, then the event is not reported to CAT.

Table 63: Equity Best Bid and Offer Event

Equity Best Bid and Offer (EBBO)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
1	type	Message Type	EBBO	R
2	marketCenterId	Exchange ID	Display-Only Facility on which the quote was displayed. See Data Dictionary: `marketCenterId`	R
3	eventTimestamp	Timestamp	The date/time of quote receipt	R
4	sequenceNumber	Unsigned	The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps	C

Equity Best Bid and Offer (EBBO)				
#	**Field Name**	**Data Type**	**Description**	**Include Key**
5	seqNumSub	Text (10)	A sequence number subsystem identifier	C
6	symbol	Symbol	The stock symbol, in either the symbology of the listing exchange or a valid alias	R
7	routingParty	Text (8)	The ID string used to identify the entity that routed the quote to the display-only facility	R
8	routedOrderID	Text (40)	The quote ID that the firm used in the API message when they sent the quote to the display only facility (e.g., in FIX it would be Tag 117, quoteID)	R
9	session	Text (40)	The ID assigned to the specific session that the routing member used to route the order to the exchange	R
10	lockedCrossedOverrideFlag	Boolean	Identifies whether a quote should be considered valid even if it will lock or cross the market. 'True' indicates that the quote is still valid 'False' indicates that the quote is not valid if it locks or crosses. If no value is provided, it's assumed to be false. Value provided by the submitting firm should be passed through.	C
11	bidPrice	Price	Price for the bid. Must be provided when the bidQty is provided. If the bidPrice is not provided, then the bidQty must not be provided. NOTE: in the absence of a bidPrice the most recently reported bidPrice remains in effect.	C
12	bidQty	Unsigned	Quantity of the bid in shares. Must be provided when the bidPrice is provided. If the bidQty is not provided, then the bidPrice must not be provided. NOTE: in the absence of a bidQty the most recently reported bidQty remains in effect.	C
13	askPrice	Price	Price for the ask. Must be provided when the askQty is provided. If the askPrice is not provided, then the askQty must not be provided. NOTE: in the absence of a askPrice the most recently reported askPrice remains in effect.	C

Equity Best Bid and Offer (EBBO)				
#	Field Name	Data Type	Description	Include Key
14	askQty	Unsigned	Quantity of the ask in shares. Must be provided when the askPrice is provided. If the askQty is not provided, then the askPrice must not be provided. NOTE: in the absence of a askQty the most recently reported askQty remains in effect.	C
15	quoteCondition	Text(8)	Indicator used to determine whether a quote is eligible to participate in the NBBO. Value provided by the submitting firm should be passed through.	C
16	quoteInstructions	Name/Value Pairs	Defines any additional instructions or attributes for the quote, as described in the Data Dictionary	C
17	quoteID	Text (40)	The internal order ID assigned to the quote by the Participant. If no internal identifier is assigned, then: • The routedOrderID value can be repeated, provided it is unique by date, symbol and Participant. • If no unique internal identifier is created, then this field can be left empty.	C

Lifecycle keys for this event:

- **Route Link Key:** date, *symbol*, *routingParty*, *routedOrderID*, *session*, *exchange*

Currently the only SRO display-only facility is the FINRA ADF, which does not assign or maintain an quote identifier. Since top of book quotations cannot be canceled, cancel/replaced or executed, no quote link key will be assigned. Each quotation can be tracked by its route link key.

7. Stock Exchange Event Examples

7.1. Order Accepted Event Example

This section will illustrate examples for an order accepted event, an order modified event, and an order canceled event using the following scenario: A new order is routed to the exchange, accepted by the exchange, updated by the firm that sent the order, and is finally canceled by the exchange.

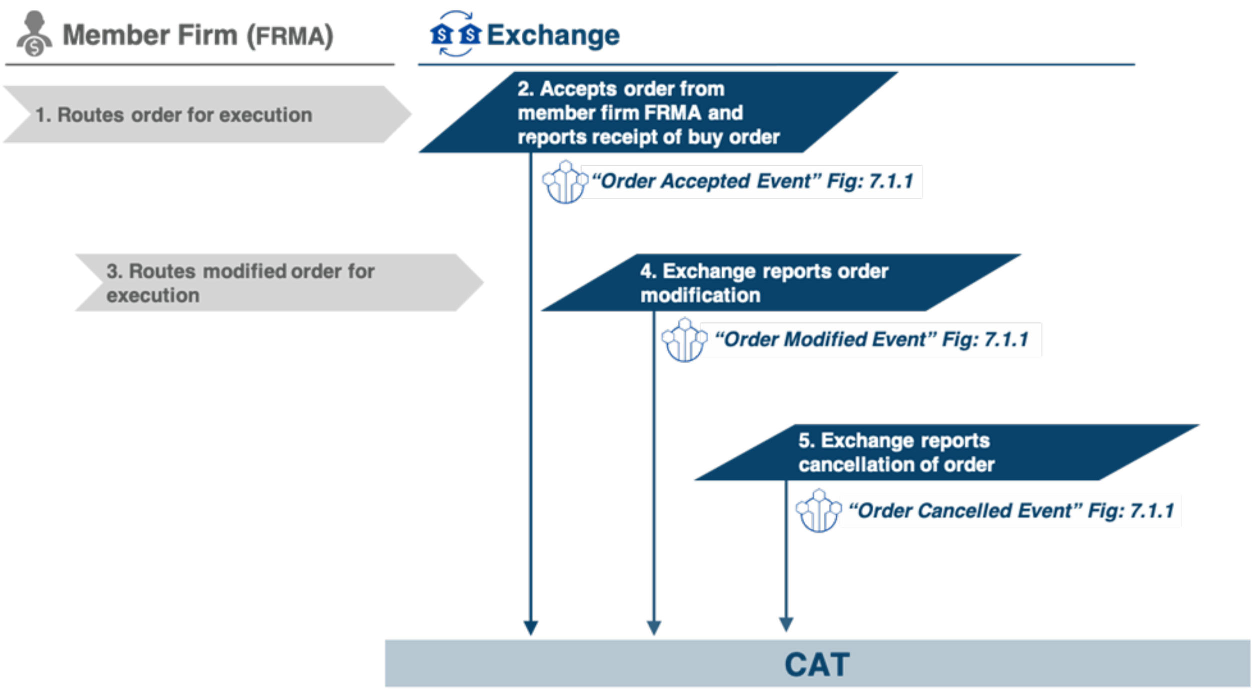

Figure 4: Order Event Lifecycle Example

Table 64: Order Event Lifecycle Example

#	Step	Reported Event	Comments
1	Member Firm Routes order for Execution	NA	• A member firm routes an order to Exchange "Exch1" over session ID 7 with the order ID of 2156. This order is a buy order for the symbol ABCD, with a quantity of 300
2	Exchange accepts the order and reports an order accepted event to CAT	***Order Accepted Event:*** type: EOA	• The exchange accepts the buy order and assigns it the internal order ID: 98765.

#	Step	Reported Event	Comments
		exchange: Exch1 eventTimestamp: 20170307T103242.123456789 sequenceNumber: 11133 symbol: ABCD orderID: 98765 routingParty: FRMA routedOrderID: 2156 session: 7 side: Buy price: 157.00 quantity: 300 displayQty: 300 displayPrice: 157.00 workingPrice: 157.00 orderType: LMT timeInForce: GTT capacity: Principal handlingInstructions: XTIME=20170315T123456.123456789 nbbPrice: 157.00 nbbQty: 100 nboPrice: 157.25 nboQty: 100 member: Mem01	• The ID that was used by the member firm is included as the Routed Order ID because Time in Force = GTC, the order expires at a particular time: requires XTIME • In handling instructions to provide the order's expire time. The NBBO is as the exchange saw it just before accepting the order. Note that after accepting the order, the aggregate NBB quantity would go up by 300 to account for this order, which is at the NBB price.
3	Member routes a modification of the order to the exchange	NA	• The member firm modifies their existing order, increasing the price to 157.01
4	Exchange modifies order	***Order Modified Event:*** type: EOM exchange: Exch1 eventTimestamp: 20170307T103350.123456789 sequenceNumber: 11140 symbol: ABCD orderID: 99564 originalOrderID: 98765 initiator: Firm nbbPrice: 157.00 nbbQty: 400 nboPrice: 157.25 nboQty: 100 price: 157.01 displayPrice: 157.01 workingPrice: 157.01 side: Buy quantity: 300 displayQty: 300 leavesQty: 300	• The exchange reports a firm-initiated modification to the order described in the previous section. In this case, the price of the order is increased to 157.01. • Some exchanges assign a new internal order ID after an update, in this case The new internal order ID is 99564

#	Step	Reported Event	Comments
		orderType: LMT timeInForce: GTT capacity: Principal handlingInstructions: XTIME=20170315T123456.123456789 member: Mem01	
5	Exchange cancels the order	***Order Canceled Event:*** type: EOC exchange: Exch1 eventTimestamp: 20170307T103552.000001089 sequenceNumber: 11453 symbol: ABCD orderID: 99564 cancelQty: 300 leavesQty: 0 initiator: Exchange member: Mem01	• The order has passed its expiration time and is canceled by the exchange • Initiator value = exchange given that the XTIME has passed

7.1.1.JSON Examples

Order Accepted Event


```
{
  "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T103242.123456789",
  "sequenceNumber": 11133,
  "symbol": "ABCD",
  "orderID": "98765",
  "routingParty": "FRMA",
  "routedOrderID": "2156",
  "session": "7",
  "side": "Buy",
  "price": 157.00,
  "quantity": 300,
  "displayQty": 300,
  "displayPrice": 157.00,
  "workingPrice": 157.00,
  "orderType": "LMT",
  "timeInForce": "GTT",
  "capacity": "Principal",
  "handlingInstructions": "XTIME=20170315T123456.123456789",
  "nbbPrice": 157.00,
  "nbbQty": 100,
  "nboPrice": 157.25,
  "nboQty": 100,
  "member": "Mem01"
}
```

Order Modified Event


```
{
  "type": "EOM",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T103350.123456789",
  "sequenceNumber": 11140,
  "symbol": "ABCD",
  "orderID": "99564",
  "originalOrderID": "98765",
  "side": "Buy",
  "quantity": 300,
  "displayQty": 300,
  "orderType": "LMT",
  "timeInForce": "GTT",
  "handlingInstructions": "XTIME=20170315T123456.123456789",
  "initiator": "Firm",
  "price": 157.01,
  "displayPrice": 157.01,
  "workingPrice": 157.01,
  "leavesQty": 300,
  "capacity": "Principal",
  "nbbPrice": 157.00,
  "nbbQty": 400,
  "nboPrice": 157.25,
  "nboQty": 100,
  "member": "Mem01"
}
```


Order Canceled Event


```
{
  "type": "EOC",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T103552.000001089",
  "sequenceNumber": 11453,
  "symbol": "ABCD",
  "orderID": "99564",
  "cancelQty": 300,
  "leavesQty": 0,
  "initiator": "Exchange",
  "member": "Mem01"
}
```

7.2. Order Trade Event Example

This section will demonstrate a trade event example that occurs after a buy and sell order are matched. In this case, a sell order is accepted for a price of 157.20 and quantity of 100. A buy order is then accepted for a price of 157.20 and quantity of 100. The two orders are matched and a trade event is reported.

In this scenario, the exchange is required to report the following events to CAT:

1. Order Accepted Events from each of the orders; and
2. Order Trade Event

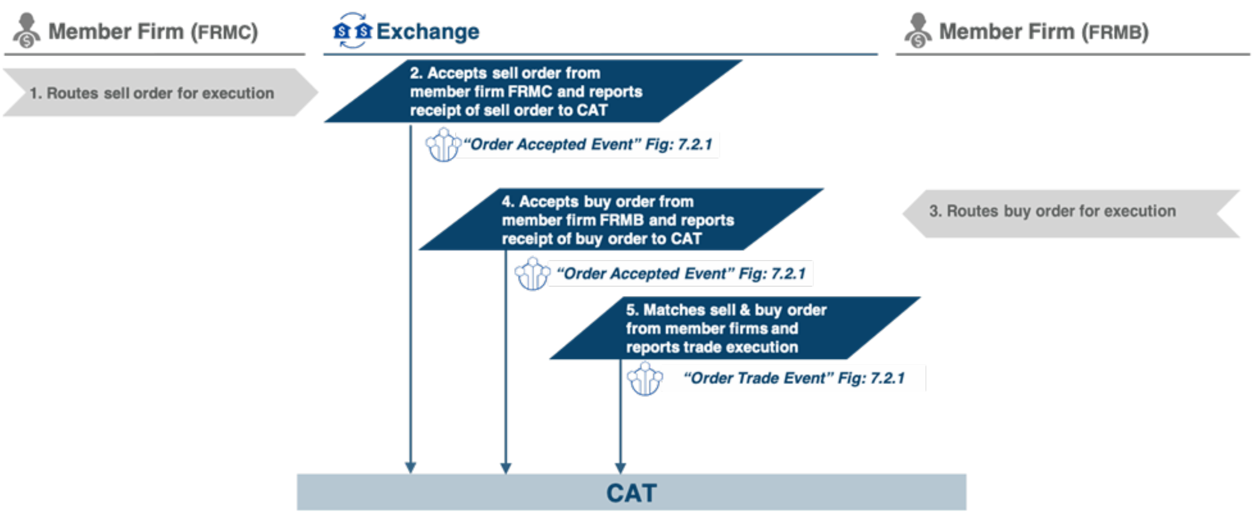

Figure 5: Order Trade Event Example

Table 65: Trade Event Example

#	Step	Reported Event	Comments
1	Member Firm FRMC Routes sell order for execution	NA	• A member firm routes a sell order to Exchange "Exch1" over session ID FRMC:123 with the order ID of 2156. This order is a sell order for the symbol ABCD, with a quantity of 100
2	Exchange accepts the sell order and reports an order accepted event to CAT	***Order Accepted Event:*** type: EOA exchange: Exch1 eventTimestamp: 20170307T134000.123456	• The exchange accepts the sell order and assigns it the internal order ID: 10999. The order type is a limit order with time in force = day.

#	Step	Reported Event	Comments
		sequenceNumber: 12345 symbol: ABCD orderID: 10999 routingParty: FRMC routedOrderID: 2156 session: FRMC:123 side: Sell price: 157.20 quantity: 100 displayQty: 100 displayPrice: 157.20 workingPrice: 157.20 orderType: LMT timeInForce: DAY capacity: Agency nbbPrice: 157.00 nbbQty: 100 nboPrice: 157.25 nboQty: 100 member: Mem01	• The ID that was used by the member firm is included as the Routed Order ID • The NBBO is as the exchange saw it just before accepting the order. Note that after accepting the order, the national best offer would change to account for this order, which is below the national best offer.
3	Member Firm FRMB Routes buy order for execution	NA	• A member firm FRMB routes a buy order to Exchange "Exch1" over session ID 7 with the order ID of 9150. This order is a buy order for the symbol ABCD, with a quantity of 100
4	Exchange accepts the buy order and reports an order accepted event to CAT	***Order Accepted Event:*** type: EOA exchange: Exch1 eventTimestamp: 20170307T134001.123456 sequenceNumber: 19190 symbol: ABCD orderID: 20263 routingParty: FRMB routedOrderID: 9150 session: 7 side: Buy price: 157.20 quantity: 100 displayQty: 0 workingPrice: 157.20 orderType: LMT timeInForce: DAY capacity: Principal nbbPrice: 157.00 nbbQty: 100 nboPrice: 157.20 nboQty: 100	• The exchange accepts the buy order and assigns it the internal order ID: 20263. The order type is a limit order with time in force = day. • The ID that was used by the member firm is included as the Routed Order ID • The NBBO is as the exchange saw it just before accepting the order.

#	Step	Reported Event	Comments
		member: Mem02	
5	Exchange matches buy and sell order and the trade is executed	***Order Trade Event:*** type: EOT exchange: Exch1 eventTimestamp: 20170307T134001.125456 sequenceNumber: 19191 symbol: ABCD tradeID: 19900422 quantity: 100 price: 157.20 saleCondition: E@ nbbPrice: 157.00 nbbQuantity: 100 nboPrice: 157.20 nboQuantity: 100 buyDetails side: Buy leavesQty: 0 orderID: 20263 clearingNumber: 5656 capacity: Principal liquidityCode: Removed member: Mem02 sellDetails side: Sell leavesQty: 0 orderID: 10999 clearingNumber: 7878 capacity: Agency liquidityCode: Added member: Mem01	• The buy and sell orders from the previous steps cross and the exchange initiates the trade, reporting an order trade event to CAT.

7.2.1.JSON Examples

Order Accepted Event: Sell


```
{
  "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T134000.123456",
  "sequenceNumber": 12345,
  "symbol": "ABCD",
  "orderID": "10999",
  "routingParty": "FRMC",
  "routedOrderID": "2156",
  "session": "FRMC:123",
```

```
  "side": "Sell",
  "price": 157.20,
  "quantity": 100,
  "displayQty": 100,
  "displayPrice": 157.20,
  "workingPrice": 157.20,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "capacity": "Agency",
  "nbbPrice": 157.00,
  "nbbQty": 100,
  "nboPrice": 157.25,
  "nboQty": 100,
  "member": "Mem01"
}
```


Order Accepted Event: Buy


```
{
  "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T134001.123456",
  "sequenceNumber": 19190,
  "symbol": "ABCD",
  "orderID": "20263",
  "routingParty": "FRMB",
  "routedOrderID": "9150",
  "session": "7",
  "side": "Buy",
  "price": 157.20,
  "quantity": 100,
  "displayQty": 0,
  "workingPrice": 157.20,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "nbbPrice": 157.00,
  "nbbQty": 100,
  "nboPrice": 157.20,
  "nboQty": 100,
  "member": "Mem02"
}
```


Order Trade Event


```
{
  "type": "EOT",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T134001.125456",
  "sequenceNumber": 19191,
  "symbol": "ABCD",
  "tradeID": "19900422",
  "quantity": 100",
  "price": 157.20,
  "saleCondition": "E@",
```

```
  "nbbPrice": 157.00,
  "nbbQty": 100,
  "nboPrice": 157.20,
  "nboQty": 100,
  "buyDetails": {
    "side": "Buy",
    "leavesQty": 0,
    "orderID": "20263",
    "clearingNumber": "5656"
    "capacity": "Principal",
    "liquidityCode": "Removed",
    "member": "Mem02"
  },
  "sellDetails": {
    "side": "Sell",
    "leavesQty": 0,
    "orderID": "10999",
    "clearingNumber": "7878"
    "capacity": "Agency",
    "liquidityCode": "Added",
    "member": "Mem01"
  }
}
```

7.3. Order Route and Order Fill Event Example

This scenario illustrates the reporting requirements to CAT when an exchange routes an order to a routing broker-dealer for execution on an away exchange, and Exchange 1's subsequent reporting obligation on fills of the routed order.

In this scenario Exchange 1 receives and reports acceptance of an order, then routes the order to their routing broker dealer for execution on an away exchange. When an execution occurs on the away exchange, the routing broker reports the fill back to Exchange 1. The following events are reported:

1. Order Accepted Event of the original order,
2. The Order Route Event, and

3. The Order Fill Event.

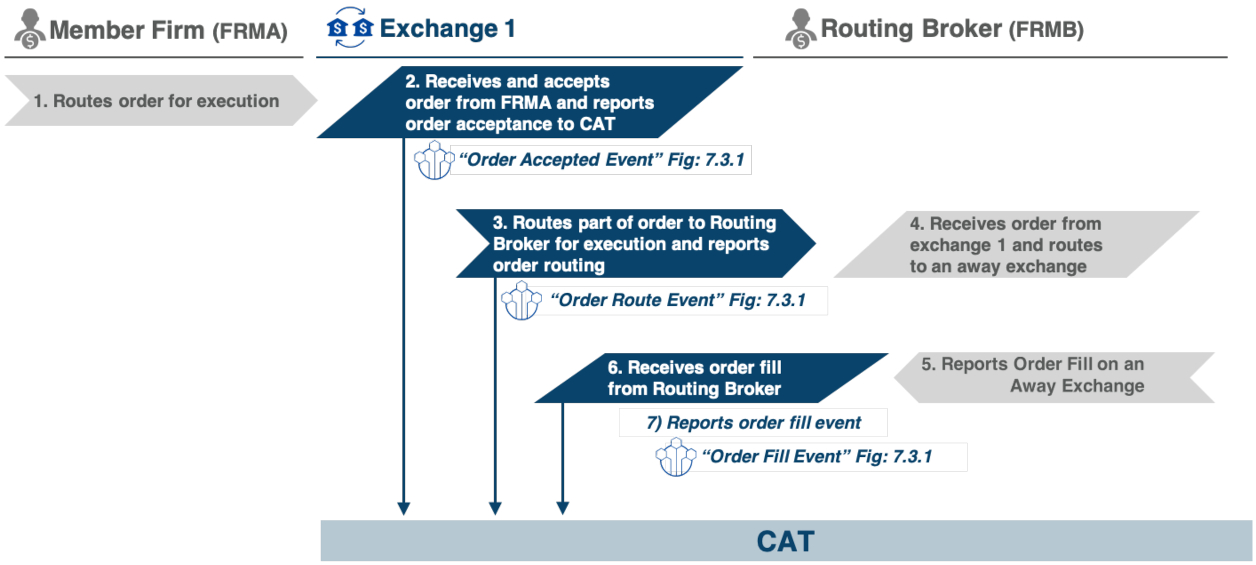

Figure 6: Order Route and Order Fill Event Example

Table 66: Order Route and Order Fill Event Example

#	Step	Reported Event	Comments
1	Member Firm FRMA Routes buy order for execution	NA	A member firm routes a buy order to Exchange "Exch1" over session ID 3 with the order ID of 567890. This order is a buy order for the symbol ABCD, with a quantity of 200 at the price of 157.25
2	Exchange accepts the buy order and reports an order accepted event to CAT	***Order Accepted Event:*** type: EOA exchange: Exch1 eventTimestamp 20170307T144010.123456789 sequenceNumber: 12345 symbol: ABCD orderID: 10001 routingParty: FRMA routedOrderID: 567890 session: 3 side: Buy price: 157.25 quantity: 200 displayQty: 100 displayPrice: 157.25	• The exchange accepts the buy order and assigns it the internal order ID: 10001. The order type is a limit order with time in force = day. • The ID that was used by the member firm is included as the Routed Order ID The NBBO is as the exchange saw it just before accepting the order.

#	Step	Reported Event	Comments
		workingPrice: 157.25 orderType: LMT timeInForce: DAY capacity: Principal nbbPrice: 157.00 nbbQty: 100 nboPrice: 157.25 nboQty: 100 member: Mem01	
3	Exch1 routes part of the order quantity to its routing broker for execution on an away exchange	**Route Order Event** type: EOR exchange: Exch1 eventTimestamp: 20170307T144010.123457789 sequenceNumber: 12346 symbol: ABCD orderID: 10001 routingParty: FRMB routedOrderID: E123456 session: 5 side: Buy price: 157.25 quantity: 100 displayQty: 0 orderType: LMT timeInForce: IOC capacity: Agency handlingInstructions: ISO|R2E=Exch2 result: ACK resultTimestamp: 20170307T144010.124457789 nbbPrice: 157.00 nbbQty: 100 nboPrice: 157.25 nboQty: 100 member: Mem01	• One hundred of the two hundred shares of the order in the previous step are routed to the exchange's routing broker FRMB for execution on an away exchange in order to meet the order protection rule • Routing Firm = FRMB • The Routed Order ID is the new order ID assigned by exchange A and sent to routing firm • Display quantity = 0, this is a non-displayed order • Time in force = IOC, hit the quote or cancel Handling instructions = ISO, inter-market sweep, routed to exchange Exch2
4	Routing broker routes the order to an away exchange		
5	Away exchange fills the order and sends a fill report back to the routing broker		
6	Routing broker receives order fill from away broker and reports order fill on an away exchange to Exch1		
7	Exch1 reports an order fill event	**Order Fill Event** type: EOF exchange: Exch1	• The exchange reports the fill to the member firm that placed the order, and arranges for clearing to flip the shares. The actual trade

#	Step	Reported Event	Comments
		eventTimestamp: 20170307T144010.129456789 sequenceNumber: 15501 fillID: 192834 symbol: ABCD price: 157.25 saleCondition: E@ side: Buy quantity: 100 leavesQty: 100 orderID: 10001 clearingNumber: 9898 contraClearingNumber: 9899 routingParty: FRMB routedOrderID: E123456 session: 3 capacity: Principal member: Mem01	took place on the away exchange, and the transaction between the two firms is handled in clearing.

7.3.1.JSON Examples

Order Accepted Event


```
{
  "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T144010.123456789",
  "sequenceNumber": 12345,
  "symbol": "ABCD",
  "orderID": "10001",
  "routingParty": "FRMA",
  "routedOrderID": "567890",
  "session": "3",
  "side": "Buy",
  "price": 157.25,
  "quantity": 200,
  "displayQty": 100,
  "displayPrice": 157.25,
  "workingPrice": 157.25,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "nbbPrice": 157.00,
  "nbbQty": 100,
  "nboPrice": 157.25,
  "nboQty": 100,
  "member": "Mem01"
}
```

Order Route Event


```
{
  "type": "EOR",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T144010.123457789",
  "sequenceNumber": 12346,
  "symbol": "ABCD",
  "orderID": "10001",
  "routingParty": "FRMB",
  "routedOrderID": "E123456",
  "session": "5",
  "side": "Buy",
  "price": 157.25,
  "quantity": 100,
  "displayQty": 0,
  "orderType": "LMT",
  "timeInForce": "IOC",
  "capacity": "Agency",
  "handlingInstructions": "ISO|R2E=Exch2",
  "result": "ACK",
  "resultTimestamp": "20170307T144010.124457789",
  "nbbPrice": 157.00,
  "nbbQty": 100,
  "nboPrice": 157.25,
  "nboQty": 100,
  "member": "Mem01"
}
```


Order Fill Event


```
{
  "type": "EOF",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T144010.129456789",
  "sequenceNumber": 15501,
  "fillID": "192834",
  "symbol": "ABCD",
  "price": 157.25,
  "side": "Buy",
  "saleCondition": "E@",
  "quantity": 100,
  "leavesQty": 100,
  "orderID": 10001,
  "clearingNumber": "9898",
  "contraClearingNumber":"9899"
  "routingParty": "FRMB",
  "routedOrderID": "E123456",
  "session": "3",
  "capacity": "Principal",
  "member": "Mem01"
}
```

7.4. Order Restatement Example

This series of examples shows a restatement of a GTC order before market open the following day. Also it is assumed that a stock split on the symbol ABCD has taken effect, and that this is reflected in the restatement.

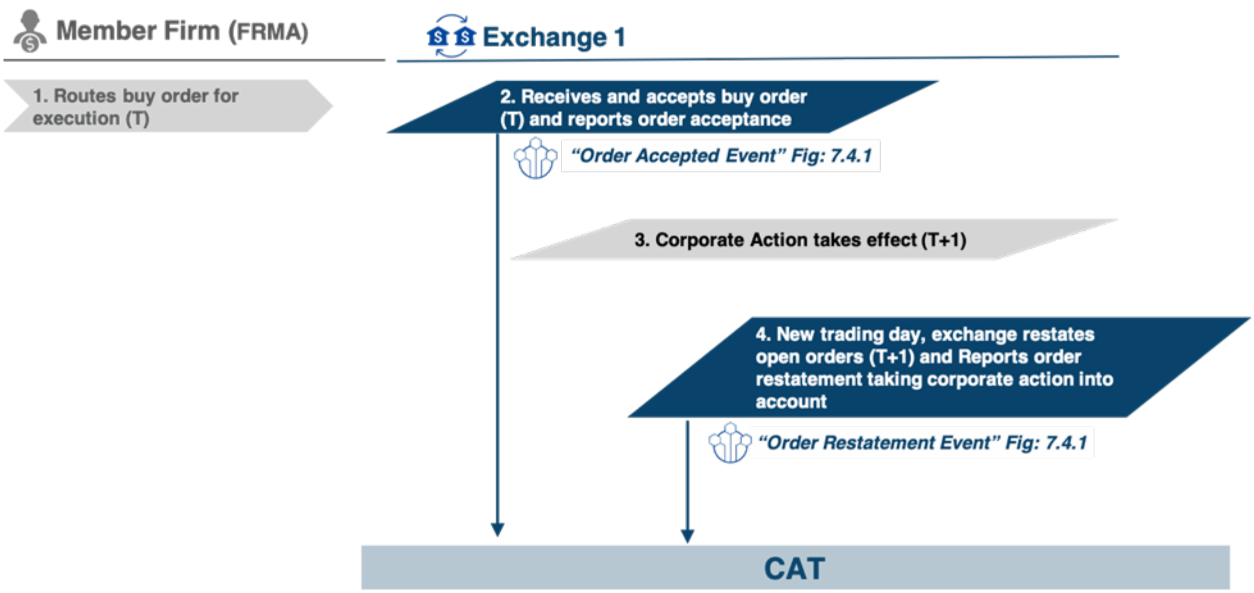

Figure 7: Order Restatement Example

Table 67: Order Restatement Example

#	Step	Reported Event	Comments
1	Member Firm FRMA Routes buy order for execution	NA	• A member firm routes a buy order to Exchange "Exch1" over session ID 7 with the order ID of 9153. This order is a buy order for the symbol ABCD, with a quantity of 500 at the price of 156.50
2	Exchange accepts the buy order and reports an order accepted event to CAT	**Order Accepted Event:** type: EOA exchange: Exch1 eventTimestamp: 20170307T134000.123456789 sequenceNumber: 11190 symbol: ABCD orderID: 1201 routingParty: FRMA	• The exchange accepts the buy order and assigns it the internal order ID: 1201. The order type is a limit order with time in force = GTC. • The ID that was used by the member firm is included as the Routed Order ID

#	Step	Reported Event	Comments
		routedOrderID: 9153 session: 7 side: Buy price: 156.50 quantity: 500 displayQty: 500 displayPrice: 156.50 workingPrice: 156.50 orderType: LMT timeInForce: GTC capacity: Agency nbbPrice: 157.00 nbbQty: 100 nboPrice: 157.25 nboQty: 100 member: Mem01	• The NBBO is as the exchange saw it just before accepting the order.
3	Corporate action takes effect		• A stock split event on the symbol ABCD takes effect 03/08/2017. This event has been reported to CAT by the listing exchange in its native CSV format since the corporate action was declared.
4	Exchanges restates open orders at the new trading day, reporting an Order Restatement Event taking the corporate action into account	**Order Restatement Event** type: EORS exchange: Exch1 eventTimestamp: 20170308T060000.123456789 sequenceNumber: 11000 symbol: ABCD orderID: 1202 originalOrderDate: 20170307 originalOrderID: 1201 side: Buy price: 78.25 quantity: 1000 displayQty: 1000 displayPrice: 78.25 workingPrice: 78.25 leavesQty: 1000 orderType: LMT timeInForce: GTC capacity: Agency member: Mem01	• This example shows the restatement of the GTC order (Order ID 1201) at market open the following day. In this example we also assume that a hypothetical stock split corporate action on the symbol ABCD has taken effect, and that none of the order has been filled. • Note that the Order ID can remain the same or be assigned anew, depending on how the exchange guarantees uniqueness within the same trading date. Also, the symbol mapping will possibly change from day to day. The symbol mapping for the new date is required. • Note that the quantity of the order has been doubled, and the price has been halved to reflect the stock split.

7.4.1.JSON Examples

Order Accepted Event


```
{
  "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170307T134000.123456789",
  "sequenceNumber": 11190,
  "symbol": "ABCD",
  "orderID": "1201",
  "routingParty": "FRMA",
  "routedOrderID": "9153",
  "session": "7",
  "side": "Buy",
  "price": 156.50,
  "quantity": 500,
  "displayQty": 500,
  "displayPrice": 156.50,
  "workingPrice": 156.50,
  "orderType": "LMT",
  "timeInForce": "GTC",
  "capacity": "Agency",
  "nbbPrice": 157.00,
  "nbbQty": 100,
  "nboPrice": 157.25,
  "nboQty": 100,
  "member": "Mem01"
}
```


Order Restatement Event


```
{
  "type": "EORS",
  "exchange": "Exch1",
  "eventTimestamp": "20170308T060000.123456789",
  "sequenceNumber": 11000,
  "symbol": "ABCD",
  "orderID": "1202",
  "origOrderDate": "20170307",
  "origOrderID": "1201",
  "side": "Buy",
  "price": 78.25,
  "quantity": 1000,
  "displayQty": 1000,
  "displayPrice": 78.25,
  "workingPrice": 78.25,
  "leavesQty": 1000,
  "orderType": "LMT",
  "timeInForce": "GTC",
  "capacity": "Agency",
  "member": "Mem01"
}
```

7.5. Order Modified Example

This section will show how an order modified event is reported when the order type is changed by the initiating member firm from a limit order to a market order. This series of events will follow the submission of a limit order from a member firm to the exchange that is subsequently modified by the member firm.

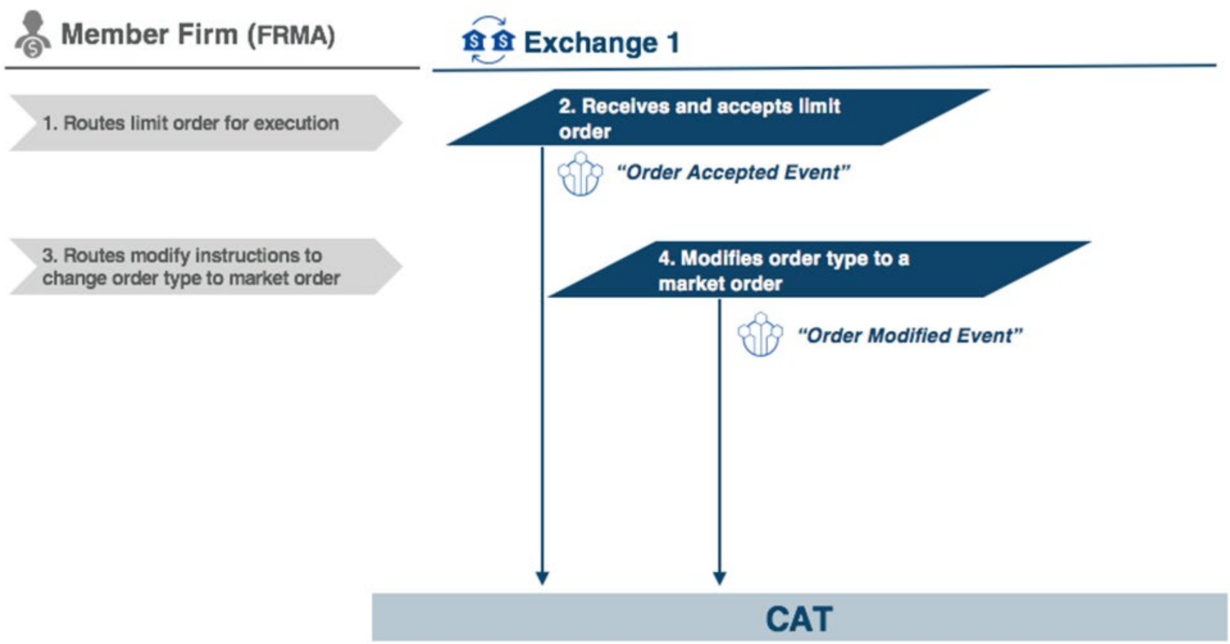

Figure 8: Order Modified Example

Table 68: Order Modified Example

#	Step	Reported Event	Comments
1	Member Firm Routes limit order for Execution		• A member firm routes an order to Exchange Exch1 over session ID 12 with the order ID of 1112. This order is a limit order for the symbol ABCD, with a quantity of 100
2	Exchange accepts the order and reports an order accepted event to CAT	***Order Accepted Event:*** type: EOA exchange: Exch1 eventTimestamp: 20170402T093001.123456789 sequenceNumber: 1001 symbol: ABCD	• The exchange accepts the order and assigns it the internal order ID: 98222. • This is order is a limit order with a limit price of 10.03

#	Step	Reported Event	Comments
		orderID: 98222 routingParty: FRMA routedOrderID: 1112 session: 12 side: Buy price: 10.03 quantity: 100 displayQty: 100 displayPrice: 10.03 workingPrice: 10.03 orderType: LMT timeInForce: DAY capacity: Principal nbbPrice: 10.00 nbbQty: 100 nboPrice: 10.05 nboQty: 100 member: Mem01	
3	Member Firm Routes modify instructions to Exchange to modify order to a Market Order		• routedOrderId = 1113 for modification to the firm order
4	Firm initiated new routedOrderId updates the order and reports an order modified event to CAT	**Order Modified Event:** type: EOM exchange: Exch1 eventTimestamp: 20170402T093055.123456789 sequenceNumber: 1091 symbol: ABCD orderID: 1_98222 originalOrderID: 98222 initiator: Firm side: Buy quantity: 100 displayQty: 100 displayPrice: 10.05 workingPrice: 10.05 leavesQty: 100 orderType: MKT timeInForce: DAY capacity: Principal nbbPrice: 10.00 nbbQty: 100 nboPrice: 10.05 nboQty: 100 member: Mem01 routedOrderId: 1113	• The exchange modifies the original order from a limit order to a market order (with no price) as initiated by FRMA • The modification results in a new order ID for the internal order. • In addition, the exchange reports to CAT the routedOrderId from the fix ClOrdId sent in to modify the order.

7.5.1.JSON Examples

Order Accepted Event


```
{
  "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093001.123456789",
  "sequenceNumber": 1001,
  "symbol": "ABCD",
  "orderID": "98222",
  "routingParty": "FRMA",
  "routedOrderID": "1112",
  "session": "12",
  "side": "Buy",
  "price": 10.03,
  "quantity": 100,
  "displayQty": 100,
  "displayPrice”: 10.03,
  "workingPrice": 10.03,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "nbbPrice": 10.00,
  "nbbQty": 100,
  "nboPrice": 10.05,
  "nboQty": 100,
  "member": "Mem01"
}
```


Order Modified Event


```
{
  "type": "EOM",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093055.123456789",
  "sequenceNumber": 1091,
  "symbol": "ABCD",
  "orderID": "1_98222",
  "originalOrderID": "98222",
  "initiator": "Firm",
  "side": "Buy",
  "quantity": 100,
  "displayQty": 100,
  "displayPrice": 10.05,
  "workingPrice": 10.05,
  "leavesQty": 100,
  "orderType": "MKT",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "nbbPrice": 10.00,
  "nbbQty": 100,
  "nboPrice": 10.05,
  "nboQty": 100,
  "member": "Mem01",
```

  "routedOrderId": "1113"
}

7.6. Order Modified for because of Partial Fill at Away Exchange

This Example is for an Equity Order Modify event where the exchange routes the order to an away exchange with a better market, and partially executes. The Order Modified Event is for the liquidity returned to the exchange after a partial execution. This example is to show how to populate the routedOrderId in the Equity Order Modified event for this scenario.

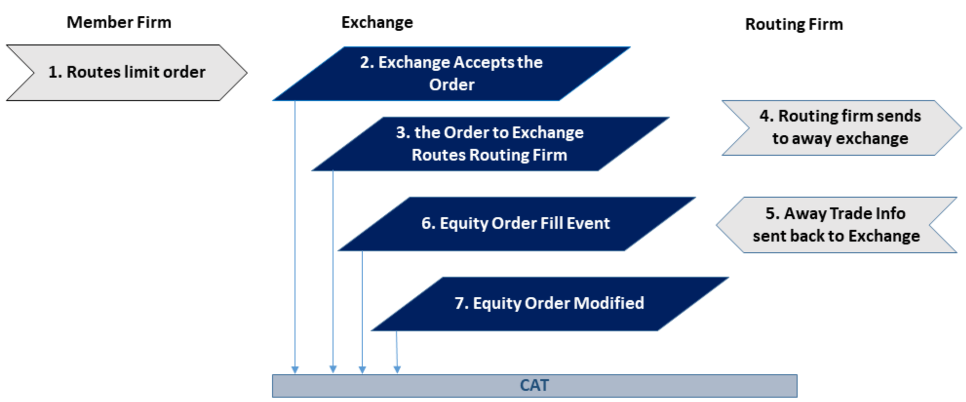

Figure 9: Order Modified Event due to a fill at an away exchange example

Table 69: Order Modified Example 2

#	Step	Reported Event	Comments
1	Member Firm Routes limit order for Execution		• A member firm routes an order to Exchange Exch1 over session ID 12 with the order ID of ZUA7197070219. This order

#	Step	Reported Event	Comments
			is a limit order for the symbol ABCD, with a quantity of 100
2	Exchange accepts the order and reports an order accepted event to CAT	***Order Accepted Event:*** type: EOA exchange: Exch1 eventTimestamp: 20170402T093001.123456789 sequenceNumber: 1001 symbol: ABCD orderID: 5882300 routingParty: FRMA routedOrderID: ZUA7197070219 session: 12 side: Buy price: 10.10 quantity: 100 displayQty: 100 displayPrice: 10.10 workingPrice: 10.10 orderType: LMT timeInForce: DAY capacity: Principal nbbPrice: 10.00 nbbQty: 100 nboPrice: 10.10 nboQty: 87 member: Mem01	• The exchange accepts the order and assigns it the internal order ID: 5882300. • This is order is a limit order with a limit price of 10.10
3	Exchange routes order to routing firm to send to an exchange with a better market	Equity Order Routed Event Type: EOR Exchange:Exch1 eventTimestamp: 20170402T093003.123456789 symbol: ABCD orderID : 5882300 routingParty : RouteFirm routedOrderId : 4827821 session: 12 side: Buy price: 10.10 quantity: 100 displayQty: 100 orderType: LMT timeInForce: DAY capacity: Principal result: ACK resultTimeStamp: 20170402T093003.123456799 member: Mem01, nbbPrice: 10.00 nboPrice: 10.10	• routedOrderId = 4827821 created by exchange to send to routing firm

#	Step	Reported Event	Comments
4	Routing firm sends firm to away exchange		
5	Routing Firm sends partial fill message back to exchange		
6	Trade occurred for 87 of the orders 100 contracts at the away exchange.	**Equity Order Fill Event** Type: Exch1 exchange: EOF eventTimestamp: 20170402T093005.123456799 fillId: 22 symbol: ABCD quantity: 87 price: 10.10 leavesQty = 13 orderID: 5882300 side: Buy clearingNumber: 355 contraClearingNumber: 888 routingParty: RouteFirm routedOrderId: 4827821 session: 12 capacity: Principal member: Mem01	
7	Exchange updates the order and reports an order modified event to CAT	**Order Modified Event:** type: EOM exchange: Exch1 eventTimestamp: 20170402T093055.123456789 symbol: ABCD orderID: 5882300 initiator: Firm nbbPrice: 10.00 nbbQty: 100 nboPrice: 10.05 nboQty: 13 Price: 10.10 quantity: 13 displayQty: 13 leavesQty: 13 orderType: LMT timeInForce: DAY capacity: Principal member: Mem01 routedOrderId: 4827821	• EOM event to change the original order quantity from 100 to 13. • The routedOrderId fields is populated with the routedOrderID sent to the routing firm in the EOR event.

7.6.1.JSON Examples

<u>Order Accepted Event</u>

```
{
  "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093001.123456789",
  "sequenceNumber": 1001,
  "symbol": "ABCD",
  "orderID": "5882300",
  "routingParty": "FRMA",
  "routedOrderID": " ZUA7197070219",
  "session": "12",
  "side": "Buy",
  "price": 10.10,
  "quantity": 100,
  "displayQty": 100,
  "displayPrice": 10.10,
  "workingPrice": 10.10,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "nbbPrice": 10.00,
  "nbbQty": 100,
  "nboPrice": 10.10,
  "nboQty": 87,
  "member": "Mem01"
```

<u>Order Route Event</u>

```
{
  "type": "EOR",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093003.123456789",
  "symbol": "ABCD",
  "orderID": "5882300",
  "routingParty": "RouteFirm",
  "routedOrderID": "4827821",
  "session": "12",
  "side": "Buy",
  "price": 10.10,
  "quantity": 100,
  "displayQty": 100,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "result": "ACK",
  "resultTimestamp": "20170402T093003.123456799",
  "nbbPrice": 10.00,
  "nboPrice": 10.10,
  "member": "Mem01"
}
```

Order Fill Event


```
{
  "type": "EOF",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093005.123456799 ",
  "fillID": "22",
  "symbol": "ABCD",
  "price": 10.10,
  "side": "Buy",
  "quantity": 87,
  "leavesQty": 13,
  "orderID": 5882300,
  "clearingNumber": "355",
  "contraClearingNumber":"888"
  "routingParty": "RouteFirm",
  "routedOrderID": "4827821",
  "session": "12",
  "capacity": "Principal",
  "member": "Mem01"
}
```


Order Modified Event


```
{
  "type": "EOM",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093055.123456789",
  "symbol": "ABCD",
  "orderID": "5882300",
  "initiator": "Firm",
  "quantity": 13,
  "displayQty": 13,
  "leavesQty": 13,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "nbbPrice": 10.00,
  "nbbQty": 100,
  "nboPrice": 10.05,
  "nboQty": 100,
  "member": "Mem01",
  "routedOrderId": "4827821"
}
```


7.7. Order Adjusted Example

This section will show how an order adjusted event is reported when a change in the NBBO causes the working price of an order to change. This series of events will follow the route of a peg order followed by an adjustment of the working price.

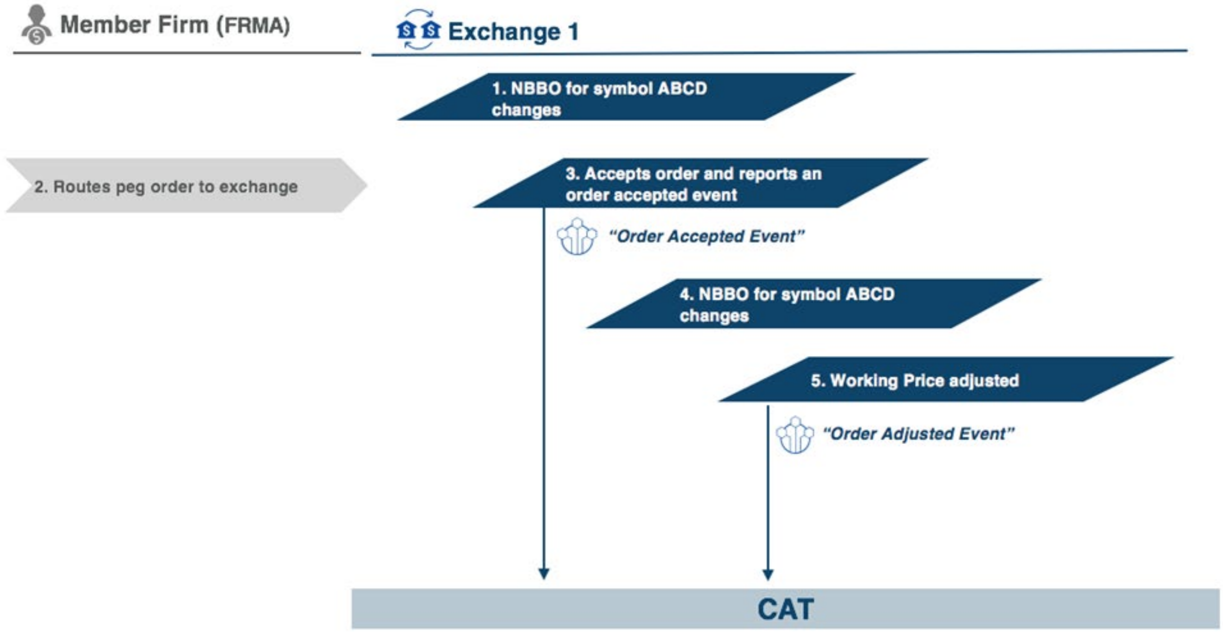

Figure 10: Order Adjusted Example

Table 70: Order Adjusted Example

#	Step	Reported Event	Comments
1	NBBO for symbol ABCD changes		• NBBO for symbol is updated to 10.00X10.05
2	Member Firm Routes order for Execution		• A member firm routes an order to Exchange Exch1 over session ID 12 with the order ID of 1112. This order is a mid-peg order for the symbol ABCD, with a quantity of 100
3	Exchange accepts the order and reports an order accepted event to CAT	***Order Accepted Event:*** type: EOA exchange: Exch1 eventTimestamp: 20170402T093001.123456789 sequenceNumber: 10001 symbol: ABCD orderID: 98222 routingParty: FRMA routedOrderID: 1112 session: 12 side: Buy	• The exchange accepts the buy order and assigns it the internal order ID: 98222 • This is order is a mid-peg order with a limit price of 10.03 • If there were no limit price, then the price field would not be included in JSON or blank in CSV

#	Step	Reported Event	Comments
		price: 10:03 quantity: 100 displayQty: 0 workingPrice: 10.025 orderType: PEG timeInForce: DAY capacity: Principal handlingInstructions: AON nbbPrice: 10.00 nbbQty: 100 nboPrice: 10.05 nboQty: 100 member: Mem01	
4	NBBO for symbol ABCD changes		• The NBBO for symbol ABCD changes from 10.00X10.05 to 10.01X10.05
5	Exchange updates the handling instructions for the peg order	***Order Adjusted Event:*** type: EOJ exchange: Exch1 eventTimestamp: 20170402T093015.123456789 sequenceNumber: 10091 symbol: ABCD orderID: 98222 initiator: Exchange price: 10.03 workingPrice: 10.03 nbbPrice: 10.01 nbbQty: 100 nboPrice: 10.05 nboQty: 100 member: Mem01 handlingInstructions: FOK	• Because the NBBO has changed, the working price will be updated. • The orderID does not change, so originalOrderID does not need to be included. • Note, routedOrderId does not need to be reported since this is an exchange initiated event (initiator = "Exchange").

7.7.1.JSON Examples

__Order Accepted Event__


```
{
  "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093001.123456789",
  "sequenceNumber": 10001,
  "symbol": "ABCD",
  "orderID": "98222",
  "routingParty": "FRMA",
  "routedOrderID": "1112",
```

```
  "session": "12",
  "side": "Buy",
  "price": 10.03,
  "quantity": 100,
  "displayQty": 0,
  "workingPrice": 10.025,
  "orderType": "PEG",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "handlingInstructions": "AON",
  "nbbPrice": 10.00,
  "nbbQty": 100,
  "nboPrice": 10.05,
  "nboQty": 100,
  "member": "Mem01"
}
```


Order Adjusted Event


```
{
  "type": "EOJ",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093015.123456789",
  "sequenceNumber": "10091",
  "symbol": "ABCD",
  "orderID": "98222",
  "initiator": "Exchange",
  "price": 10.03,
  "workingPrice": 10.03,
  "nbbPrice": 10.01,
  "nbbQty": 100,
  "nboPrice": 10.05,
  "nboQty": 100,
  "member": "Mem01"
  "handlingInstructions": "FOK"
}
```


7.8. Order Adjusted Example Firm Initiated

The following example illustrates how the routedOrderID should be populated in an order adjusted event if a firm routes in a change to the order to the exchange.

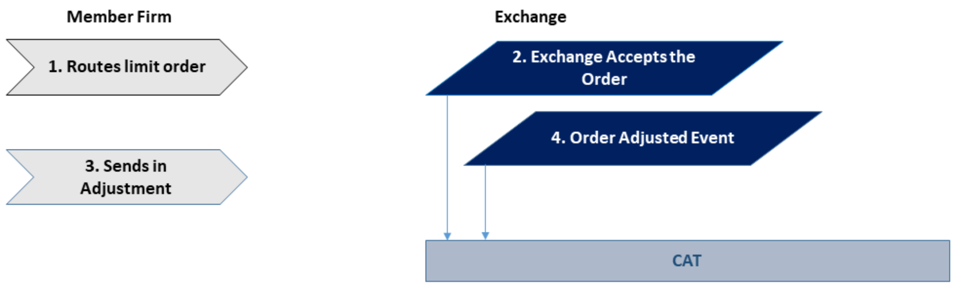

Figure 11: Order Adjusted due to a firm message example

#	Step	Reported Event	Comments
1	Firm routes buy limit peg order to exchange. Exchange Order Accepted Event created		
2	Exchange creates Equity Order Accepted Event	type: EOA exchange: Exch1 eventTimestamp: 20170402T093001.123456789 sequenceNumber: 12 symbol: TSLA orderID: 3127867394 routingParty: RFIRMA routedOrderID: 3543550 session: 12 side: Buy price: 10:03 quantity: 100 displayQty: 0 workingPrice: 10.025 orderType: PEG timeInForce: DAY capacity: Principal handlingInstructions: AON nbbPrice: 10.00 nbbQty: 100 nboPrice: 10.05	

#	Step	Reported Event	Comments
		nboQty: 100 member: Mem01	
3	Firm sends in change to order to modify the quantity from 100 to 50..		
4	Firm adjusts quantity on peg order. Order Adjusted event sent to CAT with routedOrderId sent in from firm.	***Order Adjusted Event:*** type: EOJ exchange: Exch1 eventTimestamp: 20170402T093005.123456789 sequenceNumber: 44 symbol: TSLA orderID: 3127867394 initiator: Firm quantity: 50 workingPrice: 10.025 nbbPrice: 10.01 nbbQty: 100 nboPrice: 10.05 nboQty: 100 member: Mem01 routedOrderId: 3543551	• Example of customer initiated order adjustment event with required routedOrderId

7.8.1.JSON Examples

Order Accepted Event


```
{
 "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093001.123456789",
  "sequenceNumber": 12,
  "symbol": "TSLA",
  "orderID": "3127867394",
  "routingParty": "RFIRMA",
  "routedOrderID": "3543550",
  "session": "12",
  "side": "Buy",
  "price": 10.03,
  "quantity": 100,
  "displayQty": 0,
  "workingPrice": 10.025,
  "orderType": "PEG",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "handlingInstructions": "AON",
  "nbbPrice": 10.00,
  "nbbQty": 100,
```

```
  "nboPrice": 10.05,
  "nboQty": 100,
  "member": "Mem01"
}
```


Order Adjusted Event


```
{
  "type": "EOJ",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093005.123456789",
  "sequenceNumber": "44",
  "symbol": "TSLA",
  "orderID": "3127867394",
  "initiator": "Firm",
  "quantity": 50
  "workingPrice": 10.025,
  "nbbPrice": 10.01,
  "nbbQty": 100,
  "nboPrice": 10.05,
  "nboQty": 100,
  "member": "Mem01",
  "routedOrderId": 3543551
  }
```


7.9. Order Adjusted Event because of Partial Execution at Away Exchange

This example shows the scenario where an order is partially filled at an away exchange instigating an option order adjusted event to change the quantity. The option order adjusted event has the routedOrderId populated with the value sent to the routing firm.

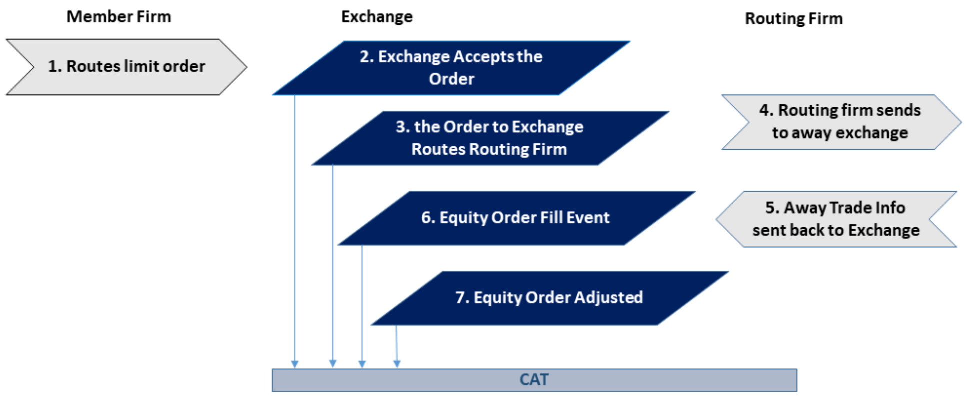

Figure 12: Order Adjusted due to a fill at and away exchange example

#	Step	Reported Event
1	Member Firm Routes limit order for Execution	
2	Exchange accepts the order and reports an order accepted event to CAT	***Order Accepted Event:*** type: EOA exchange: Exch1 eventTimestamp: 20170402T093001.123456789 sequenceNumber: 1001 symbol: ABCD orderID: 5882300 routingParty: FRMA routedOrderID: ZUA7197070219 session: 12 side: Buy price: 10.10 quantity: 100 displayQty: 100 displayPrice: 10.10 workingPrice: 10.10 orderType: LMT timeInForce: DAY capacity: Principal nbbPrice: 10.00 nbbQty: 100 nboPrice: 10.10

#	Step	Reported Event
		nboQty: 87 member: Mem01
3	Exchange routes order to routing firm to send to an exchange with a better market	Equity Order Routed Event Type: EOR Exchange:Exch1 eventTimestamp: 20170402T093003.123456789 symbol: ABCD orderID : 5882300 routingParty : RouteFirm routedOrderId : 4827821 session: 12 side: Buy price: 10.10 quantity: 100 displayQty: 100 orderType: LMT timeInForce: DAY capacity: Principal result: ACK resultTimestamp: 20170402T093003.123456799 member: MEM, nbbPrice: 10.00 nboPrice: 10.10
4	Routing Firm sends order to away exchange.	
5	Routing Firm returns executed liquidity to the exchange.	
6	Trade occurred for 87 of the orders 100 contracts at the away exchange.	**Equity Order Fill Event** Type: Exch1 exchange: EOF eventTimestamp: 20170402T093005.123456799 fillId: 22 symbol: ABCD quantity: 87 price: 10.10 leavesQty = 13 orderID: 5882300 side: Buy clearingNumber: 355 contraClearingNumber: 888 routingParty: RouteFirm

#	Step	Reported Event
		routedOrderId: 4827821 session: 12 capacity: Principal member: Mem01
7	An order adjust event is sent to CAT to represent the change in quantity.	**Order Adjusted Event:** type: EOJ exchange: Exch1 eventTimestamp: 20170402T093055.123456789 symbol: ABCD orderID: 5882300 initiator: Firm nbbPrice: 10.00 nbbQty: 100 nboPrice: 10.05 nboQty: 13 quantity: 13 capacity: Principal member: Mem01 routedOrderId: 4827821

7.9.1.JSON Examples

<u>Order Accepted Event</u>


```
{
  "type": "EOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093001.123456789",
  "sequenceNumber": 1001,
  "symbol": "ABCD",
  "orderID": "5882300",
  "routingParty": "FRMA",
  "routedOrderID": " ZUA7197070219",
  "session": "12",
  "side": "Buy",
  "price": 10.10,
  "quantity": 100,
  "displayQty": 100,
  "displayPrice": 10.10,
  "workingPrice": 10.10,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "nbbPrice": 10.00,
  "nbbQty": 100,
  "nboPrice": 10.10,
  "nboQty": 87,
  "member": "Mem01"
```

Order Route Event


```
{
  "type": "EOR",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093003.123456789",
  "symbol": "ABCD",
  "orderID": "5882300",
  "routingParty": "RouteFirm",
  "routedOrderID": "4827821",
  "session": "12",
  "side": "Buy",
  "price": 10.10,
  "quantity": 100,
  "displayQty": 100,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "capacity": "Principal",
  "result": "ACK",
  "resultTimestamp": "20170402T093003.123456799",
  "nbbPrice": 10.00,
  "nboPrice": 10.10,
  "member": "Mem01"
}
```


Order Fill Event


```
{
  "type": "EOF",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093005.123456799 ",
  "fillID": "22",
  "symbol": "ABCD",
  "price": 10.10,
  "side": "Buy",
  "quantity": 87,
  "leavesQty": 13,
  "orderID": 5882300,
  "clearingNumber": "355",
  "contraClearingNumber":"888"
  "routingParty": "RouteFirm",
  "routedOrderID": "4827821",
  "session": "12",
  "capacity": "Principal",
  "member": "Mem01"
}
```


Order Adjusted Event


```
{
  "type": "EOJ",
  "exchange": "Exch1",
  "eventTimestamp": "20170402T093055.123456789",
```

```
  "symbol": "ABCD",
  "orderID": "5882300",
  "initiator": "Firm",
  "quantity": 13,
  "capacity": "Principal",
  "nbbPrice": 10.00,
  "nbbQty": 100,
  "nboPrice": 10.05,
  "nboQty": 100,
  "member": "Mem01",
  "routedOrderId": "4827821"
}
```

8. Options Exchange Event Examples

8.1. Quote and Quote Cancel Events

Some exchanges use the term "order" to cover both quotes and non-quote orders. For the purpose of reporting to CAT, a quote is to be interpreted as an order/quote that qualifies as a market maker quote for the purposes of satisfying Section 6.4(d)(iii) of the CAT NMS Plan. That is the section which grants relief to market makers from reporting their quotes to CAT, leaving the exchanges themselves with the sole responsibility of reporting quotes to CAT. If such order/quotes received by the exchange would provide the market maker an exemption from reporting the quote, then the order/quote must be reported to CAT as a quote, not an order.

CAT accepts both one-sided and two-sided quotes.

8.1.1. Two-Sided Quotes Example

The following section will provide examples of reportable events for a two-sided market maker quote when it is posted as a new quote, updated by the market maker, then canceled by the market maker or the exchange. Both the new quote and the updated quote are expressed by the Quote Event, while the quote cancel is expressed by the Quote Cancel Event.

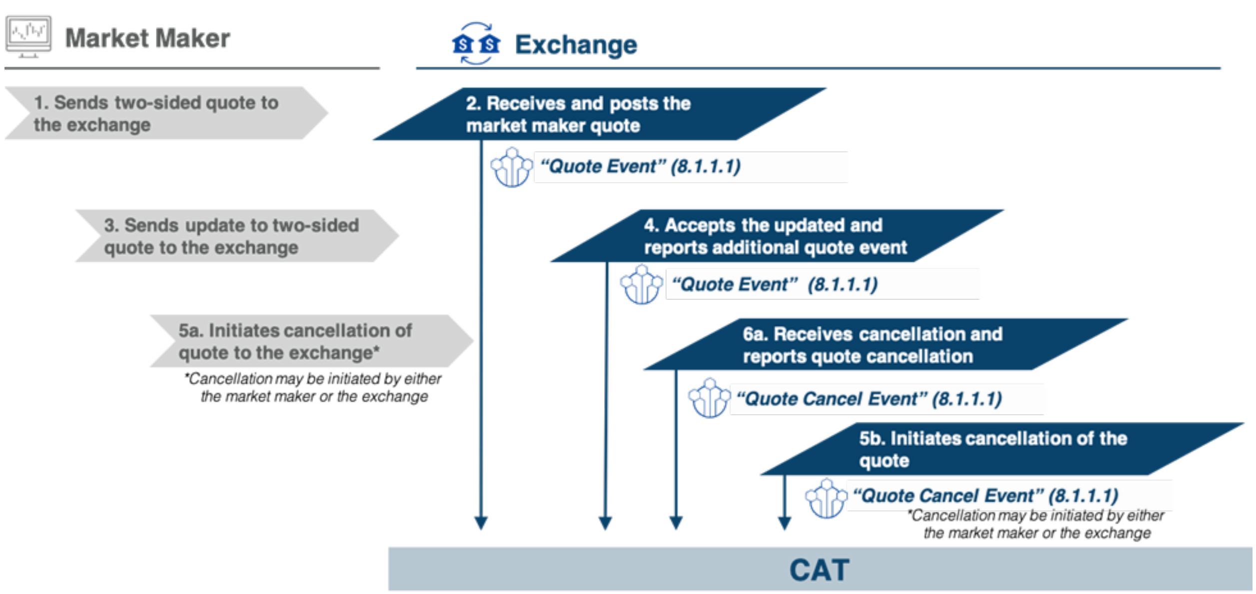

Figure 13: Two-Sided Quote Example

Table 71: Two-Sided Quote Example

#	Step	Reported Event	Comments
1	Market maker sends two-sided quote to the exchange	NA	• Market Maker sends updated two sided (buy/sell) quotes, updates them and cancels them
2.	Exchange 1 posts the market maker quote	**Quote Event** type: OQ exchange: Exch1 eventTimestamp: 20170113T132436.124039 sequenceNumber:1245 marketMaker: ABCD:A16 sentTimestamp: 20170113T132436.123456 optionID: 6779 quoteID: Q9876 onlyOneQuote: true, bidPrice: 2.40 bidQty: 10 askPrice: 2.43 askQty: 10	• The quote is a two-sided quote for an option with the ID: 6779 • The field market maker is the Member Alias assigned by the SRO to identify the market maker issuing the quote. In this case, the market maker ABCD has multiple users (e.g., acronyms used to differentiate users within the same MM), so the characters A16 denote the user or sub-account. • The sent timestamp denotes when the market maker sent the quote to the marketplace, while the event timestamp is when the exchange received the quote
3	Market maker sends an update to the two sided quote to the exchange		• The market maker sends an update raising the bid price of the original quote to 2.41
4	Exchange accepts the update and reports a quote event	**Quote Event** type: OQ exchange: Exch1 eventTimestamp: 20170113T132536.123486789 sequenceNumber: 1278 marketMaker: ABCD:A16 sentTimestamp: 20170113T132536.123456 optionID: 6779 quoteID: Q9941, onlyOneQuote: true, bidPrice: 2.41 bidQty: 10 askPrice: 2.43 askQty: 10	• The quote event reported by the exchange effectively replaces the former quote, assigning a new quote ID • Note that the quote ID is new: Q9941. Because the MM has only one quote in this optionID, the originalQuoteID is not required. • Bid Price is updated, however Bid Quantity, Ask Price, and Ask Quantity remain unchanged
5a	Market maker initiates cancellation of the quote		• Market maker sends a cancellation notice of its quote to the exchange

#	Step	Reported Event	Comments
5b	Exchange receives the cancellation and reports an order cancellation event	**Quote Cancel Event** type: OQC exchange: Exch1 eventTimestamp: 20170113T133036.123486789 sequenceNumber: 1299 marketMaker: ABCD:A16 sentTimestamp: 20170113T133036.123456 optionID: 6779 quoteID: Q9941, onlyOneQuote: true, initiator: MarketMaker cancelReason: ALL	• The value for cancel initiator must always be either market maker or exchange. • The field cancel reason allows for more detail to explain the cancel. In this case ALL represents - Market Maker canceled all quotes. Refer to the data dictionary for more possible values.
6a/b	Exchange initiates cancellation of the quote	**Quote Cancel Event** type: OQC exchange: Exch1 eventTimestamp: 20170113T133105.123456789 sequenceNumber: 1308 marketMaker: ABCD:A16 quoteID: Q9941, onlyOneQuote: true, initiator: Exchange cancelReason: DIS	• This step represents an example where the exchange cancels the quote. • There is no Sent Timestamp value because the event was initiated by the exchange, not the market maker. • The field cancel reason allows for more detail to explain the cancel, possible values may be specified by the exchange. In this case DIS represents that the quote was canceled due to a lost connection. Refer to the data dictionary for more possible values

8.1.1.1. JSON Examples

<u>Quote Event (Step 2)</u>


```
{
    "type": "OQ",
    "exchange": "Exch1",
    "eventTimestamp": "20170113T132436.124039",
    "sequenceNumber": 1245,
    "marketMaker": "ABCD:A16",
    "sentTimestamp": "20170113T132436.123456",
    "optionID": "6779",
    "quoteID": "Q9876",
    "onlyOneQuote": true,
    "bidPrice": 2.40,
    "bidQty": 10,
```

```
    "askPrice": 2.43,
    "askQty": 10
}
```

Quote Event (Step 4)

```
{
    "type": "OQ",
    "exchange": "Exch1",
    "eventTimestamp": "20170113T132536.123486789",
    "sequenceNumber": 1278,
    "marketMaker": "ABCD:A16",
    "sentTimestamp": "20170113T132536.123456",
    "optionID": "6779",
    "quoteID": "Q9941",
    "onlyOneQuote": true,
    "bidPrice": 2.41,
    "bidQty": 10,
    "askPrice": 2.43,
    "askQty": 10,
}
```

Quote Cancel Event (Step 6a)

```
{
    "type": "OQC",
    "exchange": "Exch1",
    "eventTimestamp": "20170113T133036.123486789",
    "sequenceNumber": 1299,
    "marketMaker": "ABCD:A16",
    "sentTimestamp": "20170113T133036.123456",
    "optionID": "6779",
    "quoteID": "Q9941",
    ""onlyOneQuote": true,
    "initiator": "MarketMaker",
    "cancelReason": "A"
}
```

Quote Cancel Event (Step 5b)

```
{
    "type": "OQC",
    "exchange": "Exch1",
    "eventTimestamp": "20170113T133105.123456789",
    "sequenceNumber": 1308,
    "marketMaker": "ABCD:A16",
    "quoteID": "Q9941",
    "onlyOneQuote": true,
    "initiator": "Exchange",
    "cancelReason": "DIS"
}
```

8.1.2.One-Sided Quotes Example

The following section will provide examples of reported events for a one-sided market maker quote when it is posted as a new quote, updated by the market maker, then canceled by the market maker or the exchange. Both the new quote and the update are expressed by the Quote Event, while the quote cancel is expressed by the Quote Cancel Event.

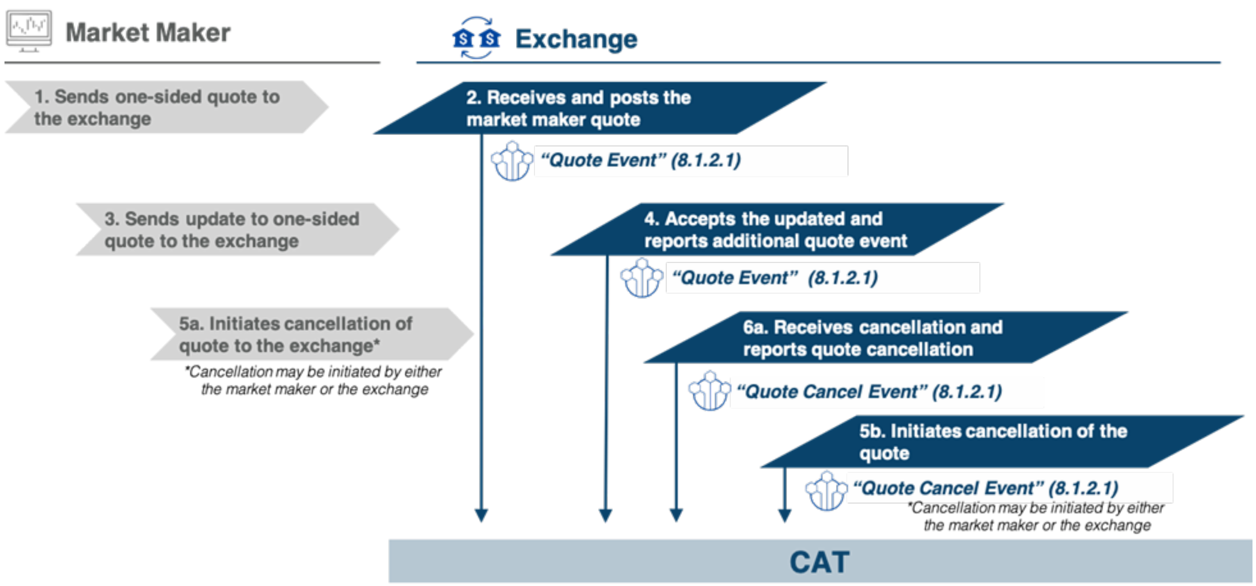

Figure 14: One-Sided Quotes Example

Table 72: One-Sided Quotes Example

#	Step	Reported Event	Comments
1	Market maker sends one-sided quote to the exchange	NA	• Market Maker sends one-sided quotes, updates them and cancels them in that sequence
2	Exchange 1 posts the market maker quote	**Quote Event** Type: OQ Exchange ID: Exch1 eventTimestamp: 20170113T142036.123486789 sequenceNumber: 1010 marketMaker: EFGH:A1 sentTimestamp: 20170113T142036.123456 optionID: 1208 quoteID: Q123456	• The quote is a one-sided quote for an option with the ID: 1208 • The field market maker is the Member Alias assigned by the SRO to identify the market maker issuing the quote. In this case, the market maker EFGH has multiple users (e.g., acronyms used to differentiate users within the

#	Step	Reported Event	Comments
		onlyOneQuote: false bidPrice: 6.10 bidQty: 20	same MM), so the characters A1 denote the user or sub-account. • The sent timestamp denotes when the market maker sent the quote to the marketplace, while the event timestamp is when the exchange received the quote • The option ID is the ID of the option as assigned by the exchange
3	Market maker sends an update to the one sided quote to the exchange		• The market maker sends an update raising the quantity of the original quote to 30
4	Exchange accepts the update and reports a quote event	**Quote Event** Type: OQ Exchange ID: Exch1 eventTimestamp: 20170113T142536.123486789 sequenceNumber: 1038 marketMaker: EFGH:A1 sentTimestamp: 20170113T142536.123456 optionID: 1208 quoteID: Q22222 originalQuoteID: Q123456 onlyOneQuote: false bidPrice: 6.10 bidQty: 30	• The quote event reported by the exchange effectively replaces the former quote, assigning a new quote ID • Note that the quote ID is new: Q22222, while the former quote ID is included in the field Original Quote ID • Bid Quantity is updated, however Bid price is unchanged
5a	Market maker initiates cancellation of the quote		• Market maker sends a cancellation notice of its quote to the exchange
5b	Exchange receives the cancellation and reports an order cancellation event	**Quote Cancel Event** type: OQC exchange: Exch1 sentTimestamp: 20170113T143036.123456 eventTimestamp: 20170113T143036.123486789 sequenceNumber: 1142 marketMaker: EFGH:A1 optionID: 1208 quoteID: Q22222 onlyOneQuote: false initiator: MarketMaker cancelReason: ALL	• The value for cancel initiator must always be either market maker or exchange. • The field cancel reason allows for more detail to explain the cancel. In this case ALLrepresents - Market Maker canceled all quotes. Refer to the data dictionary for more possible values.

#	Step	Reported Event	Comments
6a/b	Exchange initiates cancellation of the quote	**Quote Cancel Event** type: OQC exchange: Exch1 eventTimestamp: 20170113T143105.123456789 sequenceNumber: 1142 marketMaker: EFGH:A1 optionID: 1208 quoteID: Q22222 onlyOneQuote: false initiator: Exchange cancelReason: DIS	• This step displays an example where the quote is canceled by the exchange • There is no Sent Timestamp value because the event was initiated by the exchange, not the market maker. • The field cancel reason allows for more detail to explain the cancel, possible values may be specified by the exchange. In this case DIS represents that the quote was canceled due to a lost connection. Refer to the data dictionary for more possible values

8.1.2.1. JSON Examples

Quote Event (Step 2)


```
{
    "type": "OQ",
    "exchange": "Exch1",
    "eventTimestamp": "20170113T142036.123486789",
    "sequenceNumber": 1010,
    "marketMaker": "EFGH:A1",
    "sentTimestamp": "20170113T142036.123456",
    "optionID": "1208",
    "quoteID": "Q123456",
    "onlyOneQuote": false,
    "bidPrice": 6.10,
    "bidQty": 20
}
```


Quote Event (Step 4)


```
{
    "type": "OQ",
    "exchange": "Exch1",
    "eventTimestamp": "20170113T142536.123486789",
    "sequenceNumber": 1038,
    "marketMaker": "EFGH:A1",
    "sentTimestamp": "20170113T142536.123456",
    "optionID": "1208",
    "quoteID": "Q22222",
    "originalQuoteID": "Q123456",
    "onlyOneQuote": false,
    "bidPrice": 6.10,
```

```
    "bidQty": 30
}
```

Quote Cancel Event (Step 6a)


```
{
    "type": "OQC",
    "exchange": "Exch1",
    "sentTimestamp": "20170113T143036.123456",
    "eventTimestamp": "20170113T143036.123486789",
    "sequenceNumber": 1142,
    "marketMaker": "EFGH:A1",
    "optionID": "1208",
    "quoteID": "Q22222",
    "onlyOneQuote": false,
    "initiator": "MarketMaker",
    "cancelReason": "ALL"
}
```


Quote Cancel Event (Step 5b)


```
{
    "type": "OQC",
    "exchange": "Exch1",
    "eventTimestamp": "20170113T143105.123456789",
    "sequenceNumber": 1142,
    "marketMaker": "EFGH:A1",
    "optionID": "1208",
    "quoteID": "Q22222",
    "onlyOneQuote": false,
    "initiator": "Exchange",
    "cancelReason": "DIS"
}
```


8.2. Option Order Event Examples

8.2.1. Simple Option Order Accepted Example

This example describes a Simple Option Order Accepted Event in which the exchange receives and accepts an order for a simple option. Note that in this example Complex Order ID is not provided because there is no parent complex order.

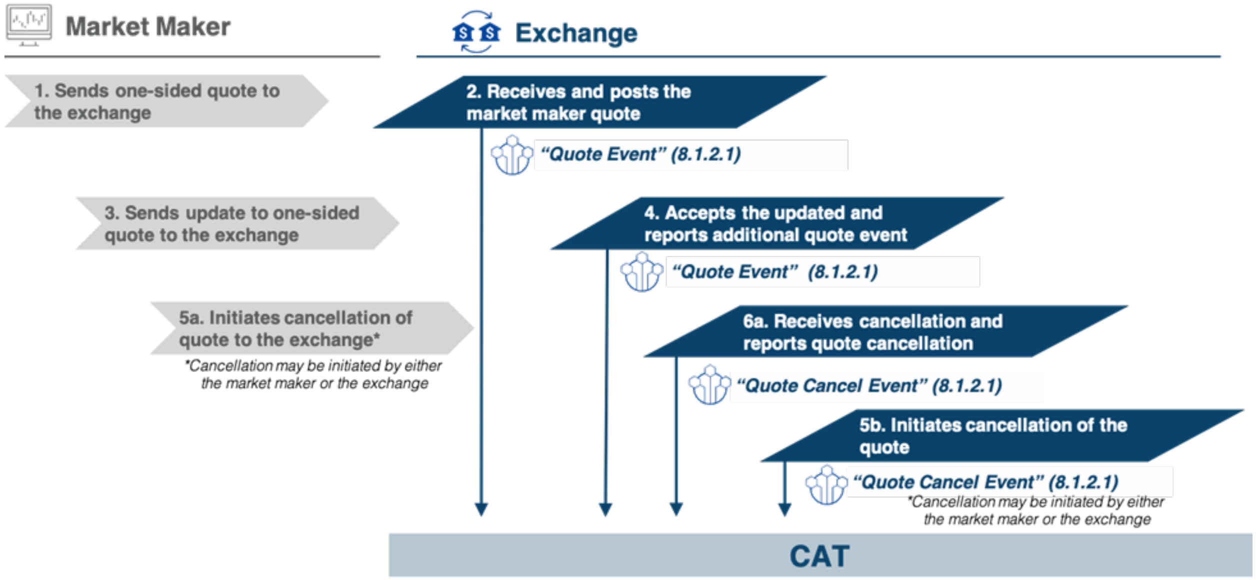

Figure 15: Simple Option Order Accepted Example

Table 73: Simple Option Order Accepted Example

#	Step	Reported Event	Comments
1	Member firm sends option order to the exchange	NA	• The order is routed over session ID 3, with a price of 18.59, quantity of 10, for the option defined by the exchange as Option ID 1208
2.	Exchange 1 accepts the order and reports a Simple Option Order Accepted Event	**Simple Option Oder Accepted Event:** type: OOA exchange: Exch1 eventTimestamp: 20170116T143105.123456789 sequenceNumber: 909 optionID: 1208 orderID: 123456 routingParty: FRMA routedOrderID: 98765 session: 3 side: Buy price: 18.59 quantity: 10 displayQty: 10 displayPrice: 18.59 workingPrice: 18.59 openCloseIndicator: Open	• The option ID is the ID of the option as assigned by the exchange. • The Order ID is the ID of the order as assigned by the exchange, while the routed order ID is the order ID as defined by the member firm. • The origin code value of C represents that the order originated from a customer

#	Step	Reported Event	Comments
		orderType: LMT timeInForce: DAY exchOriginCode: C coverage: Uncovered executingFirm: 999 nbbPrice: 18.58 nbbQty: 10 nboPrice: 18.60 nboQty: 10 member: Mem01	

8.2.1.1. JSON Example

Simple Option Order Accepted Event


```
{
  "type": "OOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170116T143105.123456789",
  "sequenceNumber": 909,
  "optionID": "1208",
  "orderID": "123456",
  "routingParty": "FRMA",
  "routedOrderID": "98765",
  "session": "3",
  "side": "Buy",
  "price": 18.59,
  "quantity": 10,
  "displayQty": 10,
  "displayPrice": 18.59,
  "workingPrice": 18.59,
  "openCloseIndicator": "Open",
  "orderType": "LMT",
  "timeInForce": "DAY",
  "exchOriginCode": "C",
  "coverage": "Uncovered",
  "executingFirm": "999",
  "nbbPrice": 18.58,
  "nbbQty": 10,
  "nboPrice": 18.60,
  "nboQty": 10,
  "member": "Mem01"
}
```


8.2.2. Complex Option Order Accepted Event Example

In the example below, the exchange only creates leg orders at the time an order is executed. Thus, an order on the complex option would have a report sent to CAT for an order accepted event at the parent level of the complex order. Any leg reports would wait until the leg orders are actually created when a trade occurs.

The examples in this section will use an order on the complex option with optionID 9843. This hypothetical complex option has two option series legs:

<u>Complex Option – optionID: 9843</u>

Table 74: Complex Option Order Example: Legs

optionID	side	ratio	primaryDeliverable	expirationDate	strikePrice	putCall	exerciseStyle	settlement
1491	Buy	1	XYZZY	21 Oct 2017	30.00	C	American	PM
1492	Sell	1	XYZZY	21 Oct 2017	32.50	C	American	PM

For this example, we suppose at 192411.121456789 on April 20, 2017 an order was accepted for 10 units of complex option 9843 at net price -65 per unit.

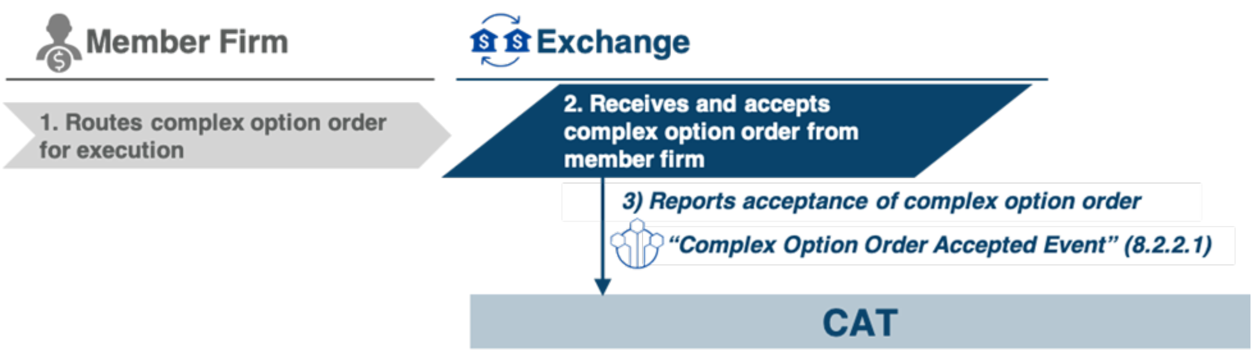

Figure 16: Complex Option Order Example

Table 75: Complex Option Order Example

#	Step	Reported Event	Comments
1	Market maker sends complex option order to the exchange	NA	• The order is routed over session ID 7, with a price of -65, quantity of 10, for the option defined by the exchange as Option ID 9843
2	Exchange 1 accepts the complex option order		

#	Step	Reported Event	Comments
3	Exchange 1 reports a complex option order accepted event. Leg events are not reported until an execution happens, so the only event reported at this time is for the complex option order.	**Complex Option Order Accepted Event** type: OCOA exchange: Exch1 eventTimestamp: 20170420T142411.121456789 sequenceNumber: 909 optionID: 9843 orderID: 8473692 side: AsDirected routingParty: FRMA routedOrderID: 4567123 session: 7 price: -65.00 quantity: 10 timeInForce: DAY member: Mem01	• The option ID is the ID of the option as assigned by the exchange. • The Order ID is the ID of the order as assigned by the exchange, while the routed order ID is the order ID as defined by the member firm.

8.2.2.1. JSON Examples

<u>**Complex Order Accepted Event (Step 3)**</u>


```
{
  "type": "OCOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142411.121456789",
  "sequenceNumber": 909,
  "optionID": "9843",
  "orderID": "8473692",
  "side": "AsDirected",
  "routingParty": "FRMA",
  "routedOrderID": "4567123",
  "session": "7",
  "price": -65.00,
  "quantity": 10,
  "timeInForce": "DAY",
  "member": "Mem01"
}
```


8.3. Simple Option Order Modified Event

This example shows how to populate the routedOrderId for a firm initiated modification.

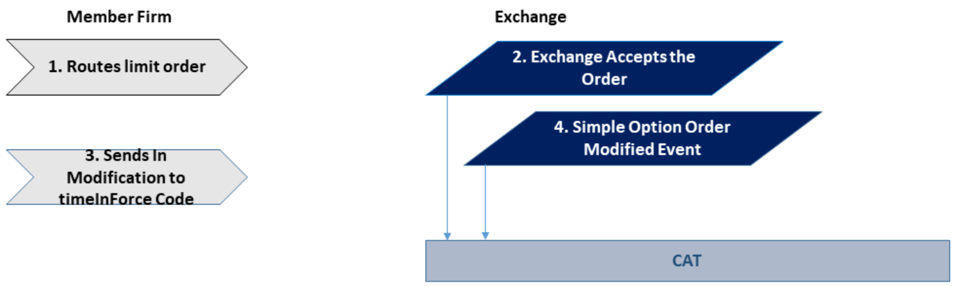

Figure 17: Simple Option Order Modify Event due to a firm change

#	Step	Reported Event	Comments
1	Member firm sends option order to the exchange	NA	• The order is routed over session ID 3, with a price of 18.59, quantity of 10, for the option defined by the exchange as Option ID 1208
2.	Exchange 1 accepts the order and reports a Simple Option Order Accepted Event	**Simple Option Oder Accepted Event:** type: OOA exchange: Exch1 eventTimestamp: 20170116T143105.123456789 sequenceNumber: 909 optionID: 1208 orderID: 123456 routingParty: FRMA routedOrderID: 98765 session: 3 side: Buy price: 18.59 quantity: 10 displayQty: 10 displayPrice: 18.59 workingPrice: 18.59 openCloseIndicator: Open orderType: LMT	• The option ID is the ID of the option as assigned by the exchange. • The Order ID is the ID of the order as assigned by the exchange, while the routed order ID is the order ID as defined by the member firm. • The origin code value of C represents that the order originated from a customer

#	Step	Reported Event	Comments
		timeInForce: DAY exchOriginCode: C coverage: Uncovered executingFirm: 999 nbbPrice: 18.58 nbbQty: 10 nboPrice: 18.60 nboQty: 10 member: Mem01	
3	Member firm sends in a request to change the timeInForce for the order from DAY to GTC		
4	An Option Order Modify Event is sent in to CAT from the exchange.	type: OOM exchange: Exch1 eventTimestamp: 20170116T143110.123456789 sequenceNumber: 912 optionID: 1208 orderID: 3312629458 coverage: Uncovered originalOrderID: 123456 initiator: Firm nbbPrice: 18.58 nbbQty: 10 nboPrice: 18.60 nboQty: 10 price: 18.59 quantity: 10 displayQty: 10 displayPrice: 18.59 workingPrice: 18.59 openCloseIndicator: Open orderType: LMT timeInForce: GTC exchOrigCode: C executingFirm: 999 member: Mem01 routedOrderId: 98766:	• Note that the inbound routedOrderId (Fix value ClOrdID Tag 11) sent in from the member firm is on the OOM event.

8.3.1.JSON Example

Simple Option Order Accepted Event

```
{
  "type": "OOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170116T143105.123456789",
  "sequenceNumber": 909,
```

```
  "optionID": "1208",
  "orderID": "123456",
  "routingParty": "FRMA",
  "routedOrderID": "98765",
  "session": "3",
  "side": "Buy",
  "price": 18.59,
  "quantity": 10,
  "displayQty": 10,
  "displayPrice": 18.59,
  "workingPrice": 18.59,
  "openCloseIndicator": "Open",
  "orderType": "LMT",
  "timeInForce": "DAY",
  "exchOriginCode": "C",
  "coverage": "Uncovered",
  "executingFirm": "999",
  "nbbPrice": 18.58,
  "nbbQty": 10,
  "nboPrice": 18.60,
  "nboQty": 10,
  "member": "Mem01"
}
```


Simple Option Order Modified Event


```
{
  "type": "OOM",
  "exchange": "Exch1",
  "eventTimestamp": "20170116T143110.123456789",
  "sequenceNumber": 912,
  "optionID": "1208",
  "orderID": "3312629458",
  "OriginalOrderID": 123456,
  "price": 18.59,
  "quantity": 10,
  "displayQty": 10,
  "displayPrice": 18.59,
  "workingPrice": 18.59,
  "openCloseIndicator": "Open",
  "orderType": "LMT",
  "timeInForce": "GTC",
  "exchOriginCode": "C",
  "coverage": "Uncovered",
  "executingFirm": "999",
  "nbbPrice": 18.58,
  "nbbQty": 10,
  "nboPrice": 18.60,
  "nboQty": 10,
  "member": "Mem01",
  "routedOrderId": "98766"
}
```

8.4. Simple Option Order Modified Event Created As a Result of Partial Execution at Away Exchange

This example shows how to populate the Simple Option Order Modify Event in the scenario where an order is routed to an away exchange. At the away exchange, the order is partially executed, leaving quantity returned to the exchange for the order. The Option Order Modify event is created showing this change in order quantity.

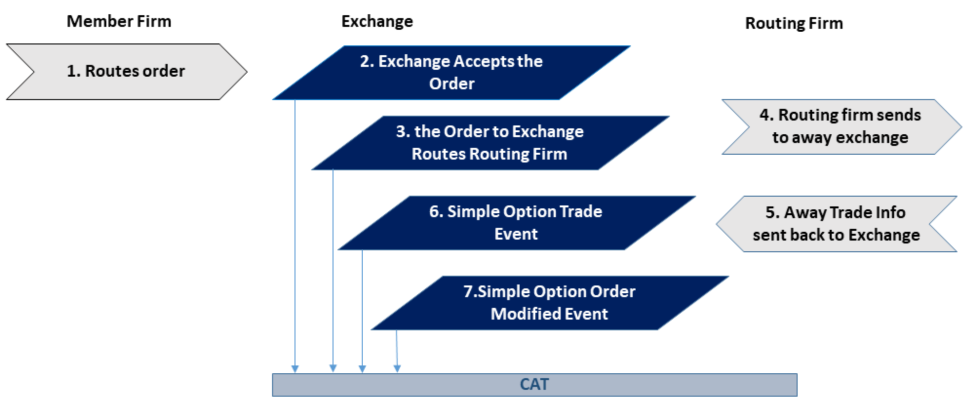

Figure 18: Simple Option Order Modification Event due to execution at away exchange

#	Step	Reported Event	Comments
1	Member firm sends option order to the exchange	NA	• The order is routed over session ID 3, with a price of 18.59, quantity of 10, for the option defined by the exchange as Option ID 1208
2.	Exchange 1 accepts the order and reports a Simple Option Order Accepted Event	**Simple Option Oder Accepted Event:** type: OOA exchange: Exch1 eventTimestamp: 20170116T143105.123456789 sequenceNumber: 909	• The option ID is the ID of the option as assigned by the exchange. • The Order ID is the ID of the order as assigned by the exchange, while the routed order ID is the order ID as defined by the member firm.

#	Step	Reported Event	Comments
		optionID: 1208 orderID: 123456 routingParty: FRMA routedOrderID: 98765 session: 3 side: Buy price: 18.59 quantity: 10 displayQty: 10 displayPrice: 18.59 workingPrice: 18.59 openCloseIndicator: Open orderType: LMT timeInForce: DAY exchOriginCode: C coverage: Uncovered executingFirm: 999 nbbPrice: 18.56 nbbQty: 10 nboPrice: 18.59 nboQty: 4 member: Mem01	• The origin code value of C represents that the order originated from a customer
3	Option order is routed to an exchange with a better Market	**OOR event** type: OOR exchange: Exch1 eventTimestamp: 20170116T143110.123456789 sequenceNumber: 911 optionID: 1208 orderID: 123456 routingParty: RoutingFirm routedOrderID: 4823326 session: 3 side: Buy price: 18.59 quantity: 10 displayQty: 10 orderType: LMT coverage: Uncovered timeInForce: DAY nbbPrice: 18.56 nbbQty: 10 nboPrice: 18.59 nboQty: 4 member: Mem01	• The order is routed to an exchange with a better offer
4	Routing Firm sends the order to the away exchange with a better market.		
5	Routing Firm returns a message with the remaining		

#	Step	Reported Event	Comments
	quantity on the order.		
6	Option order is partially executed at the away exchange, prompting an order trade event with the side routed away populated.	**Option Trade Event:** type: OT exchange: Exch1 eventTimestamp: 20170116T143111.123456789 sequenceNumber: 915 tradeID: 12345 optionID: 1208 quantity: 4 price: 18.59 nbbPrice: 18.56 nbbQty: 10 nboPrice: 18.59 nboQty: 4 Sell Side Details side: Sell executingFirm: 987 exchOriginCode: F member: BATS Buy Side Details side: Buy leavesQty: 6 openCloseIndicator: Open orderID: 123456 executingFirm: 551 exchOriginCode: C liquidityCode: Removed member: Mem01 routedOrderID: 4823326	• Quantity of 4 trades at the nbo price of 18.59 at the away exchange
7	Option Order Modified Event created to reflect the reduced quantity from an order executed at an away exchange	type: OOM exchange: Exch1 eventTimestamp: 20170116T143111.123456999 sequenceNumber: 920 optionID: 1208 orderID: 123456 coverage: Uncovered originalOrderID: 123456 initiator: Firm nbbPrice: 18.56 nbbQty: 10 nboPrice: 18.60 nboQty: 10 price: 18.59 quantity: 6 displayQty: 6 displayPrice: 18.59	• Option order modify event created for quantity change from 10 to 6. Note the routedOrderId is the routedOrderId sent from the exchange to the routing firm on the OOR event.

#	Step	Reported Event	Comments
		workingPrice: 18.59 openCloseIndicator: Open orderType: LMT timeInForce: DAY exchOrigCode: C member: Mem01 routedOrderId: 4823326:	

8.4.1. JSON Examples

Simple Option Order Accepted Event


```
{
  "type": "OOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170116T143105.123456789",
  "sequenceNumber": 909,
  "optionID": "1208",
  "orderID": "123456",
  "routingParty": "FRMA",
  "routedOrderID": "98765",
  "session": "3",
  "side": "Buy",
  "price": 18.59,
  "quantity": 10,
  "displayQty": 10,
  "displayPrice": 18.59,
  "workingPrice": 18.59,
  "openCloseIndicator": "Open",
  "orderType": "LMT",
  "timeInForce": "DAY",
  "exchOriginCode": "C",
  "coverage": "Uncovered",
  "executingFirm": "999",
  "nbbPrice": 18.56,
  "nbbQty": 10,
  "nboPrice": 18.59,
  "nboQty": 4,
  "member": "Mem01"
}
```


Option Order Route Event


```
{
  "type": "OOR",
  "exchange": "Exch1",
  "eventTimestamp": "20170116T143110.123456789",
  "sequenceNumber": 911,
  "optionID": "1208",
  "orderID": "123456",
```

```
  "routingParty": "RoutingFirm",
  "routedOrderID": "4823326",
  "session": "3",
  "side": "Buy",
  "price": 18.59,
  "quantity": 10,
  "displayQty": 10,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "coverage": "Uncovered",
  "executingFirm": "999",
  "nbbPrice": 18.56,
  "nbbQty": 10,
  "nboPrice": 18.59,
  "nboQty": 4,
  "member": "Mem01"
}
```

Option Trade Event

```
{
  "type": "OT",
  "exchange": "Exch1",
  "eventTimestamp": "20170116T143111.123456789",
  "sequenceNumber": 915,
  "tradeID": "12345",
  "optionID": "1208",
  "quantity": 4,
  "price": 18.59,
  "nbbPrice": 18.56,
  "nbbQty": 10,
  "nboPrice": 18.59,
  "nboQty": 4,
  "sellDetails": {
    "side": "Sell",
    "leavesQty": 6,
    "executingFirm": "987",
    "exchOriginCode": "F",
    "member": "BATS"
  },
  "buyDetails": {
    "side": "Buy",
    "leavesQty": 6,
    "openCloseIndicator": "Open",
    "orderID": "4823326",
    "executingFirm": "551",
    "exchOriginCode": "C",
    "liquidityCode": "Removed",
    "member": "Mem01"
  }
}
```

8.5. Simple Option Trade Event Examples

The below section will provide an example of a trade event for an option series where a broker order is executed against an existing market maker quote.

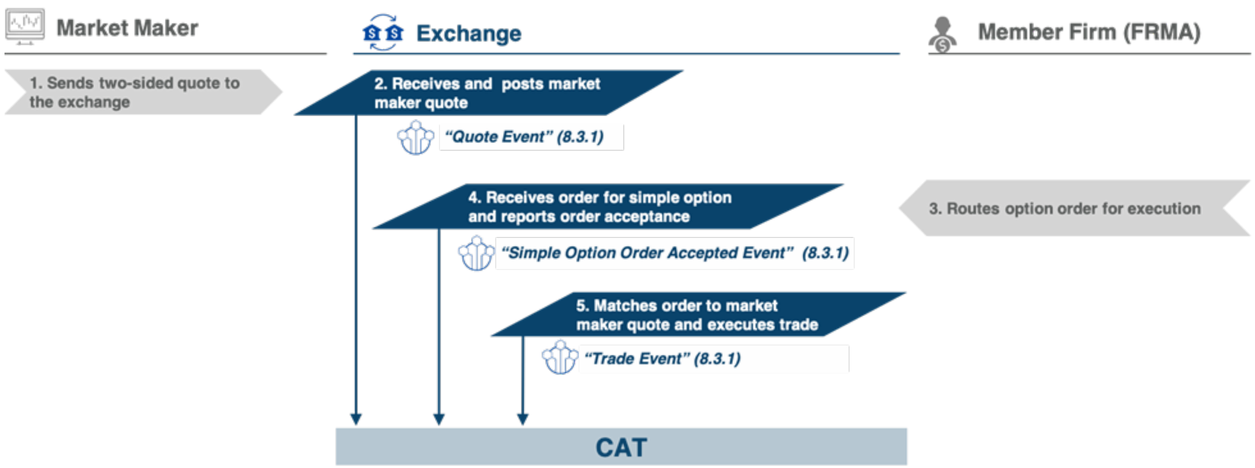

Figure 19: Simple Option Trade Event Example

Table 76: Simple Option Trade Event Example

#	Step	Reported Event	Comments
1	Market maker sends two-sided quote to the exchange	NA	• This scenario displays complete lifecycle of a simple options from Quote to Trade
2.	Exchange 1 posts the market maker quote	**Quote Event** type: OQ exchange: Exch1 sentTimestamp: 20170113T132036.123456 eventTimestamp: 20170113T132036.123486789 sequenceNumber: 1245 marketMaker: ABCD:A16 optionID: 6779 quoteID: Q9876 onlyOneQuote: true bidPrice: 2.40 bidQty: 10 askPrice: 2.43 askQty: 10	• The quote is a two-sided quote for an option with the ID: 6779 • The field market maker is the Member Alias assigned by the SRO to identify the market maker issuing the quote. In this case, the market maker ABCD has multiple users (e.g., acronyms used to differentiate users within the same MM), so the characters A16 denote the user or sub-account. • The sent timestamp denotes when the market maker sent the quote to the marketplace, while the

#	Step	Reported Event	Comments
			event timestamp is when the exchange received the quote
3	Member firm sends option order to the exchange	NA	• The order is routed over session ID 7, with a price of 2.43, quantity of 4, for the option defined by the exchange as Option ID 6779
4	Exchange 1 accepts the order and reports a Simple Option Order Accepted Event	**Simple Option Order Accepted Event:** type: OOA exchange: Exch1 eventTimestamp: 20170113T132209.123486789 sequenceNumber: 1300 optionID: 6779 orderID: 56789 routingParty: FRMA routedOrderID: 98654 session: 7 side: Buy price: 2.43 quantity: 4 displayQty: 4 displayPrice: 2.43 workingPrice: 2.43 openCloseIndicator: Open orderType: LMT timeInForce: DAY exchOriginCode: C coverage: Uncovered executingFirm: 999 nbbPrice: 2.40 nbbQty: 10 nboPrice: 2.43 nboQty: 10 member: Mem01	• The option ID is the ID of the option as assigned by the exchange. • The Order ID is the ID of the order as assigned by the exchange, while the routed order ID is the order ID as defined by the member firm. • The origin code value of C represents that the order originated from a customer
5	Exchange 1 matches order to market maker quote and executes trade	**Option Trade Event:** type: OT exchange: Exch1 eventTimestamp: 20170113T132211.123456789 sequenceNumber: 1421 tradeID: 12345 optionID: 6779 quantity: 4 price: 2.43 nbbPrice: 2.42 nbbQty: 10 nboPrice: 2.43	

#	Step	Reported Event	Comments
		nboQty: 10 saleCondition: "O " Sell Side Details side: Sell leavesQty: 6 quoteID: Q9876 executingFirm: 987 mktMkrSubAccount: ABC123 exchOriginCode: M liquidityCode: Added member: ABCD:A16 Buy Side Details side: Buy leavesQty: 0 openCloseIndicator: Open orderID: 56789 executingFirm: 999 exchOriginCode: C liquidityCode: Removed member: Mem01	

8.5.1.JSON Examples

Quote Event


```
{
    "type": "OQ",
    "exchange": "Exch1",
    "sentTimestamp: "20170113T132036.123456",
    "eventTimestamp: "20170113T132036.123486789",
    "sequenceNumber": 1245,
    "marketMaker": "ABCD:A16",
    "optionID": "6779",
    "quoteID": "Q9876",
    "onlyOneQuote": true,
    "bidPrice": 2.40,
    "bidQty": 10,
    "askPrice": 2.43,
    "askQty": 10
}
```


Simple Option Order Accepted Event


```
{
  "type": "OOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170113T132209.123486789",
  "sequenceNumber": 1300,
```

```
  "optionID": "6779",
  "orderID": "56789",
  "routingParty": "FRMA",
  "routedOrderID": "98654",
  “session": "7",
  "side": "Buy",
  "price": 2.43,
  "quantity": 4,
  "displayQty": 4,
  "displayPrice": 2.43,
  "workingPrice": 2.43,
  "openCloseIndicator": "Open",
  "orderType": "LMT",
  "timeInForce": "DAY",
  "exchOriginCode": "C",
  "coverage": "Uncovered",
  "executingFirm": "999",
  "nbbPrice": 2.40,
  "nbbQty": 10,
  "nboPrice": 2.43,
  "nboQty": 10,
  "member": "Mem01"
}
```


Option Trade Event


```
{
  "type": "OT",
  "exchange": "Exch1",
  "eventTimestamp": "20170113T132211.123456789",
  "sequenceNumber": 1421,
  "tradeID": "12345",
  "optionID": "6779",
  "quantity": 4,
  "price": 2.43,
  "nbbPrice": 2.42,
  "nbbQty": 10,
  "nboPrice": 2.43,
  "nboQty": 10,
  "saleCondition": "O ",
  "sellDetails": {
    "side": "Sell",
    "leavesQty": 6,
    "quoteID": "Q9876",
    "executingFirm": "987",
    "mktMkrSubAccount": "ABC123",
    "exchOriginCode": "M",
    "liquidityCode": "Added",
    "member": "ABCD:A16",
    "executionCodes":{
            "INTLIQ":"A",
            "SUBLIQ":"S"
            }


  },
```

```
  "buyDetails": {
    "side": "Buy",
    "leavesQty": 0,
    "openCloseIndicator": "Open",
    "orderID": "56789",
    "executingFirm": "999",
    "exchOriginCode": "C",
    "liquidityCode": "Removed",
    "member": "Mem01",
    "executionCodes":{
            "INTLIQ":"A",
            "SUBLIQ":"S"
            }

  }
}
```


Example CSV Corresponding – Options Trade Event:


```
OT,Exch1,20170113T132211.123456789,1421,,12345,6779,4,2.43,2.42,10,2.43,10, O
,,Buy,0,Open,,56789,999,,,,C,Removed,INTLIQ=A|SUBLIQ=S,Mem01,Sell,6,,
Q9876,,987,,,ABC123,M,Added,INTLIQ=A|SUBLIQ=S,ABCD:A16
```


8.6. Complex Options Trade Events Examples

In all cases, complex option trades are reported to CAT only at the leg level. There is no roll-up trade reported at the complex order level. For example, an order on the complex option (ID 9851) below would have had corresponding orders reported to CAT for each of the underlying legs. As the following examples will show, trades on this complex option will report by leg, with each leg trade event corresponding to an order event on the leg that is in turn attached to a parent-level complex order event.

Complex Option – optionID: 9851

Table 77: Complex Options Trade Events Example: Legs

optionID	side	ratio	primaryDeliverable	expirationDate	strikePrice	putCall	exerciseStyle	settlement
1491	Buy	1	XYZZY	21 Oct 2017	30.00	C	American	PM
1492	Sell	1	XYZZY	21 Oct 2017	32.50	C	American	PM
XYZZY	Buy	100						

This section follows a series of trade events on the complex option described above, along with examples of the quotes and orders that would be referenced in those trades.

- A new market maker quote is posted for the option leg 1491
- A new market maker quote is posted for the option leg 1492

- An order is placed for quantity 10 of the complex option 9851
- A trade on the first option leg 1491 is reported (10 contracts)
- A trade on the second option leg 1492 is reported (10 contracts)
- A fill on the stock leg XYZZY is reported (1,000 shares)

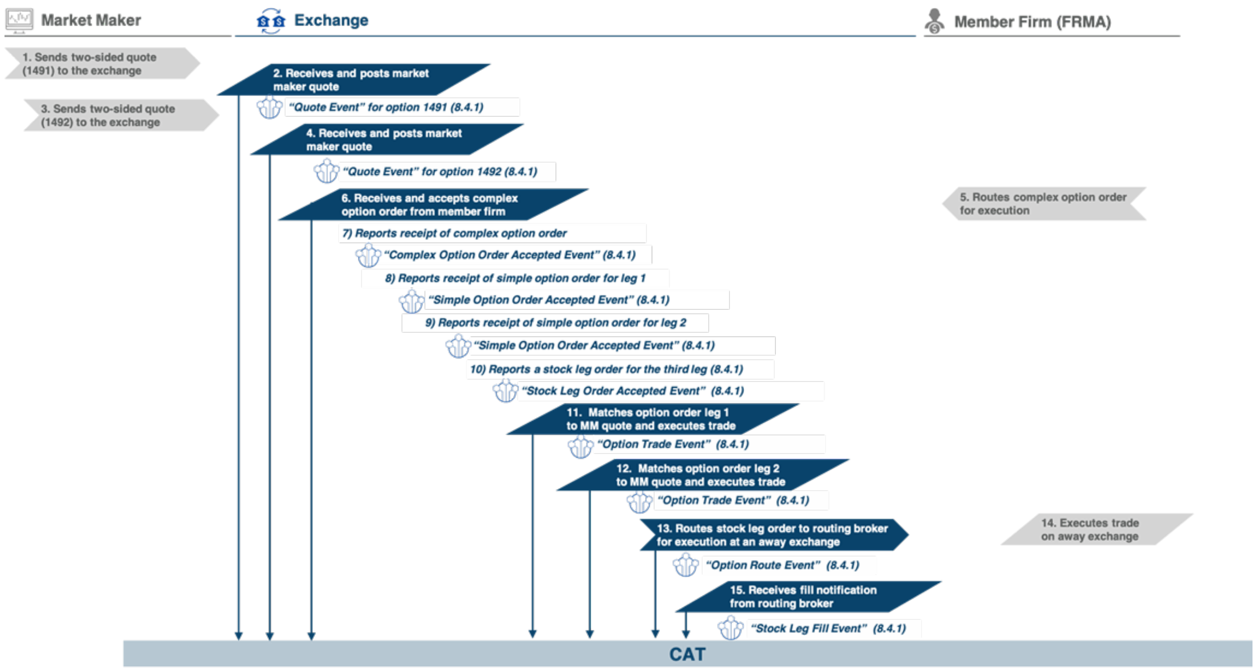

Figure 20: Complex Options Trade Events Example

Table 78: Complex Options Trade Events Example

#	Step	Reported Event	Comments
1	Market maker sends two-sided quote to the exchange	NA	• Quote is for the option the exchange identifies as option ID 1491
2.	Exchange 1 posts the market maker quote	**Quote Event** type: OQ exchange: Exch1 sentTimestamp: 20170420T142036.123456 eventTimestamp: 20170420T142036.123486789 sequenceNumber: 1112 marketMaker: ABCD:AA optionID: 1491	• The quote is a two-sided quote for an option with the option ID: 1491 • The field market maker is the Member Alias assigned by the SRO to identify the market maker issuing the quote. In this case, the market maker ABCD has

#	Step	Reported Event	Comments
		quoteID: 12345 onlyOneQuote: true bidPrice: 1.90 bidQty: 10 askPrice: 2.00 askQty: 10	multiple users (e.g., acronyms used to differentiate users within the same MM), so the characters AA denote the user or sub-account. • The sent timestamp denotes when the market maker sent the quote to the marketplace, while the event timestamp is when the exchange received the quote
3	Market maker sends two-sided quote to the exchange	NA	• Quote is for the option the exchange identifies as option ID 1492
4	Exchange 1 posts the market maker quote	**Quote Event** type: OQ exchange: Exch1 sentTimestamp: 20170420T142036.124456 eventTimestamp: 20170420T142036.124486789 sequenceNumber: 1125 marketMaker: ABCD:AA mktMkrSubAccount: A16 optionID: 1492 quoteID: 67890 onlyOneQuote: true bidPrice: 1.00 bidQty: 10 askPrice: 1.10 askQty: 10	• The quote is a two-sided quote for an option with the ID: 1492 • The field market maker is the Member Alias assigned by the SRO to identify the market maker issuing the quote. In this case, the market maker ABCD has multiple users (e.g., acronyms used to differentiate users within the same MM), so the characters A16 denote the user or sub-account. • The sent timestamp denotes when the market maker sent the quote to the marketplace, while the event timestamp is when the exchange received the quote
5	Member Firm (FRMA) sends complex option order to the exchange	NA	• The order is routed over session ID 7, with a price of -30.90, quantity of 10, for the option defined by the exchange as Option ID 9851

#	Step	Reported Event	Comments
6	Exchange 1 accepts the complex option order	*Shown in steps 7, 8, and 9*	
7	Exchange 1 reports a complex option order accepted event	**Complex Option Order Accepted Event** type: OCOA exchange: Exch1 eventTimestamp: 20170420T142411.121456789 sequenceNumber: 909 optionID: 9851 orderID: 8473692 side: AsDirected routingParty: FRMA routedOrderID: 4567123 session: 7 price: -30.90 quantity: 10 timeInForce: DAY member: Mem01	• The option ID is the ID of the complex option as assigned by the exchange. • The Order ID is the ID of the order as assigned by the exchange, while the routed order ID is the order ID as defined by the member firm.
8	Exchange 1 reports a simple option order accepted event for the first leg	**Simple Option Order Accepted Event** type: OOA exchange: Exch1 eventTimestamp: 20170420T142411.121456790 sequenceNumber: 909 optionID: 1491 orderID: 84736921 side: Buy quantity: 10 displayQty: 0 openClose: Open orderType: LEG timeInForce: DAY exchOriginCode: C coverage: Uncovered executingFirm: 999 complexOrderID: 8473692 complexOptionID: 9851 nbbPrice: 1.90 nbbQty: 10 nboPrice: 2.00 nboQty: 10 member: Mem01	• This section describes the Simple Option Order Accepted Event for Leg 1 corresponding to the complex option order described above. • Note that in this Simple Option Order Accepted Event for Leg 1, the Routed Order ID is the same as reported in the parent complex order, however, the order ID for this leg is unique.
9	Exchange 1 reports a simple option order accepted event for the second leg	**Simple Option Order Accepted Event** type: OOA exchange: Exch1 eventTimestamp: 20170420T142411.121456791	• This section describes the Simple Option Order Accepted Event for Leg 2 corresponding to the complex

#	Step	Reported Event	Comments
		sequenceNumber: 909 optionID: 1492 orderID: 84736922 side: Sell quantity: 10 displayQty: 0 openClose: Open orderType: LEG timeInForce: DAY exchOriginCode: C coverage: Uncovered executingFirm: 999 complexOrderID: 8473692 complexOptionID: 9851 nbbPrice: 1.00 nbbQty: 10 nboPrice: 1.10 nboQty: 10 member: Mem01	option order described above. • Note that in this Simple Option Order Accepted Event for Leg 2, the Routed Order ID is the same as reported in the parent complex order, however, the order ID for this leg is unique.
10	Exchange 1 reports a stock leg order accepted event for the third leg	**Stock Leg Order Accepted Event:** type: OSL exchange: Exch1 eventTimestamp: 20170420T142411.121456793 sequenceNumber: 909 symbol: XYZZY orderID: 84736923 side: Buy price: 29.90 quantity: 1000 displayQty: 0 orderType: LMT timeInForce: DAY clearingFirm: FRMA complexOrderID: 8473692 complexOptionID: 9851 nbbPrice: 29.84 nbbQty: 10 nboPrice: 29.90 nboQty: 10 member: Mem01	• This section describes the Stock Leg Order Accepted Event for Leg 3 corresponding to the complex option order described above.
11	Exchange 1 matches order for leg 1 to a market maker quote and executes trade	**Option Trade Event:** type: OT exchange: Exch1 eventTimestamp: 20170420T142411.123456795 sequenceNumber: 456 tradeID: 194378 optionID: 1491	• This event describes a trade on the first leg (option 1491) of the complex option 9851. In this case, the trade event fills all of the (buy) quantity requested by the order, and all of the (sell) quantity

#	Step	Reported Event	Comments
		quantity: 10 price: 2.00 nbbPrice: 1.90 nbbQty: 10 nboPrice: 2.00 nboQty: 10 saleCondition: O Sell Side Details side: Sell leavesQty: 0 quoteID: 12345 executingFirm: 987 mktMkrSubAccount: ABC123 exchOriginCode: M liquidityCode: Added member: ABCD:AA Buy Side Details side: Buy leavesQty: 0 openCloseIndicator: Open orderID: 84736921 executingFirm: 999 exchOriginCode: C liquidityCode: Removed member: Mem01	offered by the market maker. Note that the order for the first option leg (created as a result of the complex order) is referenced in the buy side details, while the market maker quote for the underlying option (1491) of the first leg is referenced in the sell side details.
12	Exchange 1 matches order for leg 2 to a market maker quote and executes trade	**Option Trade Event:** type: OT exchange: Exch1 eventTimestamp: 20170420T142411.123456796 sequenceNumber: 1209 tradeID: 194379 optionID: 1492 quantity: 10 price: 1.00 nbbPrice: 1.00 nbbQty: 10 nboPrice: 1.10 nboQty: 10 saleCondition: O Sell Side Details side: Sell leavesQty: 0 openCloseIndicator: Open orderID: 84736922 executingFirm: 999 exchOriginCode: C liquidityCode: Removed	• This event describes a trade on the second leg (option 1492) of the complex option 9851. Similarly, this trade event fills all of the (sell) quantity of the leg order generated as a result of the complex order. This trade has executed in ratio, as defined in complex option, to the trade on the first leg. Note that on this leg, the broker who placed the order is on the sell side, while the market maker is on the buy side.

#	Step	Reported Event	Comments
		member: Mem01 Buy Side Details side: Buy leavesQty: 0 quoteID: 67890 executingFirm: 987 mktMkrSubAccount: ABC123 exchOriginCode: M liquidityCode: Added member: ABCD:AA	
13	Exchange 1 routes stock leg order to the routing broker for execution on an away exchange	**Option Route Event** type: OOR exchange: Exch1 eventTimestamp: 20170420T142411.121656785 sequenceNumber: 2059 symbol: XYZZY orderID: 84736923 routingParty: FRMC routedOrderID: 8999999 session: 9 side: Buy price: 29.90 quantity: 1000 displayQty: 0 orderType: LMT coverage: Uncovered timeInForce: DAY result: ACK resultTimestamp: 20170420T142411.122656789 nbbPrice: 29.84 nbbQty: 10 nboPrice: 29.90 nboQty: 10 complexOrderID: 8473692 complexOptionID: 9851 member: Mem01	• This event describes a route on the stock leg (Symbol = XYZZY) of the complex option 9851 to a routing broker for execution on an away exchange.
14	Routing broker routes to the away exchange, and receives a fill report when the order executes		
15	Exchange 1 receives fill notification from the routing broker	**Stock Leg Fill Event** type: OSLF exchange: Exch1 eventTimestamp: 20170420T142412.125656789	

#	Step	Reported Event	Comments
		sequenceNumber: 2088 fillID: 95321 symbol: XYZZY quantity: 1000 price: 29.90 saleCondition: OB side: Buy leavesQty: 0 orderID: 84736923 clearingFirm: FRMA clearingNumber: 123 member: Mem01	

8.6.1.JSON Examples

Quote Event (Step 2)


```
{
    "type": "OQ",
    "exchange": "Exch1",
    "sentTimestamp: "20170420T142036.123456",
    "eventTimestamp: "20170113T142036.123486789",
    "sequenceNumber": 1112,
    "marketMaker": "ABCD:AA",
    "optionID": "1491",
    "quoteID": "12345",
    "onlyOneQuote": true,
    "bidPrice": 1.90,
    "bidQty": 10,
    "askPrice": 2.00,
    "askQty": 10
}
```


Quote Event (Step 4)


```
{
    "type": "OQ",
    "exchange": "Exch1",
    "sentTimestamp: "20170420T142036.124456",
    "eventTimestamp: "20170113T142036.124486789",
    "sequenceNumber": 1125,
    "marketMaker": "ABCD:AA",
    "optionID": "1492",
    "quoteID": "67890",
    "onlyOneQuote": true,
    "bidPrice": 1.00,
    "bidQty": 10,
    "askPrice": 1.10,
    "askQty": 10
}
```

Complex Option Order Accepted Event (Step 7)


```
{
  "type": "OCOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142411.121456789",
  "sequenceNumber": 909,
  "optionID": "9851",
  "orderID": "8473692",
  "side": "AsDirected",
  "routingParty": "FRMA",
  "routedOrderID": "4567123",
  "session": "7",
  "price": -30.90,
  "quantity": 10,
  "timeInForce": "DAY",
  "member": "Mem01"
}
```


Simple Option Order Accepted Event (Step 8)


```
{
  "type": "OOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142411.121456789",
  "sequenceNumber": 909,
  "optionID": "1491",
  "orderID": "84736921",
  "side": "Buy",
  "quantity": 10,
  "displayQty": 0,
  "openCloseIndicator": "Open",
  "orderType": "LEG",
  "timeInForce": "DAY",
  "exchOriginCode": "C",
  "coverage": "Uncovered",
  "executingFirm": "999",
  "nbbPrice": 1.90,
  "nbbQty": 10,
  "nboPrice": 2.00,
  "nboQty": 10,
  "complexOrderID": "8473692",
  "complexOptionID": "9851",
  "member": "Mem01"
}
```


Simple Option Order Accepted Event (Step 9)


```
{
  "type": "OOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142411.121456789",
  "sequenceNumber": 909,
  "optionID": "1492",
  "orderID": "84736922",
```

```
  "side": "Sell",
  "quantity": 10,
  "displayQty": 0,
  "openCloseIndicator": "Open",
  "orderType": "LEG",
  "timeInForce": "DAY",
  "exchOriginCode": "C",
  "coverage": "Uncovered",
  "executingFirm": "999",
  "nbbPrice": 1.00,
  "nbbQty": 10,
  "nboPrice": 1.10,
  "nboQty": 10,
  "complexOrderID": "8473692",
  "complexOptionID": "9851",
  "member": "Mem01"
}
```


Stock Leg Order Accepted Event (Step 10)


```
{
  "type": "OSL",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142411.121456789",
  "sequenceNumber": 909,
  "symbol": "XYZZY",
  "orderID": "84736923",
  "side": "Buy",
  "price": 29.90,
  "quantity": 1000,
  "displayQty": 0,
  "orderType": "LMT",
  "timeInForce": "DAY",
  "clearingFirm": "FRMA",
  "nbbPrice": 29.84,
  "nbbQty": 10,
  "nboPrice": 29.90,
  "nboQty": 10,
  "complexOrderID": "8473692",
  "complexOptionID": "9851",
  "member": "Mem01"
}
```


Option Trade Event (Step 11)


```
{
  "type": "OT",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142411.123456789",
  "sequenceNumber": 456,
  "tradeID": "194378",
  "optionID": "1491",
  "quantity": 10,
  "price": 2.00,
  "nbbPrice": 1.90,
```

```
  "nbbQty": 10,
  "nboPrice": 2.00,
  "nboQty": 10,
  "saleCondition": "O ",
  "sellDetails": {
    "side": "Sell",
    "leavesQty": 0,
    "quoteID": "12345",
    "executingFirm": "987",
    "mktMkrSubAccount": "ABC123",
    "exchOriginCode": "M",
    "liquidityCode": "Added",
    "member": "ABCD:AA"
  },
  "buyDetails": {
    "side": "Buy",
    "leavesQty": 0,
    "openCloseIndicator": "Open",
    "orderID": "84736921",
    "executingFirm": "999",
    "exchOriginCode": "C",
    "liquidityCode": "Removed",
    "member": "Mem01"
  }
}
```

Option Trade Event (Step 12)

```
{
  "type": "OT",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142411.123456789",
  "sequenceNumber": 1209,
  "tradeID": "194379",
  "optionID": "1492",
  "quantity": 10,
  "price": 1.00,
  "nbbPrice": 1.00,
  "nbbQty": 10,
  "nboPrice": 1.10,
  "nboQty": 10,
  "saleCondition": "O",
  "sellDetails": {
    "side": "Sell",
    "leavesQty": 0,
    "orderID": "84736922",
    "openCloseIndicator": "Open",
    "executingFirm": "999",
    "exchOriginCode": "C",
    "liquidityCode": "Removed",
    "member": "Mem01",
    "executionCodes" :{
            "INTLIQ":"A",
            "SUBLIQ":"S"
            }
```

```
  },
  "buyDetails": {
    "side": "Buy",
    "leavesQty": 0,
    "quoteID": "67890",
    "executingFirm": "987",
    "mktMkrSubAccount": "ABC123",
    "exchOriginCode": "M",
    "liquidityCode": "Added",
    "member": "ABCD:AA",
    "executionCodes":{
            "INTLIQ”: "A",
            "SUBLIQ": "S"
            }}
}
```
Example CSV Corresponding to Step 12 – Options Trade Event:

```
OT,Exch1,20170420T142411.123456789,1209,,194379,1492,10,1.0,1.0,10,1.10,10,,
,Buy,0,Open,,84736922,999,,,,C,Removed,INTLIQ=A|SUBLIQ=S,Mem01,Sell,0,,
67890,,687,,,,M,Added,INTLIQ=R|SUBLIQ=S,ABCD:AA
```

Option Route Event (Step 13)

```
{
  "type": "OOR",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142411.121656789",
  "sequenceNumber": 2059,
  "symbol": "XYZZY",
  "orderID": "84736923",
  "routingParty": "FRMC",
  "routedOrderID": "8999999",
  "session": "9",
  "side": "Buy",
  "price": 29.90,
  "quantity": 1000,
  "displayQty": 0,
  "orderType": "LMT",
  "coverage": "Uncovered",
  "timeInForce": "DAY",
  "result": "ACK",
  "resultTimestamp": "20170420T142411.122656789",
  "nbbPrice": 29.84,
  "nbbQty": 10,
  "nboPrice": 29.90,
  "nboQty": 10,
  "complexOrderID": "8473692",
  "complexOptionID": "9851",
  "member": "Mem01"
}
```

Stock Leg Fill Event (Step 14)

```
{
```

```
  "type": "OSLF",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142412.125656789",
  "sequenceNumber": 2088,
  "fillID": "95321",
  "symbol": "XYZZY",
  "quantity": 1000,
  "price": 29.90,
  "saleCondition": "OB",
  "side": "Buy",
  "leavesQty": 0,
  "orderID": "84736923",
  "clearingFirm": "FRMA",
  "clearingNumber": "123",
  "member": "Mem01"
}
```


8.7. Complex Option Order Modify Event Example

This example shows how to populate the Complex Option Order Modify Event with the routedOrderId because of a firm change to the order.

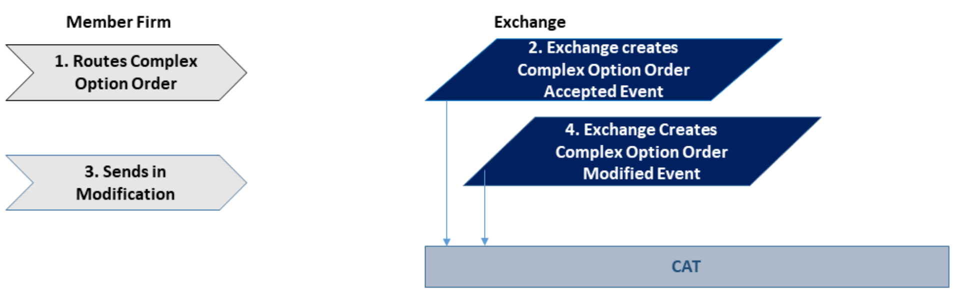

Figure 21: Complex Option Modify Event Example

#	Step	Reported Event	Comments
1	Member firm sends complex option order to the exchange	NA	
2.	Exchange 1 accepts the order and reports a Simple Complex Option Order Accepted Event	**Complex Option Order Accepted Event** type: OCOA exchange: Exch1 eventTimestamp: 20170420T142411.121456789 sequenceNumber: 909 optionID: 9851 orderID: 8473692	• The legs would be represented in OOA events as shown in example 8.9

#	Step	Reported Event	Comments
		side: AsDirected routingParty: FRMA routedOrderID: 4567123 session: 7 price: -30.90 quantity: 10 timeInForce: DAY member: Mem01	
3	Member firm sends in a new routedOrderId modifying the timeInForce value to "GTC". A complex order modify event is created to represent this scenario	NA	
4	The Exchange submits a Complex Option Order Modified Event to CAT.	**Complex Option Oder Modified Event:** type: OCOM exchange: Exch1 eventTimestamp: 20170420T142415.121456789 sequenceNumber: 922 optionID: 9851 orderID: 5790176 originalOrderID: 8473692 initiator: "Firm" price: -30.9 quantity: 10 leavesQty: 10 timeInForce "GTC" member: Mem01 routedOrderID = 4567124	• The order was modified by the firm to change from a DAY order to a GTC order. Note that the inbound routedOrderId (Fix value ClOrdID Tag 11) sent in from the member firm is on the OCOM event.

8.7.1.JSON Examples

<u>Complex Option Order Accepted Event (Step 7)</u>


```
{
  "type": "OCOA",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142411.121456789",
  "sequenceNumber": 909,
  "optionID": "9851",
  "orderID": "8473692",
  "side": "AsDirected",
  "routingParty": "FRMA",
  "routedOrderID": "4567123",
```

```
  "session": "7",
  "price": -30.90,
  "quantity": 10,
  "timeInForce": "DAY",
  "member": "Mem01"
}

{
  "type": "OCOM",
  "exchange": "Exch1",
  "eventTimestamp": "20170420T142415.121456789",
  "sequenceNumber": 922,
  "optionID": "9851",
  "orderID": "5790176",
  "originalOrderID: 8473692
  "routedOrderID": "4567124",
  "price": -30.90,
  "quantity": 10,
  "timeInForce": "GTC",
  "member": "Mem01"
}
```

9. Submission Process

This section has been removed for security purposes.

10. Feedback and Corrections

This section has been removed for security purposes.

11. Testing

This section has been removed for security purposes.

12. Additional Information

Additional information is available from the CAT Public Website or the Service Desk. Details are provided below.

12.1. Public Website

Public Website (http://www.catnmsplan.com) is to provide primary information about CAT. The content includes: Link to SEC Rule 613, Press Releases, Technical Specifications, User Manuals, FAQs, Training Materials and Contact info.

12.2. FINRA CAT Help Desk

The FINRA CAT Helpdesk is the primary source for answers to questions about CAT, including questions regarding: clock synchronization, firm reporting responsibilities, interpretive questions, technical specifications for reporting to CAT and more. The FINRA CAT Helpdesk can be reached by phone at 888-696-3348 or e-mail at help@finracat.com.

Appendices

Appendix A. Clock Synchronization Requirement

In previous sections, details are described regarding the Order Events and data elements. Timestamp, as one of the required data elements for each order event, must be correctly recorded by Participants at a predefined granularity. This section provides detailed requirements and a recommended approach on how Participants should manage clock synchronization.

In order to comply with CAT NMS Plan requirements of Clock Synchronization and correctly record the Timestamp fields for order events, Participants are required synchronize Business Clocks, at a minimum, to within 100 microseconds of the time maintained by the National Institute of Standards and Technology (NIST).

The tolerance includes:

- Difference between the NIST standard and a time provider's clock;
- Transmission delay from the source; and
- Amount of drift in the Participant's clock.

In order to ensure the accuracy of timestamps for Reportable Events, Participants are anticipated to adopt policies and procedures to verify such required synchronization each Trading Day (1) before the market opens, and (2) periodically throughout the Trading Day. Participants are recommended to keep documentation which provides details of their Business Clock synchronization process, and the resulting log files from the implementation of such processes.

Any time provider and technology may be used for clock synchronization as long as the Business Clocks are in compliance with the accuracy requirement.

If additional details are needed, please refer to the Clock Synchronization User Guide to be published separately.

Note: The tolerance for clock synchronization does not impact the amount of time allowed for CAT reporting. CAT does NOT require reporters to report order information within 100 microseconds of receiving an order.

Appendix B. Error Codes

An error code is a machine-parseable description of why a file or record was rejected. This differs from an error description, which is intended for human consumption. The following is a listing of errors codes for data ingestion and linkage processing.

B.1. Data Ingestion Errors

The tables below contain error messages that are associated with Data Ingestion.

Table 79: Ingestion Error Codes

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OE.INGEST,MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.010	Required field is missing	Required field is missing	ERROR
2	OE.INGEST,MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.020	Numeric value expected	Bad value for a numeric data type	ERROR
3	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.030	Integer Value expected	Bad value for an integer	ERROR
4	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.040	Unsigned Integer expected	Bad value for an unsigned integer	ERROR
5	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.050	Boolean Value expected	Bad value for a Boolean	ERROR
6	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.060	Invalid JSON format	Not in JSON format	ERROR
7	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.070	Exceeds maximum length of field	Invalid character length of a text or alpha numeric data type	ERROR
8	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.080	Invalid Timestamp Format	Invalid timestamp format	ERROR
9	FT.INGEST, RME.INGEST	.085	Invalid Time Format	Invalid time format	ERROR
10	OE.INGEST, MD.REC, OD.REC, MMD.REC,	.090	Invalid Date	Invalid date format	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
	FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST		Format		
11	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME. INGEST, BBO.INGEST	.100	Value is not listed as a valid choice	Value is not listed as a valid choice	ERROR
12	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.110	JSON record has invalid Field Name	JSON record has invalid Field Name	ERROR
13	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.120	Text or alphanumeric type has an illegal character	Text or alphanumeric type has an illegal character	ERROR
14	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME. INGEST, BBO.INGEST	.130	Invalid name value pair data	NameValue value does not follow documented format	ERROR
15	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME. INGEST, BBO.INGEST	.140	Numeric value is missing required fraction digits	Numeric value is missing required fraction digits	ERROR
16	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME. INGEST, BBO.INGEST	.150	Numeric value is missing required whole digits	Numeric value is missing required whole digits	ERROR
17	OE.INGEST, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.160	CSV record has invalid number of fields	When, for tokenizing, a CSV line results in less tokens than required.	ERROR
18	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.170	Number of rows with errors exceed 10% of Row Count of Data File	Number of errors exceed 10% of Row Count of Data File	ERROR
19	OE.INGEST, FT.INGEST, RME.INGEST, BBO.INGEST	.180	Record Offset referenced in Correction file is Invalid	Record Offset referenced in Correction file is Invalid	ERROR
20	OE.COUNT, MD.COUNT, OD.COUNT, MMD.COUNT, FT.COUNT, OTH.COUNT, RME.INGEST, BBO.INGEST	.190	File Record Count does not match metadata	File Record Count does not match metadata	ERROR
21	INT.META	.200	Mismatch in meta file	Mismatch in Metadata as compared to file name	ERROR
22	OE.INGEST, FT.INGEST, RME.INGEST, BBO.INGEST	.210	Invalid Symbol	Equity Symbol is Incorrect	WARNING
23	OE.INGEST, MD.REC, MMD.REC, RME.INGEST, BBO.INGEST	.220	Invalid Member ID	Member ID is invalid	WARNING
24	FILE.NAME	.230	File Name is	File Name is Invalid (Invalid Format, Duplicate, File Name	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
			Invalid	too Long, File Name for future date)	
25	INT.META	.240	Replacement File Not Permitted	A replacement file for OrderEvents, FinraTransactions, and RejectMessageEvents file kinds is not permitted after T+4 @ 8:00 a.m. ET.	ERROR
26	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.250	Unknown message type	The message type specified in the record is unknown	ERROR
27	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME. INGEST, BBO.INGEST	.260	Sub-fields in Name/Value have value errors	Sub-fields in Name/Value have value errors	ERROR
28	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME. INGEST, BBO.INGEST	.270	Invalid array value for declared type	Invalid array value for declared type	ERROR
29	OE.INGEST, MD.REC, OD.REC, MMD.REC, FT.INGEST, OTH.REC, RME.INGEST, BBO.INGEST	.280	Expected CSV format is invalid	Expected CSV format is invalid	ERROR
30	OE.INGEST, RME. INGEST	.290	Numeric value is out of range	Numeric value is out of the allowed range as defined by the Data Dictionary	ERROR
31	FILE.TIMEOUT	.1050	Time out waiting for meta file		ERROR
32	FILE.TIMEOUT	.1060	Time out waiting for data file		ERROR
33	FILE.NAME	.1070	File is not compressed		ERROR

Table 80: Conditional Validation Error Codes

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OE.INGEST, RME.INGEST	.2000	Invalid reporter	*Event(s): All Equity and Options Events, RME* `reporter` on the event must match the CAT Reporter ID in the file name	ERROR
2	OE.INGEST	.2010	Missing price	*Event(s): EOA, EOR, EIR, EMR, EORS* `price` must be provided and greater than or equal to zero when `orderType` indicates a Limit order. *Event(s): EOM*	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
				`price` must be provided and greater than zero when `orderType` indicates a Limit order and `initiator` is 'Firm' or 'MarketMaker'.	
3	OE.INGEST	.2020	Missing or Invalid displayPrice	*Event(s): EOA, EOM, EOJ, EORS* `displayPrice` must be provided and greater than or equal to zero when `displayQty` is provided and is greater than zero.	ERROR
4	OE.INGEST	.2030	Missing routedOrderID	*Events: EOM, EOJ* `routedOrderID` must be provided when `initiator` is 'Firm' or 'MarketMaker'.	ERROR
5	OE.INGEST	.2040	Missing routingParty	*Events: EOM, EOJ* `routingParty` must be provided when `initiator` is 'Firm' or 'MarketMaker'.	ERROR
6	OE.INGEST	.2050	Missing session	*Events: EOM, EOJ* `session` must be provided when `initiator` is 'Firm' or 'MarketMaker'.	ERROR
7	OE.INGEST	.2060	Missing side	*Events: EOM, EOJ* `side` must be provided when `initiator` is 'Firm' or 'MarketMaker'.	ERROR
8	OE.INGEST	.2070	Missing displayQty	*Events: EOJ* `displayQty` must be provided when `displayPrice` is provided.	ERROR
9	OE.INGEST	.2080	Missing quantity	*Events: EOJ* `quantity` must be provided when `initiator` is 'Firm' or 'MarketMaker'.	ERROR
10	OE.INGEST	.2090	Invalid orderID	*Events: EOT* `buyDetails.orderID` must be provided when and only when the 'NOBUYID' does not exist in any `executionCodes` field for the event (including `executionCodes`, `buyDetails.executionCodes`, and `sellDetails.executionCodes`). `sellDetails.orderID` must be provided when and only when the 'NOSELLID' does not exist in any `executionCodes` field for the event (including `executionCodes`, `buyDetails.executionCodes`, and	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
				sellDetails.executionCodes).	
11	OE.INGEST	.2100	Missing side	*Events: EOT* buyDetails.side must be provided if buyDetails.orderID is provided. sellDetails.side must be provided if sellDetails.orderID is provided.	ERROR
12	OE.INGEST	.2110	Missing member	*Events: EOT* buyDetails.member must be provided if buyDetails.orderID is provided. sellDetails.member must be provided if sellDetails.orderID is provided.	ERROR
13	OE.INGEST	.2120	Missing capacity	*Events: EOT* buyDetails.capacity must be provided if buyDetails.orderID is provided. sellDetails.capacity must be provided if sellDetails.orderID is provided.	ERROR
14	OE.INGEST	.2130	Missing clearingNumber	*Events: EOT* buyDetails.clearingNumber must be provided if buyDetails.orderID is provided. sellDetails.clearingNumber must be provided if sellDetails.orderID is provided.	ERROR
15	OE.INGEST	.2140	Missing quoteID or askQuoteID	*Events: OQ, OQC* At least one of quoteID or askQuoteID must be provided when onlyOneQuote flag is 'False'	ERROR
16	OE.INGEST	.2150	Missing bidPrice and askPrice	*Events: OQ* At least one of bidPrice or askPrice must be provided.	ERROR
17	OE.INGEST	.2160	Missing complexOrderID	*Events: OOA, OSL, OOM, OSLM, OOJ, OSLJ, OOR, OIR, OFP* complexOrderID must be provided if complexOptionID is provided.	ERROR
18	OE.INGEST	.2170	Missing or Invalid displayPrice	*Events: OOA, OOM, OOJ, OFP* displayPrice must be provided and greater than or equal to zero on simple option orders (i.e.	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
				`complexOrderID` is not populated) when `displayQty` is provided and is greater than zero.	
19	OE.INGEST	.2180	Missing price	*Events: OOA OOR, OIR, OOMR* `price` must be provided and greater than or equal to zero on simple option orders (i.e. `complexOrderID` is not populated) when `orderType` indicates a Limit order. Validation does not apply to Market orders. *Events: OOM* `price` must be provided and greater than or equal to zero on simple option orders (i.e. `complexOrderID` is not populated) when `orderType` indicates a Limit order and `initiator` is 'Firm' or 'MarketMaker'.	ERROR
20	OE.INGEST	.2200	Missing routingParty	*Events: OOA* `routingParty` must be provided for simple options (i.e. when `complexOrderID` is not populated). *Events: OOM, OOJ* `routingParty` must be provided for simple options (i.e. when `complexOrderID` is not populated) when `initiator` is 'Firm' or 'MarketMaker'.	ERROR
21	OE.INGEST	.2210	Missing session	*Events: OOA* `session` must be provided for simple options (i.e. when `complexOrderID` is not populated). *Events: OOM, OOJ* `session` must be provided for simple options (i.e. when `complexOrderID` is not populated) when `initiator` is 'Firm' or 'MarketMaker'.	ERROR
22	OE.INGEST	.2220	Missing timeInForce	*Events: OOM* `timeInForce` must be provided for simple options (i.e. when `complexOrderID` is not populated).	ERROR
23	OE.INGEST	.2230	Missing displayQty	*Events: OOM, OOJ*	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
				`displayQty` must be provided for simple options (i.e. when `complexOrderID` is not populated).	
24	OE.INGEST	.2240	Missing routedOrderID	*Events: OOM, OOJ* `routedOrderID` must be provided for simple options (i.e. when `complexOrderID` is not populated) when `initiator` is 'Firm' or 'MarketMaker'.	ERROR
25	OE.INGEST	.2250	Missing quantity	*Events: OOM, OOJ* `quantity` must be provided for simple options (i.e. when `complexOrderID` is not populated) when `initiator` is 'Firm' or 'MarketMaker'.	ERROR
26	OE.INGEST	.2260	Missing side	*Events: OOM, OOJ* `side` must be provided when `initiator` is 'Firm' or 'MarketMaker'	ERROR
27	OE.INGEST	.2270	Missing optionID or symbol	*Events: OOC, OOR, OIR, OOMR, OOCR* Exactly one of `optionID` or `symbol` must be provided.	ERROR
28	OE.INGEST	.2280	Both orderID and quoteID provided	*Events: OPTA* Both `optionID` and `quoteID` cannot be provided. *Events: OT* For buyDetails, both `buyDetails.optionID` and `buyDetails.quoteID` cannot be provided. For sellDetails, both `sellDetails.optionID` and `sellDetails.quoteID` cannot be provided.	ERROR
29	OE.INGEST, RME.INGEST	.2290	Invalid cycleDate	*Events: All Options Events, NOTE, RME, SHD, STE* When provided, `cycleDate` must be a Trade Date between the Event Date and Trade Date + 1, inclusive of both dates.	ERROR
30	BBO.INGEST	.2300	Invalid combination of bidPrice and bidQty	*Events: EBBO* If one of bidPrice or bidQty is provided, then both must be provided.	ERROR
31	BBO.INGEST	.2310	Invalid combination of	*Events: EBBO*	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
			askPrice and askQty	If one of askPrice or askQty is provided, then both must be provided.	

B.2. Reference Data Validation Errors

FINRA CAT will begin validating reference data at a later time.

The tables below contain error messages that are associated with Reference Data Validation.

Table 81: `MemberDictionary` Member Reference Data Validation Error Codes

The following reference data errors may be generated for a `MemberDictionary` file when invalid conditions are detected within the file. To resolve these errors, the Participant must provide a replacement `MemberDictionary` file with the correct information.

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	MD.REF	.500	Member Alias assigned to multiple Firms	*Event(s): MDE* A `memberAlias` may not be assigned to more than one Firm (`ID`) for the same exchange and trade date.	ERROR
2	MD.REF	.510	Member Details provided for missing Member Alias	*Event(s): MADE* Each `memberAlias` reported on a MADE record must correspond to a `memberAlias` reported on an MDE record for the same exchange and trade date.	ERROR

Table 82: `MarketMakerDictionary` Member Reference Data Validation Error Codes

The following reference data errors may be generated for a `MarketMakerDictionary` file when the provided member alias data does not exist in the Member Dictionary or the provided symbol is invalid.

If the error was caused by incorrect data submitted in the `MemberDictionary` file, the Participant must:

a) submit a replacement `MemberDictionary` file with the correct information

AND

b) submit a replacement `MarketMakerDictionary` file for reprocessing.

If the error was caused by incorrect data submitted in the MarketMakerDictionary file, the participant must:

a) submit a replacement MarketMakerDictionary file with the correct information.

If the error was caused by an incorrect symbol, the Participant must:

a) Determine if the symbol that was provided was correct.

b) If the symbol provided by the Participant was correct and should be in the Equity Symbol Master, contact the FINRA CAT Helpdesk for assistance in resolving the issue.

c) If the symbol provided by the Participant was incorrect, submit corrected records for reprocessing.

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	MMD.REF	.520	Market Maker not found in Member Dictionary	*Event(s): MMDE* The marketMaker is invalid. The marketMaker must correspond to a valid memberAlias in the Member Dictionary for the same exchange and trade date.	ERROR
2	MMD.REF	.530	Invalid Symbol	*Event(s): MMDE* The symbol is invalid. The symbol must correspond to a valid symbol for the same trade date.	ERROR

Table 83: OrderEvents Member Reference Data Validation Error Codes

The following reference data errors may be generated for an OrderEvents file when the provided member alias does not exist in the Member Dictionary or the provided symbol/option ID is invalid.

If the error was caused by incorrect data submitted in the MemberDictionary file, the Participant must:

a) submit a replacement MemberDictionary file with the correct information

AND

b) resubmit the rejected OrderEvents records using the corrections/replacement process described in Section 10.9.

If the error was caused by incorrect data submitted in the OptionsDictionary file, the Participant must:

a) submit a replacement `OptionsDictionary` file with the correct information

AND

b) resubmit the rejected `OrderEvents` records using the corrections/replacement process described in Section 10.9.

If the error was caused be incorrect data submitted in the `OrderEvents` file, the participant must:

a) resubmit the rejected `OrderEvents` records using the corrections/replacement process described in Section 10.9.

If the error was caused by an incorrect symbol, the Participant must:

a) Determine if the symbol that was provided was correct.

b) If the symbol provided by the Participant was correct and should be in the Equity Symbol Master, contact the FINRA CAT Helpdesk for assistance in resolving the issue.

c) If the symbol provided by the Participant was incorrect, submit corrected records for reprocessing.

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OE.REF	.540	Market Maker not found in Member Dictionary	*Event(s): OQ, OQC* The `marketMaker` is invalid. The `marketMaker` must correspond to a valid `memberAlias` in the Member Dictionary for the same exchange and trade date.	ERROR
2	OE.REF	.545	Member not found in Member Dictionary	*Event(s): All Equities and Options Events that include `member`* The `member` is invalid. The `member` must correspond to a valid `memberAlias` in the Member Dictionary for the same exchange and trade date.	ERROR
3	OE.REF	.550	Routing Party not found in Member Dictionary	*Event(s): All Equities and Options Events that include `routingParty`* The `routingParty` is invalid. The `routingParty` must correspond to a valid `memberAlias` in the Member Dictionary for the same exchange and trade date.	ERROR
4	OE.REF	.555	Floor Broker not found in Member Dictionary	*Event(s): OT* The `floorBroker` is invalid. The `floorBroker` must correspond to a valid `memberAlias` in the Member Dictionary for the same exchange and trade date.	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
5	OE.REF	.560	Invalid Symbol	*Event(s): All Equity Events, STE* The `symbol` is invalid. The `symbol` must correspond to a valid `symbol` for the same trade date.	ERROR
6	OE.REF	.565	Option ID not found in Options Dictionary	*Event(s): All Option Events, STE* The `optionID/complexOptionID` must correspond to a valid `optionID` in the Options Dictionary (provided via an OSDE or CODE record) for the same exchange and trade date.	ERROR

Table 84: `FinraTransactions` Reference Data Validation Error Codes

The following reference data errors may be generated for a `FinraTransactions` file when the provided member alias data does not exist in the Member Dictionary or the provided symbol is invalid.

If the error was caused by incorrect data submitted in the `MemberDictionary` file, the Participant must:

a) submit a replacement `MemberDictionary` file with the correct information

AND

b) resubmit the rejected `FinraTransactions` records using the corrections/replacement process described in Section 10.9.

If the error was caused be incorrect data submitted in the `FinraTransactions` file, the participant must:

a) resubmit the rejected `FinraTransactions` records using the corrections/replacement process described in Section 10.9.

If the error was caused by an incorrect symbol, the Participant must:

a) Determine if the symbol that was provided was correct.

b) If the symbol provided by the Participant was correct and should be in the Equity Symbol Master, contact the FINRA CAT Helpdesk for assistance in resolving the issue.

c) If the symbol provided by the Participant was incorrect, submit corrected records for reprocessing.

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	FT.REF	.570	Reporting Executing MPID not found in Member Dictionary	*Event(s): TRF* The `reportingExecutingMpid` is invalid. The `reportingExecutingMpid` must correspond to a valid `memberAlias` in the Member Dictionary for the same exchange and execution date.	ERROR
2	FT.REF	.575	Contra Executing MPID not found in Member Dictionary	*Event(s): TRF* The `contraExecutingMpid` is invalid. The `contraExecutingMpid` must correspond to a valid `memberAlias` in the Member Dictionary for the same exchange and execution date.	ERROR
3	FT.REF	.580	Invalid Symbol	*Event(s): TRF* The `symbol` is invalid. The `symbol` must correspond to a valid `symbol` for the same execution date (for TRF events).	ERROR

Table 85: `OTCHalts` Reference Data Validation Error Codes

The following reference data errors may be generated for an `OTCHalts` file when the provided symbol is invalid.

If the error was caused by an incorrect symbol, the Participant must:

a) Determine if the symbol that was provided was correct.

b) If the symbol provided by the Participant was correct and should be in the Equity Symbol Master, contact the FINRA CAT Helpdesk for assistance in resolving the issue.

c) If the symbol provided by the Participant was incorrect, submit a replacement `OTCHalts` file with the correct information.

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OTH.REF	.590	Invalid Symbol	*Event(s): FHR* The `symbol` is invalid. The `symbol` must correspond to a valid symbol for the same trade date.	ERROR

B.3 Linkage Discovery Errors

Linkage Discovery errors are generated by performing event comparisons that result in the identification of unlinked events. See §10.10 for information on the Linkage Discovery process.

Unlinked error codes are assigned based on a processing order when determining the reason for an unlinked event. The process begins with the check associated with the codes having the lowest sequence value. When the “Multiple Fields did not Match” reason is assigned, it is because a determination could not be made. In such cases, it is possible that the unlink reason is because the other party’s event was not reported or had a processing error which prevented the event from participating in Linkage Discovery. In cases when linkage did not occur between venues, separate error codes will be assigned to the CAT Reporter whose record did not link and the CAT Reporter that was named.

Table 86: Duplicate Record Error Codes

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OE.INTRAEXCHLNK	.301	Duplicate Event	*Event(s): All Equity and Option Events, FHR, TRF* The Exchange or FINRA event has already been received by CAT. The first instance of the event is retained; all subsequent submissions are rejected. This rejection is not repairable.	WARNING

Table 87: Intravenue Linkage Error Codes

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OE.INTRAEXCHLNK	.5000	Missing a parent	The event in question does not have a required parent.	ERROR
2	OE.INTRAEXCHLNK	.5001	Trade Event – Order/Quote not found	The Trade Event side details reference an Order Key/Quote Key that does not exist in CAT because it was not reported or was rejected.	ERROR
3	OE.INTRAEXCHLNK	.5002	Paired Orders - Corresponding Paired Order Not found	The paired order in orderAttributes name/value pair does not match another order.	ERROR
4	OE.INTRAEXCHLNK	.5003	Originating event not found for long Lived order	This is for the order restatement event errors specifically. Occurs if OORS event is received, and the events from the previous day(s) are not found.	ERROR
5	OE.INTRAEXCHLNK	.5004	Matching trade not found	A post trade allocation/supplemental trade event refers to a trade that cannot be located	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
6	OE.INTRAEXCHLNK	.5005	Late record, correction, or file replacement received after correction window (received after t+3 at 8:00 a.m. ET)	Warning will be retired as of June 15, 2022. Any input received after the correction window	WARNING
7	OE.INTRAEXCHLNK	.5007	Duplicate Trade Key	More than one Trade event was reported with the same Trade Key on the current CAT Processing Date. All events associated with the duplicate Trade Key will be rejected.	ERROR
8	OE.INTRAEXCHLNK	.5009	Duplicate Quote Key	More than one New Quote event were reported with the same Quote Key on the current CAT Processing Date. All events associated with the duplicate Quote Key will be rejected.	ERROR
9	OE.INTRAEXCHLNK	.5011	Duplicate Order Key	More than one primary order event and/or secondary order event which reassigned an Order Key was reported with the same Order Key on the current CAT Processing Date. All events associated with the duplicate Order Key will be rejected.	ERROR
10	OE.INTRAEXCHLNK	.5012	Duplicate Fulfillment Key	More than one Order Fulfillment events or Fulfillment Amendment events which assigned a new Fulfillment key were reported with the same Fill Key on the current CAT Processing Date. All events with a duplicate Fulfillment Key will be rejected	ERROR

Table 88: Intervenue Linkage Error Codes (Reported by Exchange/Display-Only Facility)

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OE.INTERVENUELNK	.6004	*routedOrderID* not found	The routedOrderID on the exchange Order Route/Order Accept/Equity Best Bid and Offer event does not match to a corresponding routedOrderID on the industry member order	ERROR
2	OE.INTERVENUELNK	.6006	*routingParty* did not match	A matching *routedOrderID* was identified; however, the routedOrderID on the exchange Order Route/Order Accept/Equity Best Bid and Offer event does not match to a corresponding *senderIMID* on the industry member order	ERROR
3	OE.INTERVENUELNK	.6008	*symbol* did not match	A matching *routedOrderID* was identified, however the *symbol* [for equity	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
			OR *optionID* did not match	events] or *optionID* [for option events] on the exchange Order Route/Order Accept/Equity Best Bid and Offer event does not match to a corresponding *symbol* or *optionID* on the industry member order	
4	OE.INTERVENUELNK	.6010	*session* did not match	A matching *routedOrderID* was identified, however, the *session* on the exchange Order Route/Order Accept/Equity Best Bid and Offer event does not match to a corresponding session on the industry member order	ERROR
5	OE.INTERVENUELNK	.6012	Multiple fields did not match	A matching routedOrderID was identified, however, the symbol, senderIMID or a combination of fields on the exchange Order Route/Order Accept/Equity Best Bid and Offer event does not match to corresponding field(s) on the industry member order	ERROR
6	OE.INTERVENUELNK	.6014	*destination* did not match	A matching *routedOrderID* was identified in an industry member order; however, the *ExchangeID* on the Order Route/Order Accept/Equity Best Bid and Offer event did not match the *destination* on the corresponding industry member order.	ERROR
7	OE.INTERVENUELNK	.6016	Duplicate Route Linkage Key on Route to Firm	Unlinked due to duplicated Route Linkage Key on an outbound Route from an Exchange.	ERROR
8	OE.INTERVENUELNK	.6018	Duplicate Route Linkage Key on Accept/Modify	Unlinked due to a duplicated Route Linkage Key on an Accept/Modify/Equity Best Bid and Offer event even from an Exchange.	ERROR
9	OE.INTERVENUELNK	.6020	Duplicate Exchange/Firm Trade Key	More than one Trade event was reported with the same Exchange/Firm Trade Key on the current CAT Processing Date. All events associated with the duplicate Exchange/Firm Trade Key will be rejected.	ERROR
10	OE.INTERVENUELNK	.6022	*tapeTradeID* did not match	The *MOOTLINK* value (provided via the *executionCodes* MOOTLINK Name/Value pair) on the exchange Option Trade event does not match to a corresponding *tapeTradeID* on the industry member trade. *Effective December 5, 2022.*	ERROR
11	OE.INTERVENUELNK	.6024	*marketCenterID* did not match	A matching *tapeTradeID* was identified, however, the *exchangeID* on the exchange Option Trade event does not match to a corresponding *marketCenterID* on the industry member trade. *Effective December 5, 2022.*	ERROR
12	OE.INTERVENUELNK	.6026	*side* in *buyDetails*	A matching *tapeTradeID* was identified, however, the *side* in *buyDetails* on the	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
			did not match	exchange Option Trade event does not match to a corresponding *side* in *buyDetails* on the industry member trade. *Effective December 5, 2022.*	
13	OE.INTERVENUELNK	.6028	*side* in *sellDetails* did not match	A matching *tapeTradeID* was identified, however, the *side* in *sellDetails* on the exchange Option Trade event does not match to a corresponding *side* in *sellDetails* on the industry member trade. *Effective December 5, 2022.*	ERROR

Table 89: Intervenue Linkage Error Codes (Reported by Firm)

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OE.INTERVENUELNK	.7005	*Named routedOrderID* not found	The *routedOrderID* reported by the Industry Member on the Order Route event does not match to a corresponding *routedOrderID* on the exchange order/Equity Best Bid and Offer event.	ERROR
2	OE.INTERVENUELNK	.7007	*Named routingParty* did not match	A matching *routedOrderID* was identified; however, the *senderIMID* on industry member order *did not match the routingParty on the exchange order*/Equity Best Bid and Offer event.	ERROR
3	OE.INTERVENUELNK	.7009	*Named symbol* did not match OR *Named optionID* did not match	A matching routedOrderID was identified, however the *symbol* [for equity events] or *optionID* [for option events] did not match on the corresponding *symbol/optionID* on the exchange order/Equity Best Bid and Offer event.	ERROR
4	OE.INTERVENUELNK	.7011	*Named session* did not match	A matching *routedOrderID* was identified, however, the *session* did not match the *session* on the corresponding exchange order/Equity Best Bid and Offer event.	ERROR
5	OE.INTERVENUELNK	.7013	Named Multiple fields did not match	A matching routedOrderID was identified, however, the symbol, senderIMID or a combination of fields reported on the Order Route event did not match the symbol or routingParty on the corresponding exchange order/Equity Best Bid and Offer event.	ERROR
6	OE.INTERVENUELNK	.7015	Named *destination* did not match	Named in an industry member order but the *destination* on the industry member order route did not match the *ExchangeID* reported on the corresponding Order Route event/Equity Best Bid and Offer event.	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
7	OE.INTERVENUELNK	.7017	Named *tapeTradeID* did not match	The *tapeTradeID* reported by the Industry Member on the Trade event did not match the unique identifier (e.g. MOOTLINK) provided on the exchange trade.	Error
8	OE.INTERVENUELNK	.7019	Named *marketCenterID* did not match	A matching *tapeTradeID* was identified; however, the *marketCenterID* reported on the Industry Member Trade event did not match the exchange ID on the exchange Trade Event.	Error
9	OE.INTERVENUELNK	.7021	Named *side* in buyDetails did not match	A matching *tapeTradeID* was identified; however, the *side* reported on the **buy side** of the Industry Member Trade event did not match the side on the exchange Trade Event.	Error
10	OE.INTERVENUELNK	.7023	Named *side* in sellDetails did not match	A matching *tapeTradeID* was identified; however, the *side* reported on the **sell side** of the Industry Member Trade event did not match the side on the exchange Trade Event.	Error

Table 90: Off-Exchange Trade Linkage Error Codes (Reported by Exchange)

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OE.TRADELNK	.4002	Matching *tapeTradeID* cannot be found	The unique identifier (e.g., Branch Sequence Number, Compliance ID) provided on the TRF/ADF/ORF Trade Report did not match the tapeTradeID reported by the Industry Member on a Trade event	ERROR
2	OE.TRADELNK	.4004	*marketCenterId* cannot be found	A matching *tapeTradeID* was identified, but Market Center Id provided on the TRF/ADF/ORF Trade Report did not match the marketCenterId reported by the Industry Member on a Trade event	ERROR
3	OE.TRADELNK	.4006	*symbol* cannot be found	A matching *tapeTradeID* was identified, but the *symbol* provided on the TRF/ADF/ORF Trade Report did not match the symbol reported by the Industry Member on a Trade event	ERROR
4	OE.TRADELNK	.4010	*Reporting or Contra IMID* cannot be found	A matching *tapeTradeID* was identified, but the *Reporting or Contra IMID* provided on the TRF/ADF/ORF Trade Report did not match the CATReporterIMID reported by the Industry Member on a Trade event	ERROR
5	OE.TRADELNK	.4012	Multiple fields did not match	A TRF/ADF/ORF Trade Report with a matching unique identifier (e.g. Branch Sequence Number) was found however symbol, MarketCenterID, CATReporterIMID or a combination of fields provided on the TRF/ADF/ORF Trade Report did not match the	ERROR

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
				corresponding fields in the Industry Member on a Trade event	

Table 91: Off-Exchange Trade Linkage Error Codes (Reported by TRF)

#	Error Prefix	Error Code	Error Code Description	Explanation	Warning/ Error
1	OE.TRADELNK	.5003	*Named* - Matching *tapeTradeID* cannot be found	The *tapeTradeID* reported by the Industry Member on a Trade event did not match to a corresponding *tapeTradeID* Compliance ID in the TRF/ADF/ORF Trade Report	ERROR
2	OE.TRADELNK	.5005	*Named - marketCenterId* cannot be found	A matching *tapeTradeID* was identified, but marketCenterId reported by Industry Member did not match the Execution Time on the TRF/ADF/ORF trade report	ERROR
3	OE.TRADELNK	.5007	*Named - symbol* cannot be found	A matching *tapeTradeID* was identified, but the *symbol* reported by Industry Member did not match the *symbol* on the TRF/ADF/ORF trade report	ERROR
4	OE.TRADELNK	.5009	*Named* - Multiple fields did not match	A TRF/ADF/ORF Trade Report with a matching unique identifier (such as Branch Sequence Number) was found however symbol, MarketCenterID, CATReporterIMID or a combination of fields reported by Industry Member did not match to the corresponding fields on the TRF/ADF/ORF trade report	ERROR
5	OE.TRADELNK	.5011	*Named – CATReporterIMID* cannot be found	A matching *tapeTradeID* was identified, but the CATReporterIMID reported by Industry Member did not match the *Reporting or Contra IMID* on the TRF/ADF/ORF trade report	ERROR

B.4. Error Prefix Definition

Table 92: Error Prefix Definitions

#	Error Prefix	Definition
1	FILE.NAME	File name validation errors
2	FILE.TIMEOUT	Data and corresponding acknowledgement
3	FT.INGEST	Error on FINRA TRF/ADF/ORF file validation
4	BBO.INGEST	Error on Best Bid and Offer file validations.
5	FT.REF	Member or Symbol Reference Date validation error on FinraTransactions file
6	INT.META	Error on metadata file validation against its corresponding data file name.

7	MD.REC	Error on Member Dictionary file validation
8	MD.REF	Member Reference Data validation error on MemberDictionary file
9	OD.REC	Error on Options Dictionary file validation
10	MMD.REC	Error on Market Maker Dictionary file validation
11	MMD.REF	Member or Symbol Reference Data validation error on MarketMakerDictionary file
12	OE.INGEST	Error on Order Events file validation
13	OE.INTRAEXCHLNK	Error or warning in Order Events during Intra Exchange Linkage Validation
14	OE.INTERVENUELNK	Error during linkage between Industry Member Order Events and Exchange Order Events (applicable to option and equity exchanges)
15	OE.TRADELNK	Error during linkage between Industry Member Order Events and TRF reported data
16	OE.REF	Member, Symbol, or OptionID Reference Data validation error on OrderEvents file
17	OTH.REC	Error on OTC Halts/Resumes file validation
18	OTH.REF	Symbol Reference Data validation on OTCHalts file

Appendix C. Placeholder

The previous content of Appendix C has been removed because it is not applicable to the FINRA CAT Plan Participant Technical Specifications. The heading is being retained to maintain the structure of the Appendix. This section may be repurposed in the future.

Appendix D. CAT Date Definitions and Reporting Guidelines

The following key date terms are used throughout the document for reporting instructions:

Table 93: Key Date Terms

Term	Definition	Usage
Event Timestamp	The date and time the event occurred.	*eventTimestamp* is a field defined on every CAT event. Used to assign the CAT Trading Day.
Event Date	The date portion of the Event Timestamp.	Part of all Route Linkage Keys, the TRF Linkage Key, and the MOOT Linkage Key. Used to link records within the Event Date.
File Generation Date	The date the file was generated or reported. File Naming convention requires that the TRADE DATE (as defined below) be used in the File Name.	Used to guarantee uniqueness for a file across dates.
CAT Trading Day	Trading Day for Plan Participants is defined as beginning at midnight immediately following a Trade Date and ending immediately prior to midnight on the next Trade Date. Weekends and holidays are not considered a Trading Day. Refer to Section 9.7 and the table below for more information, including examples demonstrating the calculation of CAT Trading Day, Submissions Due Date and Corrections Due Date.	Used to calculate the submission due date and corrections due date. For an event occurring on CAT Trading Day T: **Submissions Due By:** CAT Trading Day + 1 @ 8:00 a.m. ET **Corrections Due By:** CAT Trading Day + 3 @ 8:00 a.m. ET
Trade Date	Trade Date for Plan Participants is defined as beginning at midnight immediately following a Trade Date and ending immediately prior to midnight on the next Trade Date. Weekends and holidays are not considered a Trade Date. An event occurring on a weekend or holiday will be assigned to the next Trade Date.	Used to calculate the due date of data delivered to Regulatory Users. **Due Date for Data and Associated Lifecycle Assignment delivery to Regulatory Users:** Trade Date + 5 8:00 a.m. ET Used to calculate summaries and present feedback on the CAT Reporter Portal representing events for the same Trade Date, regardless of when the events were reported.
CAT Processing Date	Date representing the set of events reported for a CAT Trading Day. Events reported late to CAT will be assigned the CAT Processing Date reflective of when they were reported. For example, an event, that occurred at 2:00 p.m. ET on T which is reported to CAT after T+1 8 a.m. ET and prior to T+2 @ 8:00 a.m. ET will be assigned CAT Processing Date of T+1.	Used to identify late submissions and late repairs. Used to calculate summaries and present feedback on the CAT Reporter Portal representing events reported on the CAT Processing Date, regardless of the Event Date.

Term	Definition	Usage
Cycle Date	The exchange's effective business date. "DAY" orders are effective until close of regular business on the Trading Day. Date format YYYYMMDD. This must be a Trade Date.	Used to support linkage.
Order Key Date	The date and time the OrderID was assigned.	*orderKeyDate* is a field defined on Order events, and other events which specify an Order Key. Used to support uniqueness of an Order Key. If time is not needed to guarantee a unique Order Key, the time portion may be populated with zeros.
Trade Key Date	The date and time the TradeID was assigned.	*tradeKeyDate* is a field defined on Trade events. Used to support uniqueness of a Trade Key. If time is not needed to guarantee a unique Trade Key, the time portion may be populated with zeros.
Quote Key Date	The date and time the QuoteID was assigned.	*quoteKeyDate* is a field defined on Quote events. Used to support uniqueness of a Quote Key. If time is not needed to guarantee a unique Quote Key, the time portion may be populated with zeros.

Table 94: Event Scenarios

#	Event Timestamp	Event Date	Trade Date	Cycle Date	CAT Trading Day	Submission Due	Corrections Due
Weekend Scenario							
1	Sun, 11/15/20 20:16:00 ET	11/15/2020	11/16/2020	11/16/2020	11/16/2020	11/17/20, 8:00 a.m. ET	11/19/20 8:00 a.m. ET
2	Mon, 11/16/20 00:00:01 ET	11/16/2020	11/16/2020	11/16/2020	11/16/2020	11/17/20, 8:00 a.m. ET	11/19/20 8:00 a.m. ET
3	Mon, 11/16/20 03:00:00 ET	11/16/2020	11/16/2020	11/16/2020	11/16/2020	11/17/20, 8:00 a.m. ET	11/19/20 8:00 a.m. ET
4	Mon, 11/16/20 09:30:01 ET	11/16/2020	11/16/2020	11/16/2020	11/16/2020	11/17/20, 8:00 a.m. ET	11/19/20 8:00 a.m. ET
5	Mon, 11/16/20 16:35:00 ET	11/16/2020	11/16/2020	11/16/2020	11/16/2020	11/17/20, 8:00 a.m. ET	11/19/20 8:00 a.m. ET
Mid-week (Regular) Scenario							
6	Mon, 11/16/20 20:16:00 ET	11/16/2020	11/16/2020	11/17/2020	11/16/2020	11/17/20, 8:00 a.m. ET	11/19/20 8:00 a.m. ET
7	Tues, 11/17/20 03:00:00 ET	11/17/2020	11/17/2020	11/17/2020	11/17/2020	11/18/20, 8:00 a.m. ET	11/20/20 8:00 a.m. ET
8	Tues, 11/17/20 09:35:00 ET	11/17/2020	11/17/2020	11/17/2020	11/17/2020	11/18/20, 8:00 a.m. ET	11/20/20 8:00 a.m. ET
9	Tues, 11/17/20 16:45:00 ET	11/17/2020	11/17/2020	11/17/2020	11/17/2020	11/18/20, 8:00 a.m. ET	11/20/20 8:00 a.m. ET
Holiday Scenario (Thanksgiving, Thursday 11/26/2020)							
10	Wed, 11/25/20 20:30:05 ET	11/25/2020 *11/26 holiday	11/25/2020	11/27/2020	11/25/2020	11/27/20 8:00 a.m. ET	12/1/20 8:00 a.m. ET
11	Thur, 11/26/20 01:00:00 ET	11/26/2020	11/27/2020	11/27/2020	11/27/2020	11/27/20 8:00 a.m. ET	12/1/20 8:00 a.m. ET
12	Thur, 11/26/20 09:14:00 ET	11/26/2020	11/27/2021	11/27/2020	11/27/2020	11/27/20 8:00 a.m. ET	12/1/20 8:00 a.m. ET
13	Thur, 11/26/20 22:00:00 ET	11/26/2020	11/27/2021	11/27/2020	11/27/2020	11/27/20 8:00 a.m. ET	12/1/20 8:00 a.m. ET
14	Fri, 11/27/20 01:00:00 ET	11/27/2020	11/27/2021	11/27/2020	11/27/2020	11/30/20 8:00 a.m. ET	12/2/20 8:00 a.m. ET
15	Fri, 11/27/20 11:00:00 ET	11/27/2020	11/27/2021	11/27/2020	11/27/2020	11/30/20 8:00 a.m. ET	12/2/20 8:00 a.m. ET

Appendix E. Placeholder

The previous content of Appendix E has been removed because it is not applicable to the FINRA CAT Plan Participant Technical Specifications. The heading is being retained to maintain the structure of the Appendix. This section may be repurposed in the future.

Appendix F. Data Dictionary

Each field presented in this technical specification is defined below in terms data type, related message types, description, and allowed values.

Table 95: Data Dictionary

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

exchOriginCode executionCodes handlingInstructions orderAttributes orderType Participant ID saleCondition timeInForce type

Field Name	Data Type	Description
acceptTime	Time	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Time the trade was accepted by the contra party.
acceptTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the trade was accepted by the contra party.
actionType	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates if this is a new event, a FINRA-initiated correction, or a firm-initiated correction. **Allowed Values** **NEW** New Record **COR** Correction Record **FCOR** Firm Correction Record (indicates the FINRA submission of an update or correction of data made by a firm)
askPrice	Price	*Event(s): Equity Best Bid and Offer Event (EBBO), Quote (OQ)* The price being asked for the option in a quote.
askQuoteID	Text (40)	*Event(s): Option Quote (OQ), Option Quote Cancel (OQC)* Contains the quote ID for the ask for two-sided quote reporting.
askQty	Unsigned	*Event(s): Equity Best Bid and Offer Event (EBBO), Quote (OQ)* The quantity being asked for the option in a quote.
asOfFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates as-of trade. **Allowed Values** **Y** Trade Reported As-Of **N** Trade Reported on Execution Date
assumedExecutionTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the trade is assumed to have been executed based on available information.
Ats	Boolean	*Reference Data: Member Alias Detail Entry (MADE)* Indicates that the memberAlias is an ATS.
awayExchange	Exchange ID	*Event(s): Self-Help Declaration (SHD)* Exchange ID of the exchange affected by the self-help event.

Field Name	Data Type	Description
bidPrice	Price	*Event(s): Equity Best Bid and Offer Event (EBBO), Option Quote (OQ)* The price being bid for the option (can be zero in two-sided quote) in a quote event.
bidQty	Unsigned	*Event(s): Equity Best Bid and Offer Event (EBBO), Option Quote (OQ)* The quantity being bid for the option (can be zero in two-sided quote) in a quote event.
buyDetails	Side Trade Details	*Event(s): Order Trade Event, Trade Correction Event, Option Trade Event, Options Trade Correction Event* Object in a trade event that contains information for the buy side of the trade. Format and element definitions for Buy Details are described in Trade Side Details. For side trade details for equities, please refer to section 4.5. For side trade details for options, please refer to section 5.2.5.1.
cancellationTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the reporting party cancelled the trade.
cancelQty	Unsigned	*Event(s): Order Canceled Event, Options Order Canceled Event* The quantity being canceled in Order Cancel Event and Options Order Canceled Event. A value of zero means that the cancel was for the full remaining quantity. For example, if an order for 500 shares had partially executed 200 shares, and then the remainder was canceled, the cancelQty could contain either 300 or 0.
cancelReason	Choice	*Event(s): Order Canceled Event, Quote Cancel Event, Options Order Canceled Event* Expresses the cancellation reason for a quote or order with one of the below accepted values. Additional values may be added by request. **Allowed Values** **IOC** Immediately canceled **EXP** Expired **REQ** Explicit request to cancel the order **DIS** Session disconnected **ALL** Market Maker Canceled All Quotes **Allowed Values: Cboe Legacy (C1) Only** *active 3/29/2019 - 10/4/2019* **NOTHING_DONE** **USER** **SYSTEM** **LOST_CONNECTION** **INSUFFICIENT_QUANTITY** **SPECIAL_ADJUSTMENT** **QRM_REMOVED** **INSUFFICIENT_QUANTITY_BUY_SIDE** **INSUFFICIENT_QUANTITY_SELL_SIDE** **WASH_TRADE_PREVENTION** **QUOTE_UPDATE_CONTROL** **FAILOVER** **QUOTE_IN_TRIGGER** **INVALID_SESSION_ID** **SAL_IN_PROGRESS** **CROSS_IN_PROGRESS**

Field Name	Data Type	Description
cancelReason ***(continued)***		**INVALID_NBBO** **NOT_WITHIN_NBBO** **TRADE_THROUGH_CBOE** **INSUFFICIENT_CUSTOMER_ORDER_QUANTITY** **INSUFFICIENT_CROSS_ORDER_SIZE** **INSUFFICIENT_CROSS_ORDER_DOLLAR_AMOUNT** **SELL_SHORT_RULE_VIOLATION** **CANCEL_ON_RSS** **CALL_BID_EXCEEDS_UNDERLYING_PRICE** **PUT_BID_EXCEEDS_STRIKE_PRICE** **LIMIT/EXECUTION_PRICE_WOULD_BE_DEBIT** **LIMIT/EXECUTION_PRICE_EXCEEDS_MAX_VALUE** **NO_USER_ACTIVITY** **BROKER_OPTION** **CANCEL_PENDING** **CROWD_TRADE** **DUPLICATE_ORDER** **EXCHANGE_CLOSED** **GATE_VIOLATION** **INVALID_ACCOUNT** **INVALID_AUTOEX_VALUE** **INVALID_CMTA** **INVALID_FIRM** **INVALID_ORIGIN_TYPE** **INVALID_POSITION_EFFECT** **INVALID_PRICE** **INVALID_PRODUCT** **INVALID_PRODUCT_TYPE** **INVALID_QUANTITY** **INVALID_SIDE** **INVALID_SUBACCOUNT** **INVALID_TIME_IN_FORCE** **INVALID_USER** **LATE_PRINT** **NOT_FIRM** **MISSING_EXEC_INFO** **NO_MATCHING_ORDER** **NON_BLOCK_TRADE** **NOT_NBBO** **COMM_DELAYS** **ORIGINAL_ORDER_REJECTED** **OTHER** **PROCESSING_PROBLEMS** **PRODUCT_HALTED** **PRODUCT_IN_ROTATION** **STALE_EXECUTION** **STALE_ORDER**

Field Name	Data Type	Description	
cancelReason ***(continued)***		**ORDER_TOO_LATE**	
		TRADE_BUSTED	
		TRADE_REJECTED	
		ORDER_TIMEOUT	
		REJECTED_LINKAGE_TRADE	
		SATISFACTION_ORD_REJ_OTHER	
		UNKNOWN_ORDER	
		INVALD_EXCHANGE	
		TRANSACTION_FAILED	
		NOT_ACCEPTED	
		SUSPENDED	
		AWAY_EXCHANGE_CANCEL	
		LINKAGE_CONDITIONAL_FIELD_MISSING	
		LINKAGE_EXCHANGE_UNAVAILABLE	
		LINKAGE_INVALID_MESSAGE	
		LINKAGE_INVALID_DESTINATION	
		LINKAGE_INVALID_PRODUCT	
		LINKAGE_SESSION_REJECT	
		Allowed Values: CBOE	
		Admin	Admin
		CloseOnly	Options only - attempt to open a position when a series is in a "close only" status
		Consent	Both parties agreed to break trade
		DefaultRiskNotSet	Options only - risk configuration is incomplete
		Duplicate	Duplicate
		Erroneous	Clearly erroneous
		Expired	GTC orders
		FailedToQuote	Could not reflect on SUMO
		FloorError	
		NoGlobalLiquidity	Ran out of liquidity to execute against
		Halted	Halted
		IncorrectDataCenter	Tried to send order to DR site
		TooLate	Too late to cancel
		OrderRateThreshold	Exceeded order rate threshold
		LockOrCross	Order would lock or cross NBBO
		MaxSizeExceeded	Exceeded client specific maximum order size
		NoLiquidity	Ran out of liquidity to execute against
		OrderUnknown	Supplied order id doesn't match a known order
		Pending	Can't modify an order that is routed away
		WaitingForTape	Waiting for first trade before allowing executions
		RouteUnavailable	Route unavailable
		QuoteUnavailable	Quote unavailable
		Short	short price violation
		TradeThrough	order would have caused a trade-through violation
		User	user requested
		WouldWash	Execution would Wash Trade

Field Name	Data Type	Description
cancelReason *(continued)*		**WouldRemove** AddLiquidityOnly order would have removed liquidity **Symbol** symbol not supported **Other** unforeseen reason **BulkOrder** Cancel due to BulkOrder (BOE) **OrdersDisallowed** order entry disallowed **MassCancelSingleAck** mass cancel with single ack option **RiskMgmtFirmLevel** Risk Management Trigger Hit at "Firm" Level **NoOddLotIPOs** On IPO day opening print must be at least as large as a round lot - No odd lots **MarketAccessLimit** (US) Market Access Risk limit exceeded in router **MaxOpenOrdersExceeded** exceeded maximum open orders permitted **MismatchedRemainder** remainder on incoming request does not match remainder in our system **Reload** restatement for reserve reload **RiskMgmtSymbolLevel** Risk Management Trigger Hit at "Symbol/OSI" Level **RiskMgmtGroupLevel** Risk Management Trigger Hit at "Group" Level **LimitUpDown** LU/LD (e.g., tried to rest through the LU/LD bands) **WouldRemoveUnSlide** AddLiquidityOnly order tried to unslide but would have resulted in removing liquidity **MarketCrossed** Crossed Market Protection **InReplay** message received during replay **Persist** GTC order done for today (will get restated next trading day) **SessionEnd** canceled automatically at end of regular or extended trading session based on customer send coding **ClearingFailure** Trade Failed to Clear **GroupLevelRiskManagement** Risk Management Trigger Hit at “Group” Level **Allowed Values: BOX** **TraderCanceled** **Eliminated** **EliminatedOutOfLimits** **EliminatedDueToUnpricedLeg** **CancelledBySupervisor** **CancelPending** **EliminatedByCircuitBreaker** **EliminatedOnDisconnection** **EliminatedByMarketControl** **EliminatedDueToTradingRestriction** **EliminatedDueToTradeLimitExceeded** **EliminatedDueToTradeActivityLimitExceeded** **EliminatedDueToMaximumNbTriggersLimitExceeded** **EliminatedDueToDrillThroughProtection** **EliminatedDueToMMProtection** **Allowed Values: LTSE** **0** Other - This order was canceled for some other reason not listed.

Field Name	Data Type	Description
cancelReason ***(continued)***		**1** UserRequestedCancel - The client sent a OrderCancelRequest or OrderMassCancelRequest for this order **4** EndOfTrading - The order was sent with the DAY time in force set, and the DAY trading session completed. **5** LimitUpLimitDown - The price of the order fell outside market LULD bands, and the re-pricing modifier was not specified on the order. **6** Halted - The market on the order's security was halted. **7** ExchangeSupervisory - Operational or supervisory actions taken by MEMX resulted in the cancellation of this order. **8** OrderExpired - The order was sent with an expiration time and had the "good for time" time in force set, and the supplied expiration time passed. **9** LockOrCrossBook - The order was not externally routable, and market conditions would have resulted in this order crossing or locking the order book **10** SelfTradePrevention - This or another associated order's specified self trade prevention behavior triggered the cancellation of this order. **11** InsufficientQuotes - The order was cancelled because there are insufficient quotes on the book for the symbol. **12** NonCompliantPrice - The order was cancelled because the price in the order was non-compliant. **13** ParticipantDisconnect - The participant directed that their orders should be canceled when the trading system detects a disconnection, and the participant disconnected. **14** OrderNotBookable – The Order is not of bookable type (this may include market orders, IOC, FOK, etc) **15** TradeProtectionLimits - The price of the order fell outside market trade protection limits rule, and the re- pricing modifier was not specified on the order. **16** UnableToRoute – The Order was canceled because it was externally routable but could not be routed. **17** FirmDisabled – The order was cancelled because the firm was disabled. **18** MPIDDisabled – The order was cancelled because the MPID was disabled. **19** AccountDisabled – The order was cancelled because the Account was disabled. **20** NotionalExposureRiskBreached – The order was cancelled because a Notional Exposure Risk Rule was breached. **21** InvalidClOrdId **Allowed Values: MEMX** **0** Other - This order was canceled for some other reason not listed. **1** UserRequestedCancel - The client sent a OrderCancelRequest or OrderMassCancelRequest for this order **4** EndOfTrading - The order was sent with the DAY time in force set, and the DAY trading session completed. **5** LimitUpLimitDown - The price of the order fell outside market LULD bands, and the re-pricing modifier was not specified on the order. **6** Halted - The market on the order's security was halted. **7** ExchangeSupervisory - Operational or supervisory actions taken by MEMX resulted in the cancellation of this order. **8** OrderExpired - The order was sent with an expiration time and had the "good for time" time in force set, and the supplied expiration time passed.

Field Name	Data Type	Description
cancelReason ***(continued)***		**9** LockOrCrossBook - The order was not externally routable, and market conditions would have resulted in this order crossing or locking the order book **10** SelfTradePrevention - This or another associated order's specified self trade prevention behavior triggered the cancellation of this order. **11** InsufficientQuotes - The order was cancelled because there are insufficient quotes on the book for the symbol. **12** NonCompliantPrice - The order was cancelled because the price in the order was non-compliant. **13** ParticipantDisconnect - The participant directed that their orders should be canceled when the trading system detects a disconnection, and the participant disconnected. **14** OrderNotBookable – The Order is not of bookable type (this may include market orders, IOC, FOK, etc) **15** TradeProtectionLimits - The price of the order fell outside market trade protection limits rule, and the re- pricing modifier was not specified on the order. **16** UnableToRoute – The Order was canceled because it was externally routable but could not be routed. **17** FirmDisabled – The order was cancelled because the firm was disabled. **18** MPIDDisabled – The order was cancelled because the MPID was disabled. **19** AccountDisabled – The order was cancelled because the Account was disabled. **20** NotionalExposureRiskBreached – The order was cancelled because a Notional Exposure Risk Rule was breached. **21** InvalidClOrdId **Allowed Values: MIAX** **MIAMI_0004** UserCanceled **MIAMI_0005** HelpDeskCanceled **MIAMI_0006** WdCanceled **MIAMI_0007** CrossSameMpidCanceled **MIAMI_0009** OversizedAuctionCanceled **MIAMI_0010** ReintroduceCanceled **MIAMI_0018** TimeInForceCanceled **MIAMI_0019** NonTradeableCanceled **MIAMI_0020** CanceledOnClosing **MIAMI_0021** ProductHalted **MIAMI_0022** UserPurged **MIAMI_0023** MpidDeleted **MIAMI_0024** MpidPermissionDeleted **MIAMI_0025** RiskPurged **MIAMI_0026** SystemPurged **MIAMI_0027** InternalPurged **MIAMI_0029** GtcSpinCanceled **MIAMI_0030** LuldCanceled **MIAMI_0031** RpmBlockedMpidCanceled **MIAMI_0032** ComplexTradingSuspendedForCloudCanceled **MIAMI_0033** ComplexFeatureDisabledForUnderlyingCanceled **MIAMI_0034** ComplexStrategyNonTradeableCanceled

Field Name	Data Type	Description
cancelReason ***(continued)***		**MIAMI_0035** ComplexStrategyLegWithWideMbboCanceled **MIAMI_0036** ComplexStrategyLegWithPrimeAuctionCanceled **MIAMI_0037** ComplexStrategyLegWithRouteTimerCanceled **MIAMI_0038** ComplexStrategyLegWithLiqRefreshTimerCanceled **MIAMI_0039** ComplexIneligiblePriceCanceled **MIAMI_0040** ComplexStrategyAuctionInProgressCanceled **MIAMI_0041** ComplexOrderExhaustedDcMbboAfterEndOfComplexTimeCanceled **MIAMI_0042** ComplexStrategyPreOpenCanceled **MIAMI_0045** ComplexCollarPriceProtectionCanceled **MIAMI_0046** DerivedOrderFeatureDisableCanceled **MIAMI_0047** DerivedOrderStrategyNotFreeTradingCanceled **MIAMI_0048** DerivedOrderStrategyTopChangeCanceled **MIAMI_0049** DerivedOrderStrategyTopLockCanceled **MIAMI_0050** DerivedOrderReplaceCanceled **MIAMI_0051** DerivedOrderWorseSameSideMbboCanceled **MIAMI_0052** DerivedOrderLeanMbboWorseAbboCanceled **MIAMI_0053** DerivedOrderLeanMbboChangeCanceled **MIAMI_0054** DerivedOrderComponentNotFreeTradingCanceled **MIAMI_0055** DerivedOrderWideMarketCanceled **MIAMI_0056** DerivedOrderSystemIssueCanceled **MIAMI_0057** DerivedOrderTraded **MIAMI_0058** SspCanceled **MIAMI_0059** ComplexStrategyLegWithLiqExposureTimerCanceled **MIAMI_0060** MpppCanceled **MIAMI_0061** ComplexManagedProtectionOverrideCanceled **MIAMI_0062** ComplexMiaxStrategyPriceProtectionAssigned **MIAMI_0064** SellMarketOrderInZeroBidWideNbboCanceled **MIAMI_0065** QuoteReplaceRejectRestingQuoteCanceled **Allowed Values: MIAX - Pearl** **PEARL_0004** UserCanceled **PEARL_0005** HelpDeskCanceled **PEARL_0007** CrossSameMpidCanceled **PEARL_0012** RoutedToAwayMarket **PEARL_0018** TimeInForceCanceled **PEARL_0019** NonTradeableCanceled **PEARL_0021** ProductHalted **PEARL_0029** GtcSpinCanceled **PEARL_0030** LuldCanceled **PEARL_0031** RpmBlockedMpidCanceled **PEARL_0032** PriceProtectionCanceled **PEARL_0033** UserPurged **PEARL_0034** SystemPurged **PEARL_0035** PostOnlyLockingManagedCanceled **PEARL_0036** IrpAssigned **PEARL_0037** SspCanceled **PEARL_0038** MpppCanceled

Field Name	Data Type	Description
cancelReason ***(continued)***		**PEARL_0039** AutoReplaceOrderCanceledDueToRepaceReject **Allowed Values: MIAX - PEARLEQ Equities** **PEARLEQ_0001** UserMeo **PEARLEQ_0002** UserFoi **PEARLEQ_0003** UserPurgePort **PEARLEQ_0004** HelpDesk **PEARLEQ_0005** MFP **PEARLEQ_0006** ACOD **PEARLEQ_0007** ACOSF **PEARLEQ_0008** CRM **PEARLEQ_0009** OrderExpired **PEARLEQ_0010** PostOnlyCancelSymbolNotTrading **PEARLEQ_0011** ACOCR **PEARLEQ_0012** CrmNetNotional **PEARLEQ_0013** MinQtyCancelSymbolNotTrading **PEARLEQ_0014** PacMassCancel **PEARLEQ_0015** PacBlacklist **PEARLEQ_0016** CrmOpenOrderGrossNotional **PEARLEQ_0017** CrmOpenOrderNetNotional **PEARLEQ_0018** RpmOrderRate **PEARLEQ_0019** PurgeGroupUserPurgePort **PEARLEQ_0020** PurgeGroupMFP **PEARLEQ_0021** CrmTradeAndOpenOrderGrossNotional **PEARLEQ_0022** CrmTradeAndOpenOrderNetNotional **PEARLEQ_0023** WDFXDACOD **PEARLEQ_0024** WDFXDACOSF **PEARLEQ_0025** WDOFXDACOD **PEARLEQ_0026** WDOFXDACOSF **PEARLEQ_0027** SingleOrderMFPUI **PEARLEQ_0028** SingleOrderHelpDesk **PEARLEQ_0100** SelfTradeProtection **PEARLEQ_0101** TimeInForce **PEARLEQ_0102** PostOnlyLockingCrossingMbbo **PEARLEQ_0103** TradingCollarProtection **PEARLEQ_0104** RePriceFrequencyNoPriceSliding **PEARLEQ_0105** RePriceFrequencyOnce **PEARLEQ_0106** RePriceFrequencyOnceButCancelIfCrossedAtEntry **PEARLEQ_0107** IsoSellShortRegShoLockCrossNbbo **PEARLEQ_0108** LimitOrderPriceProtection **PEARLEQ_0109** RouteToPrimaryListingMarketRejected **PEARLEQ_0110** UnexpectedCancelByPrimaryListingMarket **PEARLEQ_0111** RoutedOrderTimeOut **PEARLEQ_0112** PacUnsolicitedBlacklist **PEARLEQ_0113** PacMarketOrderDuringTrading **PEARLEQ_0114** PacOrderReturnedDuringPacBlacklist **PEARLEQ_0115** PLEAcceptedCancelOfRejectedReplace

Field Name	Data Type	Description
cancelReason *(continued)*		**PEARLEQ_0116** ReplaceDuringPacBlacklist **PEARLEQ_0117** PacOrderReturnedAfterPacOrderAcceptEndTime **PEARLEQ_0118** PeggedOrderInvalidAdjustedReferencePrice **PEARLEQ_0119** ReplaceViolatesMarketImpactCollar **PEARLEQ_0120** SelfTradeProtectionCancelNewest **PEARLEQ_0121** SelfTradeProtectionCancelOldest **PEARLEQ_0122** SelfTradeProtectionCancelBoth **PEARLEQ_0123** SelfTradeProtectionDecrementAndCancel **PEARLEQ_0126** NotNbboSetter **PEARLEQ_0127** MassCancelMFPUI **PEARLEQ_0128** MassCancelMFPAPI **PEARLEQ_0129** MassCancelHelpDesk **PEARLEQ_0130** MassCancelUserPurgePort **PEARLEQ_0131** MassCancelPurgeGroupUserPurgePort **PEARLEQ_0132** MassCancelPurgeGroupMFPUI **PEARLEQ_0133** MassCancelPurgeGroupHelpDesk **PEARLEQ_0134** MassCancelPurgeGroupMFPAPI **PEARLEQ_9002** ReserveDisplayPartUpdateReason_Execution **PEARLEQ_9003** ReserveDisplayPartUpdateReason_Update **Allowed Values: MIAX - EMLD** **EMLD_0004** UserCanceled **EMLD_0005** HelpDeskCanceled **EMLD_0006** WdCanceled **EMLD_0007** CrossSameMpidCanceled **EMLD_0009** OversizedAuctionCanceled **EMLD_0010** ReintroduceCanceled **EMLD_0018** TimeInForceCanceled **EMLD_0019** NonTradeableCanceled **EMLD_0020** CanceledOnClosing **EMLD_0021** ProductHalted **EMLD_0022** UserPurged **EMLD_0023** MpidDeleted **EMLD_0024** MpidPermissionDeleted **EMLD_0025** RiskPurged **EMLD_0026** SystemPurged **EMLD_0027** InternalPurged **EMLD_0029** GtcSpinCanceled **EMLD_0030** LuldCanceled **EMLD_0031** RpmBlockedMpidCanceled **EMLD_0032** ComplexTradingSuspendedForCloudCanceled **EMLD_0033** ComplexFeatureDisabledForUnderlyingCanceled **EMLD_0034** ComplexStrategyNonTradableCanceled **EMLD_0035** ComplexStrategyLegWithWideMbboCanceled **EMLD_0036** ComplexStrategyLegWithPrimeAuctionCanceled **EMLD_0039** ComplexIneligiblePriceCanceled **EMLD_0040** ComplexStrategyAuctionInProgressCanceled

Field Name	Data Type	Description
cancelReason *(continued)*		**EMLD_0041** ComplexOrderExhaustedDcMbboAfterEndOfComplexTimeCanceled **EMLD_0042** ComplexStrategyPreOpenCanceled **EMLD_0045** ComplexCollarPriceProtectionCanceled **EMLD_0046** DerivedOrderFeatureDisableCanceled **EMLD_0047** DerivedOrderStrategyNotFreeTradingCanceled **EMLD_0048** DerivedOrderStrategyTopChangeCanceled **EMLD_0049** DerivedOrderStrategyTopLockCanceled **EMLD_0050** DerivedOrderReplaceCanceled **EMLD_0051** DerivedOrderWorseSameSideMbboCanceled **EMLD_0052** DerivedOrderLeanMbboWorseAbboCanceled **EMLD_0053** DerivedOrderLeanMbboChangeCanceled **EMLD_0054** DerivedOrderComponentNotFreeTradingCanceled **EMLD_0055** DerivedOrderWideMarketCanceled **EMLD_0056** DerivedOrderSystemIssueCanceled **EMLD_0057** DerivedOrderTraded **EMLD_0058** SspCanceled **EMLD_0059** ComplexStrategyLegWithLiqExposureTimerCanceled **EMLD_0060** PostOnlyLockingManagedCanceled **EMLD_0061** ComplexManagedProtectionOverrideCanceled **EMLD_0062** ComplexMiaxStrategyPriceProtectionAssigned **EMLD_0063** MpppCanceled **EMLD_0064** SellMarketOrderInZeroBidWideNbboCanceled **EMLD_0065** QuoteReplaceRejectRestingQuoteCanceled **Allowed Values: MIAX - SPHR** **SPHR_0004** UserCanceled **SPHR_0005** HelpDeskCanceled **SPHR_0007** CrossSameMpidCanceled **SPHR_0012** RoutedToAwayMarket **SPHR_0018** TimeInForceCanceled **SPHR_0029** GtcSpinCanceled **SPHR_0030** LuldCanceled **SPHR_0031** RpmBlockedMpidCanceled **SPHR_0032** PriceProtectionCanceled **SPHR_0033** UserPurged **SPHR_0034** SystemPurged **SPHR_0035** PostOnlyLockingManagedCanceled **SPHR_0036** IrpAssigned **SPHR_0037** SspCanceled **SPHR_0038** MpppCanceled **SPHR_0041** ComplexStrategyLegWithWideMbboCanceled **SPHR_0042** ComplexStrategyPreOpenCanceled **SPHR_0043** IncomingComplexOrderFinishedTradingOnArrival **SPHR_0044** ComplexCollarPriceAssigned **SPHR_0061** ComplexManagedProtectionOverrideCanceled **SPHR_0062** ComplexMiaxStrategyPriceProtectionAssigned **SPHR_0063** AutoReplaceOrderCanceledDueToRepaceReject

Field Name	Data Type	Description
cancelReason ***(continued)***		**Allowed Values: CHX** **A001_02A** New SNAP Order Reject - Order Terms are not valid for SNAP **A001_02B** New SNAP Order Reject - Invalid market condition **A001_07** Cancel Order, SNAP auction end **A001_11** SNAP Auction - Cancel of Satisfy/Route Order **A001_13** SNAP Auction - Reject of Satisfy/Route Order **A001_15** Cancel Order on SNAP Auction - Resting **U400_01** order reject-invalid content **U400_04** order reject-invalid trading session **U400_05** order reject-invalid market state **U400_06** order reject-invalid market conditions **U400_07** order message cannot be parsed **U400_08** order from PMM not is registered stock **U400_09** order from PMM did not include position **U400_10** order from PMM with position/side discrepancy **U400_11** IOC Order Reject-No PM LS **U400_14** Market IOC orders not allowed during extended sessions **U400_17** New AOO reject **U415_01** ME DAS Order Cancel on Restart **U430_01** satisfy cross reject-not regular-way settlement **U430_02** satisfy cross reject-short sale test failure **U430_03** satisfy cross reject-NBBO trade through **U430_04** satisfy cross reject-insufficient satisfy volume available **U430_05** satisfy cross reject-outside crossed NBBO **U430_06** satisfy cross reject-crossed market **U431_01** yield cross reject-not regular-way settlement **U431_02** yield cross reject-short sale test failure **U431_03** yield cross reject-NBBO trade through **U431_04** yield cross reject-unwilling to yield appropriate side **U431_05** yield cross reject-outside crossed NBBO **U431_06** yield cross reject-crossed market **U432_01** cross reject-too late for cash settlement **U432_02** cross reject-short sale test failure **U432_03** cross reject-NBBO trade through **U432_04** cross reject-outside crossed NBBO **U432_05** cross reject-crossed market **U432_06** cross reject-CHX trade through **U432_07** cross reject-CHX lock-insufficient size out **U432_09** Cross Reject - Price is outside the band **U432_10** For cross order rejected price at trade-at **U433_01** order reject-outside crossed market NBBO **U433_02** order reject-crossed market **U433_03** order cancel-unable to display remaining volume **U433_04** FOK/IOC Cancel-No Match Opportunity **U436_01** midpoint cross reject-market crossed **U436_02** midpoint cross reject-market halted

Field Name	Data Type	Description	
cancelReason ***(continued)***		**U437_01**	order cancel-TIF expired
		U441_01A	reject incoming order-NBBO trade through
		U441_01B	cancel resting undisplayed order-NBBO trade through
		U441_02	Post Only Cancel
		U441_03	Quote Only
		U441_05	order was canceled because received reject message from away market
		U441_06	SSH Violation
		U441_07	New incoming order get canceled because of order's limit price cross price band (reserved, un-displayed order)
		U441_08	Resting order get canceled because of order's limit price cross price band (reserved, un-displayed order)
		U441_09	Order was canceled because of stale order.
		U450_01	cancel order activity
		U450_03	cancel reject-order not found
		U451_01	cancel change reject-market halted
		U451_02	cancel change-cancel original order
		U451_06	cancel change reject-order not open
		U451_08	cancel change reject-order not found
		U451_11	Reject cancel replace to MKT of DAY order
		U480_02	order canceled on halt
		U482_02	close time expiration-cancel order activity
		U482_05	manual close-cancel order activity
		U482_06	Order gets canceled because of trading pause.
		U485_05	Manual Open-Cancel Opening Crosses
		U485_06	Primary Quote Open-Cancel Opening Crosses
		U490_02	open timer expiration-cancel opening cross order activity
		U491_02	firm disconnect-cancel order activity
		U495_01	ME DAS Order Cancel on Disconnect
		U496_01	ME DAS Order Cancel on DAS Instruction
		U497_01	Manual Unsolicited Order Cancel
		U498_01	Unsolicited cancel because of MTP Cancel Incoming (N)
		U498_02	Unsolicited cancel because of MTP Cancel Resting (O)
		U498_03	Unsolicited cancel of the incoming order because of MTP Cancel Both (B)
		U498_04	Unsolicited cancel of the resting order because of MTP Cancel Both (B)
		U499_01	Unsolicited Cancel or Reject because Kill Switch Flag is ON
		U499_02	Unsolicited cancel because of Kill Switch Cancel Request
		U900_03	ME receives an Order Cancel from ORS
		U900_05	ME receives an Order Reject from ORS
		U900_06	ME receives an internal Order Reject from ORS
		Allowed Values: IEX	
		AdminCancel	Order was administratively canceled
		ExceededMaxSnapshots	Cancel sent by router when orders are not filled within time constraint
		IexOrderCollar	Order cannot be executed outside of collar

Field Name	Data Type	Description
cancelReason ***(continued)***		boundaries **InvalidBookPrice** Order cannot be validly priced **InvalidOrderQty** Invalid quantity for market maker peg order **MPIDDisabled** **OrderExceedsLimit** Order canceled because of constraints on IEX router **OrderSizeLessThanMinQty** Order with Minimum Quantity can no longer be satisfied **RouterConstraint** Routable Order cannot be routed outside of collar boundaries **SelfTradePrevention** Order Canceled by SelfTradePrevention **UnmatchedIneligibleToRest** Unmatched order, ineligible to rest on IEX **PostOnlyCancelled** Post only order cancelled, not subject to display-price sliding **Allowed Values: Nasdaq - PHLX, NOM, NOBO** **1** AUTOPURGE **2** POD **3** FIRM **4** REASSIGN **5** HALT **6** AIQ **7** MANUPURGE **8** OPENPURGE **9** REPRICE **10** SUSPEND **11** LIQUIDITY TAKER **12** RAPID FIRE VOL **13** ZAP DELETE **14** KILLSWITCH AUTO **15** KILLSWITCH CMD LINE **16** KILLSWITCH TRADEINFO **17** notPermitted **18** badStopPrice **19** systemClosed **20** invalidDisplay **21** invalidType **22** invalidFirm **23** invalidClearing **24** halt **25** invalidTime **26** invalidCross **27** invalidMpid **28** invalidMinSize **29** alreadyOpened **30** restrictedSymbol **31** closeCross **32** invalidSymbol

Field Name	Data Type	Description
cancelReason ***(continued)***		**33** testmode **34** invalidPrice **35** tiedToStockNotAllowed **36** invalidSize **37** limitTooDeep **38** featureNotSupported **39** systemError **40** invalidAttribute **41** suspend **42** notFreeTrading **43** nbboTooWide **44** changeContractsNoOrder **45** changeContractsInvalid **46** reentry **47** killswitch_reentry **48** postOnlyReprice **49** undLULD **50** invalidPreOpenIoc **51** userCancel **52** ioc **53** timeout **54** unsolictedOutReentry **55** routeRequest **56** staleOrder **57** sppLimit **58** auctionInProgress **59** engineCancel **60** tooLateToAct **61** noAuction **62** invalidTIF **63** aonNotAllowed **64** bboCross **65** purge **66** orderExpired **67** aiq **68** cnbboLimit **69** noBbo **70** mktOrder **71** treasuryOptionsNotAllowed **72** openingCancel **73** executionNotPossible **74** badCapacity **75** optionNotOpen **76** openDelay **77** liquidityTaker **78** killSwitch **79** adminCancel

Field Name	Data Type	Description
cancelReason *(continued)*		**80** systemCancel **81** brokerOption **82** invalidCrossSurrender **83** cod **84** eodCancel **OTHER** Other **Allowed Values: Nasdaq - PHLX, NOM** **85** missingClearingAccount **86** invalidStrategy **87** undReentry **88** invalidSelfReplenishVolume **Allowed Values: Nasdaq – NOBO, Mercury, GEMX** *in addition to values defined above* **1017** KILLSWITCH_USER **1018** notPermitted **1020** systemClosed **1021** invalidDisplay **1022** invalidType **1023** invalidFirm **1024** invalidClearing **1025** halt **1026** invalidTime **1027** invalidCross **1028** invalidMpid **1029** invalidMinSize **1030** alreadyOpened **1031** restrictedSymbol **1032** closeCross **1033** invalidSymbol **1034** testmode **1035** invalidPrice **1036** tiedToStockNotAllowed **1037** invalidSize **1038** limitTooDeep **1039** featureNotSupported **1040** systemError **1041** invalidAttribute **1042** suspend **1043** notFreeTrading **1044** nbboTooWide **1045** changeContractsNoOrder **1046** changeContractsInvalid **1047** reentry **1048** killswitchReentry **1049** postOnlyReprice

Field Name	Data Type	Description
cancelReason ***(continued)***		**1050** undLULD **1051** invalidPreOpenIoc **1052** userCancel **1053** ioc **1054** timeout **1055** unsolictedOutReentry **1056** routeRequest **1057** staleOrder **1058** sppLimit **1059** auctionInProgress **1060** engineCancel **1061** tooLateToAct **1062** noAuction **1063** invalidTIF **1064** aonNotAllowed **1065** bboCross **1066** purge **1067** orderExpired **1068** aiq **1069** cnbboLimit **1070** noBbo **1071** mktOrder **1072** treasuryOptionNotAllowed **1073** openingCancel **1074** executionNotPossible **1075** invalidCapacity **1076** optionNotOpen **1077** openDelay **1078** liquidityTaker **1079** killswitchPurge **1080** adminCancel **1081** systemCancel **1082** brokerOption **1083** invalidSide **1084** invalidSpread **1085** invalidAuctionType **1086** invalidFormat **1087** frozen **1088** requestPending **1089** cancelUp **1090** cancelDown **1091** postOnlyTaker **1092** invalidState **1093** tooManyAuctions **1094** invalidAuctionParams **1095** rejectedReplace **1096** massCancel

Field Name	Data Type	Description
cancelReason ***(continued)***		**1097** invalidReprice **1098** price **1099** size **1100** nbboLimit **1101** impliedExec **1102** tooManyImplieds **1103** complexInstrExists **1104** exceededMaxComplexInstr **1105** firmExceededMaxComplexInstr **1106** invalidPtaContracts **1107** invalidMatchId **1108** invalidTradeId **1109** invalidCrossId **1110** invalidClientId **1111** dnttNotAllowed **1112** instrumentClosed **1113** atrLimitReached **1114** invalidISO **1115** invalidStepupPrice **1116** threeTickLimitReached **1117** pending **1118** pennyNbboRestriction **1119** invalidDntt **1120** invalidInstrType **1121** invalidOrderType **1122** invalidALO **1123** invalidFlashInst **1124** invalidPrefParty **1125** invalidReserveInfo **1126** invalidPersist **1127** invalidShortSaleInd **1128** invalidProduct **1129** invalidScope **1130** invalidOpenClose **1131** invalidToken **1132** invalidKillAction **1133** invalidLegCount **1134** invalidLegType **1135** invalidLegRatio **1136** invalidCrossType **1137** prefNotAllowed **1138** orderNotFound **1139** actionNotAllowed **1140** instrumentState **1141** qccNotAllowed **1142** qccWithStockNetPriceNotAllowed **1143** qccWithMultiOptLegNotAllowed

Field Name	Data Type	Description
cancelReason ***(continued)***		**1144** invalidDestination **1145** maxRoutesAttempted **1146** destinationNotAvailable **1147** minQtyNotSatisfied **1148** sorRespTimeout **1149** invalidAllocSplits **1150** qccWithStockPriceNotAllowed **1151** tooManyStockTradeAttempts **1152** notTob **1153** cod **1154** poolExhausted **1155** eodCancel **1521** invalidStrategy **1522** undReentry **1523** invalidSelfReplenishVolume **Allowed Values: Nasdaq – NOBO** *in addition to values defined above* **1019** InvalidStopPrice *(NOBO Only)* **1187** rfaReentry *(NOBO Only)* **Allowed Values: Nasdaq – Mercury, GEMX** *in addition to values defined above* **1019** badStopPrice **1156** unAuthorizedGiveup **1157** invalidTriggerId **1158** invalidAccount **1159** invalidAccountNoKill **1160** invalidAccountFirm **1161** beforeGtc **1162** afterNothingDone **1163** invalidRoutingStrategy **1164** invalidTargetFirm **1165** time **1166** minReserveOrderNotFullfilled **1167** closingCancel **1168** portRateBreached **1169** invalidTraderId **1170** stopOrderMissingPreviousTradePrice **1171** stopPriceOnlyAllowedForStopOrder **1172** firmSuspended **1173** traderSuspended **1174** portSuspended **1175** invalidInvestmentDecision **1176** invalidExecutionDecision **1177** invalidDea **1178** invalidPartyRoleQualifier **1179** instrumentExpired **1180** invalidBrokerPct **1181** invalidExecutionSourceCode **1182** prmGroupBlocked **1183** prmLimitsMissing **1184** prmGroupProductBlocked **1185** prmMaxOrderVolume

Field Name	Data Type	Description
cancelReason *(continued)*		**1186** prmMaxOrderValue **1188** maxOrderValue **1189** invalidPrmGroup **1190** prmProductOpenOrderVol **1191** prmProductOpenDelta **1192** prmProductOpenVega **1193** prmProductTradedVol **1194** prmProductTradedDelta **1195** prmProductTradedVega **1196** prmProductTotalVol **1197** prmProductTotalDelta **1198** prmProductTotalVega **1199** firmExceededMaxQuoteRequest **1200** circuitBreaker **1201** quoteRequestInProgress **1202** invalidEvent **1203** invalidMatchEventId **1205** invalidRfaInstruction **1206** rfaInstructionWithRfaId **1207** tobRepriced **1208** invalidPrmLimit **1209** invalidPrmActionBlock **1210** prmGroupUnblocked **1211** prmProductUnblocked **1212** missingClearingAccount **1213** free_10001 **1214** orej_system_error **1215** orej_duplicate_order_id **1216** orej_invalid_time_for_acceptance **1217** orej_not_open_for_trading **1218** orej_unacceptable_volume **1219** orej_invalid_auction_response_attribute **1220** orej_limit_too_far_below_bid **1221** orej_limit_too_far_above_ask **1222** orej_giveup_override_not_allowed **1223** orej_aon_replace_not_allowed **1224** orej_opg_after_opening **1225** orej_off_floor_acct_not_allowed **1226** orej_invalid_volume **1227** orej_mkt_is_invalid **1228** orej_fok_is_invalid **1229** orej_auction_response_not_allowed **1230** orej_post_only_reprice **1231** free_10019 **1232** free_10020 **1233** free_10021 **1234** orej_invalid_limit_price **1235** orej_invalid_stop_price **1236** orej_buy_stop_lteq_bid **1237** orej_sell_stop_gteq_ask **1238** free_10026 **1239** orej_mm_must_be_limit **1240** orej_firm_must_be_limit **1241** orej_bd_must_be_limit **1242** free_10030 **1243** orej_aon_not_allowed_for_mm **1244** orej_aon_not_allowed_for_firm **1245** orej_aon_not_allowed_for_bd **1246** free_10034 **1247** free_10035

Field Name	Data Type	Description
cancelReason *(continued)*		**1248** free_10036 **1249** free_10037 **1250** free_10038 **1251** orej_missing_account_id **1252** free_10040 **1253** free_10041 **1254** orej_restricted_option **1255** orej_invalid_open_close **1256** orej_mm_only **1257** orej_must_be_straight_cancel **1258** orej_target_not_found **1259** orej_target_cancel_pending **1260** orej_target_filled **1261** orej_target_cancelled **1262** free_10050 **1263** orej_target_not_open **1264** free_10052 **1265** orej_cancel_buy_sell_mismatch **1266** orej_cancel_symbol_mismatch **1267** orej_repl_symbol_mismatch **1268** orej_cancel_volume_mismatch **1269** orej_cancel_price_mismatch **1270** orej_cancel_origin_mismatch **1271** orej_cancel_mm_mismatch **1272** free_10060 **1273** free_10061 **1274** free_10062 **1275** orej_cancel_bad_leaves_volume **1276** free_10064 **1277** orej_missing_mm_badge **1278** free_10066 **1279** free_10067 **1280** orej_mm_badge_not_allowed **1281** free_10069 **1282** orej_broker_option **1283** orej_stale_order **1284** orej_listed_routing_only **1285** orej_in_trading_halt **1286** free_10074 **1287** free_10075 **1288** orej_unknown_clearing_firm **1289** orej_mar_too_many_routes **1290** orej_mar_duplicate_order **1291** orej_mar_exch_direct_not_allowed **1292** orej_mar_exch_direct_cust_only **1293** orej_luld **1294** orej_suspend **1295** orej_killswitch **1296** orej_liquidity_taker **1297** free_10085 **1298** free_10086 **1299** free_10087 **1300** free_10088 **1301** orej_tltc **1302** free_10090 **1303** orej_purge **1304** free_10092 **1305** orej_aiq **1306** orej_reentry_required **1307** orej_nbbo_too_wide

Field Name	Data Type	Description
cancelReason ***(continued)***		**1308** orej_invalid_msg_type **1309** orej_required_tag_missing **1310** free_10098 **1311** free_10099 **1312** free_10100 **1313** orej_invalid_firm **1314** orej_invalid_cross_surrender **1315** orej_invalid_br_seqno **1316** orej_invalid_side **1317** orej_invalid_kind **1318** orej_off_floor_req_exch **1319** orej_off_floor_req_multacc **1320** orej_invalid_multacc **1321** orej_off_floor_req_multiacc **1322** orej_invalid_strike_price **1323** orej_invalid_order_type **1324** orej_invalid_cust_firm **1325** free_10113 **1326** orej_invalid_send_time **1327** orej_invalid_tif **1328** free_10116 **1329** orej_invalid_aon **1330** orej_iso_aon_is_invalid **1331** orej_opg_co_not_allowed **1332** orej_opg_iso_not_allowed **1333** orej_invalid_qualifier **1334** free_10122 **1335** orej_invalid_orig_mkt **1336** orej_invalid_option_symbol **1337** orej_cancel_cmta_mismatch **1338** orej_cancel_supp_mismatch **1339** orej_cancel_crosstype_mismatch **1340** orej_cancel_openclose_mismatch **1341** orej_cancel_execbroker_mismatch **1342** orej_cancel_fbnum_mismatch **1343** orej_supp_id_too_long **1344** orej_invalid_mm_badge **1345** free_10133 **1346** free_10134 **1347** free_10135 **1348** free_10136 **1349** free_10137 **1350** free_10138 **1351** free_10139 **1352** free_10140 **1353** orej_invalid_strategy **1354** orej_invalid_leg_ratio **1355** orej_duplicate_leg_ref_id **1356** orej_invalid_num_legs **1357** free_10145 **1358** orej_invalid_non_conforming_ratio **1359** orej_price_violates_spp_limit **1360** orej_feature_not_supported **1361** free_10149 **1362** orej_open_delay **1363** orej_preopen_ioc **1364** orej_iso_must_be_limit **1365** orej_invalid_security_type **1366** free_10154 **1367** orej_invalid_cl_order_id

Field Name	Data Type	Description
cancelReason ***(continued)***		**1368** orej_invalid_orig_cl_order_id **1369** orej_invalid_ifi **1370** orej_invalid_exec_inst **1371** orej_invalid_route_inst **1372** orej_iso_opg_is_invalid **1373** orej_poss_dup **1374** free_10162 **1375** free_10163 **1376** orej_invalid_exp **1377** orej_invalid_leg_ref_id **1378** orej_cancel_clearing_mismatch **1379** orej_iso_not_allowed **1380** orej_invalid_handling_inst **1381** orej_opg_stop_limit_not_allowed **1382** orej_auction_eligibility_mismatch **1383** orej_cannot_change_stop_class **1384** orej_exp_day_invalid **1385** orej_invalid_prin_agency **1386** orej_invalid_stock_leg **1387** orej_auction_in_progress **1388** orej_invalid_nwt_price **1389** orej_invalid_auction_id **1390** orej_invalid_cross_specs **1391** orej_straight_cxl_not_allowed **1392** orej_cxl_replace_not_allowed **1393** orej_invalid_num_orders **1394** orej_order_ids_same **1395** orej_must_improve_price **1396** orej_msg_too_late_to_process **1397** orej_no_auction **1398** orej_nbbo_crossed **1399** orej_attribute_mismatch **1400** orej_symbol_not_open **1401** orej_exch_direct_must_be_limit **1402** orej_invalid_max_floor **1403** orej_invalid_min_quantity **1404** orej_invalid_underlying **1405** orej_invalid_risk_request **1406** orej_wait_iso_not_allowed **1407** orej_opg_aon_not_allowed **1408** orej_buy_market_order **1409** orej_bbo_invalid **1410** free_10198 **1411** orej_reserve_not_allowed **1412** orej_postonly_not_allowed **1413** orej_invalid_floor_brk **1414** orej_invalid_priv_ref **1415** orej_invalid_effective_time **1416** orej_invalid_good_til_date **1417** orej_invalid_cross_client_order_id **1418** orej_invalid_num_sides **1419** orej_invalid_display_when **1420** orej_invalid_price_prot_scope **1421** orej_invalid_auction_inst **1422** orej_invalid_stepup_price **1423** orej_invalid_stepup_price_type **1424** orej_invalid_spec_order_type **1425** orej_invalid_exposure **1426** orej_invalid_broker_pct **1427** orej_invalid_price_delta

Field Name	Data Type	Description
cancelReason ***(continued)***		**1428** orej_must_be_limit **1429** orej_must_be_routable **1430** orej_must_persist **1431** orej_must_be_aon **1432** orej_opg_stop_not_allowed **1433** orej_reserve_modification_invalid **1434** orej_invalid_entitlement_req_id **1435** orej_invalid_no_party_entitlements **1436** orej_invalid_list_update_action **1437** orej_invalid_no_party_details **1438** orej_invalid_party_detail_id **1439** orej_invalid_party_detail_role **1440** orej_invalid_id_source **1441** orej_invalid_security_id **1442** orej_invalid_alloc_id **1443** orej_invalid_alloc_trans_type **1444** orej_invalid_trade_date **1445** orej_invalid_no_allocs **1446** orej_invalid_alloc_shares **1447** orej_invalid_no_execs **1448** orej_invalid_exec_id **1449** orej_exec_broker_required **1450** orej_invalid_shares **1451** orej_invalid_display_range **1452** orej_postonly_replace **1453** orej_invalid_maturity_date **1454** orej_invalid_security_exchange **1455** orej_too_many_auctions **1456** orej_mar_cust_limit_qty **1457** orej_mar_cust_limit_notional **1458** orej_mar_cust_limit_agg_qty **1459** orej_mar_cust_limit_agg_notional **1460** orej_invalid_match_id **1461** orej_invalid_pta_account **1462** orej_invalid_pta_contracts **1463** orej_invalid_client_id **1464** orej_preferencing_not_allowed **1465** orej_invalid_stock_leg_giveup **1466** orej_invalid_contra_side_short_sell **1467** orej_pta_not_allowed **1468** orej_qcc_invalid_stock_ratio **1469** orej_cancel_strategy_mismatch **1470** orej_destination_not_available **1471** orej_invalid_underlying_price **1472** orej_invalid_underlying_qty **1473** orej_invalid_rfp_id **1474** orej_invalid_root_parties **1475** away_status_New **1476** away_status_PartiallyFilled **1477** away_status_Filled **1478** away_status_Done **1479** away_status_Canceled **1480** away_status_Replaced **1481** away_status_PendingCancel **1482** away_status_Stopped **1483** away_status_Rejected **1484** away_status_Suspended **1485** away_status_PendingNew **1486** away_status_Calculated **1487** away_status_Expired

Field Name	Data Type	Description
cancelReason ***(continued)***		**1488** away_status_Accepted **1489** away_status_PendingReplace **1490** away_status_Restated **1491** away_status_Trade **1492** away_status_TradeCancel **1493** away_status_TradeCorrect **1494** alloc_status_Accepted **1495** alloc_status_BlockLevelReject **1496** alloc_status_PartialAccept **1497** alloc_status_NotYetProcessed **1498** invalidTimeOfAgreement **1499** invalidTradeReportId **1500** invalidTradeReportRefId **1501** invalidAgencyCross **1502** invalidHandlingInstr **1503** invalidEqualLeg **1504** invalidMinBlockTradeSize **1505** invalidDeferralThreshold **1506** invalidTradePublishIndicator **1507** invalidMaximumTradeReportSize **1508** invalidTradeType **1509** flexInstrExists **1510** invalidCircuitBreakerId **1511** invalidPriceProtectionTableCode **1512** invalidStrikePrice **1513** invalidExpirationDate **Allowed Values: Nasdaq - ISE, GEMINI, Mercury Options** **1** AUTOPURGE **2** POD **3** FIRM **4** REASSIGN **5** HALT **6** AIQ **7** MANUPURGE **8** OPENPURGE **9** REPRICE **10** SUSPEND **11** LIQUIDITY TAKER **12** RAPID FIRE VOL **13** ZAP DELETE **14** KILLSWITCH AUTO **15** KILLSWITCH CMD LINE **16** KILLSWITCH TRADEINFO **17** KILLSWITCH USER **18** notPermitted **19** invalidStopPrice **20** systemClosed **21** invalidDisplay **22** invalidType **23** invalidFirm **24** invalidClearing

Field Name	Data Type	Description
cancelReason ***(continued)***		**25** halt **26** invalidTime **27** invalidCross **28** invalidMpid **29** invalidMinSize **30** alreadyOpened **31** restrictedSymbol **32** closeCross **33** invalidSymbol **34** testmode **35** invalidPrice **36** tiedToStockNotAllowed **37** invalidSize **38** limitTooDeep **39** featureNotSupported **40** systemError **41** invalidAttribute **42** suspend **43** notFreeTrading **44** nbboTooWide **45** changeContractsNoOrder **46** changeContractsInvalid **47** reentry **48** killswitchReentry **49** postOnlyReprice **50** undLULD **51** invalidPreOpenIoc **52** userCancel **53** ioc **54** timeout **55** unsolictedOutReentry **56** routeRequest **57** staleOrder **58** sppLimit **59** auctionInProgress **60** engineCancel **61** tooLateToAct **62** noAuction **63** invalidTIF **64** aonNotAllowed **65** bboCross **66** purge **67** orderExpired **68** aiq **69** cnbboLimit **70** noBbo **71** mktOrder

Field Name	Data Type	Description	
cancelReason *(continued)*		**72**	treasuryOptionNotAllowed
		73	openingCancel
		74	executionNotPossible
		75	invalidCapacity
		76	optionNotOpen
		77	openDelay
		78	liquidityTaker
		79	killswitchPurge
		80	adminCancel
		81	systemCancel
		82	brokerOption
		83	invalidSide
		84	invalidSpread
		85	invalidAuctionType
		86	invalidFormat
		87	frozen
		88	requestPending
		89	cancelUp
		90	cancelDown
		91	postOnlyTaker
		92	invalidState
		93	tooManyAuctions
		94	invalidAuctionParams
		95	rejectedReplace
		96	massCancel
		97	invalidReprice
		98	price
		99	size
		100	nbboLimit
		101	impliedExec
		102	tooManyImplieds
		103	complexInstrExists
		104	exceededMaxComplexInstr
		105	firmExceededMaxComplexInstr
		106	invalidPtaContracts
		107	invalidMatchId
		108	invalidTradeId
		109	invalidCrossId
		110	invalidClientId
		111	dnttNotAllowed
		112	instrumentClosed
		113	atrLimitReached
		114	invalidISO
		115	invalidStepupPrice
		116	threeTickLimitReached
		117	pending
		118	pennyNbboRestriction

Field Name	Data Type	Description
cancelReason ***(continued)***		**119** invalidDntt **120** invalidInstrType **121** invalidOrderType **122** invalidALO **123** invalidFlashInst **124** invalidPrefParty **125** invalidReserveInfo **126** invalidPersist **127** invalidShortSaleInd **128** invalidProduct **129** invalidScope **130** invalidOpenClose **131** invalidToken **132** invalidKillAction **133** invalidLegCount **134** invalidLegType **135** invalidLegRatio **136** invalidCrossType **137** prefNotAllowed **138** orderNotFound **139** actionNotAllowed **140** instrumentState **141** qccNotAllowed **142** qccWithStockNetPriceNotAllowed **143** qccWithMultiOptLegNotAllowed **144** invalidDestination **145** maxRoutesAttempted **146** destinationNotAvailable **147** minQtyNotSatisfied **148** sorRespTimeout **149** invalidAllocSplits **150** qccWithStockPriceNotAllowed **151** tooManyStockTradeAttempts **152** notTob **153** cod **154** poolExhausted **155** eodCancel **156** CLOSEPURGE **157** PRICE_LIMIT **158** ORDER_SIZE **159** SPP_LIMIT **160** EXECUTION_NOT_POSSIBLE **OTHER** OTHER **Allowed Values: Nasdaq - ISE, GEMX** **161** missingClearingAccount **162** invalidStrategy

Field Name	Data Type	Description
cancelReason ***(continued)***		**163** undReentry **164** invalidSelfReplenishVolume **Allowed Values: MEMXOP** **0** Other - This order was canceled for some other reason not listed. **1** UserRequestedCancel - The client sent a OrderCancelRequest or OrderMassCancelRequest for this order. **5** ExecutionPriceCollar - The price of the order fell outside execution price collar bands. **6** Halted - The market on the order's security was halted. **7** ExchangeSupervisory - Operational or supervisory actions taken by MEMXOP resulted in the cancellation of this order. **8** OrderExpired - The order was sent with an expiration time and had the "good for time" time in force set, and the supplied expiration time passed. **10** MatchTradePrevention - This or another associated order's specified self trade prevention behavior triggered the cancellation of this order. **13** ParticipantDisconnect - The participant directed that their orders should be canceled when the trading system detects a disconnection, and the participant disconnected. **14** OrderNotBookable – The Order is not of bookable type (this may include market orders, IOC, FOK, etc). **17** FirmDisabled – The order was cancelled because the firm was disabled. **18** EFIDDisabled – The order was cancelled because the EFID was disabled. **19** AccountDisabled – The order was cancelled because the Account was disabled. **Allowed Values: Nasdaq Equities – NSDQ, PSX, BX** **1** User requested cancel. Sent in response to a Cancel Order Message or a Replace Order Message **2** Immediate or Cancel order. **3** Timeout. The Time In Force for this order has expired **4** Supervisory. **5** This order cannot be executed because of a regulatory restriction **6** Self-Match Prevention. **7** System cancel. **8** Cross-canceled. Non-bookable cross orders that did not execute in the cross. **9** Order canceled due to insufficient quantity **10** This order cannot be executed because of Market Collars **11** Halted. The on-open order was canceled because the symbol remained halted after the opening cross-completed. **13** Closed. Any DAY order that was received after the closing cross is complete in a given symbol will receive this cancel reason. **15** Administrative cancel **16** Post Only Cancel. This Post Only order was canceled because it would have been price slid for NMS. **17** Post Only Cancel. This Post Only order was canceled because it would have been price slid due to a contra side displayed order on the book **18** Direct Listing with Capital Raise amt exceeded

Field Name	Data Type	Description
cancelReason ***(continued)***		**19** Open Protection **20** Discretion **21** Cross Supervisory **22** Managed Orders **23** Cancel On Disconnect **24** User Modified **25** Oddlot **26** User Replaced **27** User Split **28** System Downtick **29** Market Collar **30** Late Cancel **31** Quality Cancel **32** Forced Cancel **33** Reject Cancel **ADMIN** for an administrative cancel **FEATURE** in the service of a customer-requested feature **OTHER** OTHER
capacity	Choice	*Event(s): Order Accepted Event, Order Route Event, Order Modified Event, Order Trade Event, Order Fill Event, Order Modify Route Event, Order Restatement Event* Specifies the capacity of a given order or side of a trade. **Allowed Values** **Agency** **Principal** **RisklessPrincipal** **Allowed Values: NYSE Equities** **ErrorAccount**
carryoverFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates that the trade transaction was carried over (not accepted/declined by the contra firm on T+0) for processing. **Allowed Values** **C** Carryover
clearingFirm	Text (10)	*Event(s): Stock Leg Order Event, Stock Leg Fill Event* The Member Alias of the clearing firm.

Field Name	Data Type	Description
clearingFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Clearing and matching specifications of the trade transaction. **Allowed Values** **A** Nasdaq AGU for Clearing **C** Customer (no matching, no clearing) **G** Automatic Give Up (Auto Lock-in and Clearing) **N** No Clearing **Q** QSR (no matching, no clearing) **R** Risk Update Only (not sent to clearing) **S** Self-clearing (no matching, no clearing) **U** AGU Clearing, Non-risk Eligible **Y** Clearing ACT Only: **L** Do not match; send to clearing (locked-in) received via external system interface only. **Z** Do not match; send to clearing (locked-in).
clearingNumber	Text (20)	*Event(s): Order Trade Event, Order Fill Event, Stock Leg Fill Event* DTCC clearing number reported for each side of a stock trade or for the reporting side of a fill event.
clearingPrice	Price	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Trade price inclusive of commissions. This information is only currently available for reported trades to the Nasdaq TRF.
cmtaFirm	Alphanumeric(8)	*Event(s): Simple Option Order Accepted Event, Option Order Modified Event, Option Trade Event, Post Trade Allocation Event, Options Order Restatement Event* The OCC number of the CMTA firm (only valid for CMTA trades).
complexOptionID	Text (40)	*Event(s): Simple Option Order Accepted Event, Option Order Adjusted Event, Stock Leg Order Event, Option Order Modified Event, Stock Leg Modified Event, Option Route Event* When present in an event, the complexOptionD will contain the same value as the optionID field from the Complex Order Accepted event to which this event is associated.
complexOrderID	Text (40)	*Event(s): Simple Option Order Accepted Event, Option Order Adjusted Event, Stock Leg Order Event, Option Order Modified Event, Stock Leg Modified Event, Option Route Event* When present in an event, the complexOrderID identifies the complex option order that is the parent order for an leg orders. Note that this will be the same value as the orderID field from the Complex Order Accepted event.
contraClearingNumber	Unsigned	*Event(s): Order Fill Event* DTCC clearing number for contra side of a trade.
contraControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Number for the contra party.

Field Name	Data Type	Description
contraEntryFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates that the contra party is the only side that reported the trade. **Allowed Values** **O** Contra Entry
contraExecutingMpid	Member Alias	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* MPID of the contra-side executing party.
contraExecutionTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the contra party reported that the execution took place.
contraReportDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date the contra party reported the trade.
contraReportingObligationFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies if the contra-side firm had the reporting obligation for the trade under FINRA trade reporting rules. **Allowed Values** **Y** Contra Firm Has Reporting Obligation **N** Contra Firm Does Not Have Reporting Obligation
contraReportTime	Time	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Time the contra party reported the trade.
contraReportTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the contra party reported the trade.
contraSideBranchSequenceIdentifier	Text (20)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Branch/sequence number of the contra-side firm.
contraSideCapacityCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Capacity of the contra-side firm. **Allowed Values** **A** Agency **P** Principal **R** Riskless Principal
contraSideClearingNumber	Unsigned	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Clearing number of the firm that cleared the trade for the contra-side firm.
contraSideReportingMpid	Member Alias	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* MPID of the contra-side firm that reported the trade.
contraSideShortSaleCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies a short sale by the contra firm and indicates the type of short. **Allowed Values** **SS** Short Sale **SX** Short Sale Exempt

Field Name	Data Type	Description
contraSubmittingEntityId	Text (4)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates the entity that initiated the submission. For a FINRA-initiated submission on behalf of the firm, this will be 'FNRA'. Otherwise, for a firm-initiated submission, it will be the firm MPID. For NC TRF, NQ TRF and NY TRF, this is always NQTC, NQTR or NYTR. For ADF and ORF it is the MPID of the submitting firm.
controlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Unique identifier for the reporting side of each trade transaction.
coverage	Choice	*Event(s): Simple Option Order Accepted Event, Option Order Modified Event, Option Route Event, Complex Option Route (OCOR), Modify Option Route Event, Options Order Restatement Event* Specifies whether an option order is covered or uncovered. Field may also be filled in as unspecified. **Allowed Values** **Covered** **Uncovered** **Unspecified**
cycleDate	Date	*Event(s): All Options Exchange Events, Note (NOTE), Self-Help Declaration (SHD), Supplemental Trade Event (STE)* The Trade Date upon which the trading cycle of an event ends. The trading cycle refers to the period of time when an order is eligible to trade and may include one or more trading sessions.
declaredTimestamp	Timestamp	*Event(s): Self-Help Declaration (SHD)* Date and time self-help was declared.
declineTime	Time	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Time the trade was declined by the contra party.
declineTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the trade was declined by the contra party.

Field Name	Data Type	Description
definedMMDEData	Name Value Pairs	*Event(s): Market Maker Dictionary Entry (MMDE)* A list of key/value pairs, providing machine parseable exchange specific regulatory context data for the Equity Market Maker. **Allowed Values: Nasdaq Equities – NSDQ, PSX, BX** **MMSTATE** Market Maker Status for Intraday Changes. Value must be one of the following **O** Open **C** Closed **W** Withdrawn **E** Excused Withdrawn **S** Suspended **D** Deleted **N** None **Allowed Values: IEX** **MMRegistrationEvent** Registration event codes for intraday Market Maker status changes. When provided, it must be one of the following values (e.g. MMRegistrationEvent=S): **A** Accidental Termination **B** Reinstate Accidental **E** Reinstate Regulatory **R** Reinstate Excused **S** Start of Day **T** Reinstate Voluntary **V** Voluntary Termination **W** Excused Withdrawal **X** Regulatory Termination
definedNoteData	Name Value Pairs	*Event(s): Note (NOTE)* A list of key/value pairs, providing machine parseable data for the notation. The attributes must be defined in this specification. **Allowed Values: Cboe Legacy (C1) Only** *active 3/29/2019 - 10/4/2019* **SubNoteType** Requires a Choice value (e.g SubNoteType=XXX) where XXX must be one of the following choices. **SELECTED** PAR Order Select Time and NBBO at the time **RECEIVED** PAR Order Received Time and NBBO at the time **TRADED** PAR Order Trade Time and NBBO at the time **REPRESENT** PAR Order represent time and NBBO at the time **UID** A unique number assigned by the originating system to identify the row in SBT_ORDER_HIST. The value must be Unsigned (e.g. UID=12345). **RemQty** Quantity remaining after the fill. The value must be Unsigned (e.g., RemQty=700). **RouteSrc** The source of the route as a text field (Text<40>) of workstation name, PAR broker, etc (e.g., RouteSrc=ABC123). **RouteDest** The destination of the route as a text field (Text<40>) of workstation name, PAR broker, etc (e.g., RouteSrc=ABC123).

Field Name	Data Type	Description
definedNoteData ***(continued)***		**RouteSrcType** The location type where the order is routed from. The value is one of the following integer values (e.g., RouteSrcType=3): **0** Unspecified **1** CMI **3** TE **4** PAR **5** BOOTH_OMT **6** CROWD_OMT **7** HELP_DESK_OMT **8** OHS **9** LINKAGE **10** DISPLAY **11** Broker Dealer (Stock orders derived from CPS Cross) **12** Broker Dealer (Stock Orders derived from CPS Market Order Split)
		RouteDestType The location type where the order is routed to. The value is one of the same as described in RouteSrcType.
		RouteRes Indicates the reason for the route. The value is one of the integer values (e.g., RouteRes=7) from the following list: **1** VOLUME_CHECK **2** AUTO_EXECUTION **3** DIRECT_ROUTE **4** ALTERNATE_ROUTE **5** DISCRETIONARY_OR_NH_ORDER **6** ALL_ROUTING_ATTEMPT_FAILED For reroute attempts: **7** HAL_REROUTING **8** REROUTING_TO_SENDER **9** REROUTING_TO_DEFAULT_OMT **10** LINKAGE_ROUTE For PAR print requests: **11** PAR_PRINT_ORDER_INTRA_DAY **12** PAR_PRINT_ORDER_END_OF_DAY **13** PAR_PRINT_CANCEL **14** PAR_PRINT_CANCEL_REPLACE For PAR order reroute TA and TB: **15** MANUAL_REROUTE_ORDER_TA **16** MANUAL_REROUTE_ORDER_TB **17** MANUAL_REROUTE_ORDER_BOOK **18** MANUAL_REROUTE_ORDER_AUCTION **19** CANCEL_FOLLOW_ORDER For PAR order and fill timeouts: **20** MANUAL_ORDER_TIMEOUT **21** MANUAL_ORDER_FILL_TIMEOUT **22** CABINET_ORDER

Field Name	Data Type	Description	
definedNoteData ***(continued)***			**23** SIMPLE_FILL_REJECT **24** COMPLEX_FILL_REJECT **25** CANCEL_REQUEST_ON_RSS **26** NBBO_REJECT **27** TRADE_NOTIFICATION_BUNDLE_TIMEOUT **28** TRADE_NOTIFICATION_ACK_TIMEOUT **29** TRADE_NOTIFICATION_REJECT **30** FILL_REPORT_DROP_COPY **31** CANCEL_REPORT_DROP_COPY **32** PREMIUM_EXCEEDS_REASONABILITY **33** VOLUME_DEVIATION_CHECK_FAILED_ALL_LEVELS **34** VOLUME_DEVIATION_CHECK_PASSED_LEVEL_1 **35** VOLUME_DEVIATION_CHECK_PASSED_LEVEL_2 **36** VOLUME_DEVIATION_CHECK_PASSED_LEVEL_3 **37** CANCEL_REQUEST_ON_FALLBACK **38** TOO_MANY_ROUTES **39** PRODUCT_STATE_ROUTE **40** VOLUME_MAINTENANCE_MISMATCH **41** FORCED_LOGOFF_PAR **42** MANUAL_REROUTE_ORDER_SR **46** MANUAL_REROUTE_ORDER_FR **302** LINKAGE_STALE_EXECUTION
		BBOBP	BBO bid price; the value is of type Price.
		BBOBS	BBO bid size; the value is of type Unsigned.
		BBOAP	BBO ask price; the value is of type Price.
		BBOAS	BBO ask size; the value is of type Unsigned.
		NBBOBP	NBBO bid price; the value is of type Price.
		NBBOBV	NBBO bid exchange volume; the value is of type Unsigned.
		NBBOAP	NBBO ask price; the value is of type Price.
		NBBOAV	NBBO ask exchange volume; the value is of type Unsigned.
		DSMBP	Derived Spread Market bid price; the value is of type Price
		DSMBS	Derived Spread Market bid size; the value is of type Unsigned
		DSMAP	Derived Spread Market ask price; the value is of type Price
		DSMAS	Derived Spread Market: The (Integer)
		BBP	Book bid price; the value is of type Price.
		BBS	Book bid size; the value is of type Unsigned.
		BAP	Book ask price; he value is of type Price.
		BAS	Book ask size; the value is of type Unsigned.
		AuctionType	The type of auction; the value is one of the following integers **0** Auction Unspecified **1** AUCTION_INTERNALIZATION (AIM/Complex AIM) **2** AUCTION_STRATEGY **3** AUCTION_REGULAR_SINGLE **4** AUCTION_HAL **5** AUCTION_SAL **8** AUCTION_DAIM (for Directed AIM)

Field Name	Data Type	Description
definedNoteData *(continued)*		**-4** AUCTION_HALO **-8** AUCTION_NEW_HAL **AucTradeQty** auction trade quantity; the value will be Unsigned **AucEarlyTerm** indicates if an auction ended early; the value is Boolean (true or false) **AuctionID** Optional field of type UNSIGNED **ActTime** The actual time at which activity happened on PAR or ME; the value will be Timestamp **Allowed Values: Cboe Options** *active 10/7/2019 – present* **BBOBP** BBO bid price; the value is of type Price. **BBOBS** BBO bid size; the value is of type Unsigned. **BBOAP** BBO ask price; the value is of type Price. **BBOAS** BBO ask size; the value is of type Unsigned. **NBBOBP** NBBO bid price; the value is of type Price. **NBBOBV** NBBO bid exchange volume; the value is of type Unsigned. **NBBOAP** NBBO ask price; the value is of type Price. **NBBOAV** NBBO ask exchange volume; the value is of type Unsigned. **BBP** Book bid price; the value is of type Price. **BBS** Book bid size; the value is of type Unsigned. **BAP** Book ask price; he value is of type Price. **BAS** Book ask size; the value is of type Unsigned. **SubNoteType** Requires a Choice value (e.g SubNoteType=XXX) where XXX must be one of the following choices. **SELECTED** PAR Order Select Time and NBBO at the time **RECEIVED** PAR Order Received Time and NBBO at the time **TRADED** PAR Order Trade Time and NBBO at the time **REPRESENT** PAR Order represent time and NBBO at the time **UID** A unique number assigned by the originating system to identify the row in SBT_ORDER_HIST. The value must be Unsigned (e.g. UID=12345). **RouteDest** The destination of the route as a text field (Text<40>) of workstation name, PAR broker, etc (e.g., RouteSrc=ABC123). **Allowed Values: NYSE** **Cabinet** **FLEX** **FLEXPCT** **FloorTrade** **FloorTradeNamesLater** **FloorTradeNamesLaterAllocation** **Allowed Values: NYSE Equities** **AucPrc** Price on the auction request (AucPrc=<price value>) **DMM** Designated Market Maker (DMM=<MPID>) **Allowed Values: BOX**

Field Name	Data Type	Description
definedNoteData *(continued)*		**ST** Requires a choice from the following list: **InOrderBook** **Executed** **Exposed** **ToOla** **Directed** **CancelPending** **Eliminated** **TraderCancelled** **EliminatedOutOfLimit** **EliminatedByCircuitBreaker** **EliminatedOnDisconnection** **EliminatedByMarketControl** **EliminatedDueToUnpricedLeg** **EliminatedDueToTradingRestriction** **CancelledBySupervisor** **Received** **EliminatedDueToTradeLimitExceeded** **EliminatedDueToTradeActivityLimitExceeded** **EliminatedDueToMaximumNbTriggersLimitExceeded** **EliminatedDueToDrillThroughProtection**
desiredLeavesQty	Unsigned	*Event(s): Order Cancel Route Event, Option Cancel Route Event* The desired number of shares remaining in the order after the cancel request has been issued for a routed order. A value of zero indicates a full cancel.
displayPrice	Price	*Event(s): Order Accepted Event, Order Modified Event, Order Restatement Event, Simple Option Order Accepted Event, Option Order Modified Event, Options Order Restatement Event* The displayed price for an order.
displayQty	Unsigned	*Event(s): Order Accepted Event, Order Route Event, Order Modified Event, Order Modify Route Event, Order Restatement Event, Simple Option Order Accepted Event, Stock Leg Order Event, Option Order Modified Event, Stock Leg Modified Event, Option Route Event, Modify Option Route Event, Options Order Restatement Event* The displayed quantity for an order.
eventTimestamp	Timestamp	*Event(s): All* eventTimestamp generally refers to when an event occurred, however this is subjective depending on the event. Refer to the events definitions to see what this timestamp represents within the context of that event.
exchange	Exchange ID	*Event(s): All Stock Exchange Events, All Options Exchange Events, Supplemental Trade Event (STE), Reject Message Event (RME), Equity Best Bid and Offer Event (EBBO)* The exchange ID of the exchange associated with the event being reported. Refer to each individual event definition for more specific details.
exchangeInternalID	Text (40)	*Event(s): Reject Message Event (RME)* The internal ID assigned to the order or quote by the exchange.
exchOriginCode	Choice	*Event(s): Simple Option Order Accepted Event, Complex Option Order Accepted Event, Option Order Modified Event, Internal Complex Option Route Event, Option Trade Event, Options Order Restatement Event, Post Trade Allocation Event*

Field Name	Data Type	Description
exchOriginCode ***(continued)***		Exchange-specific codes that specify the origin of an order. CAT will map all of these exchange-defined codes to either C - Customer, F - Firm, or M - Market Maker internally. Only the exchange specific codes as defined below need to be included in this field. Below are the accepted values for each exchange, with their description, and their mapping to C, F, or M in CAT in parentheses. Note that some values are marked as "C/M," C/M will map to customer unless an order has mktMkrSubAccount, when it will map to M. **Allowed Values: Cboe Legacy (C1)** *active 3/29/2019 – 10/4/2019* **B** Broker Dealer (C) **C** Customer (C) **D** Customer Floor Broker Workstation (C) **E** Customer Internal (C) **F** Firm (F) **H** Firm Internal (F) **I** In Crowd Market Maker (M) **J** Firm Floor Broker Workstation (F) **K** Broker Dealer Floor Broker Workstation (C) **L** B/Ds that are billed as 'Firm' but clear in the 'C' range at OCC (C) **M** Market Maker (M) **N** Away Market Maker (M) **R** Broker Dealer Internal (C) **U** MM from FBW (C/M) **W** Broker Dealer Floor Broker Workstation (C/M) **X** Customer BD (C/M) **Z** N,Y from FBW (C/M) **Allowed Values: NYSE Options** **C** Customer (C) **F** Firm (F) **BD** Broker Dealer (C/M) **M** Market Maker (M) **P** Professional Customer (C) **Allowed Values: Cboe** **B** Broker Dealer (C) **C** Customer (C) **F** Firm (F) **J** Joint Back Office (F) **L** Non TPH Affilliate (C) **M** Market Maker (M) **N** NonRegMarketMaker (M) **U** ProCustomer (C) **Allowed Values: BOX** **6** Public Customer (C)

Field Name	Data Type	Description
exchOriginCode ***(continued)***		**7** Broker Dealer (F) **8** Market Maker (M) **W** Broker Customer (C) **X** Away Affiliated Market Maker (M) **T** Professional Customer **Y** Away Broker or Floor Broker (F) **Z** Away Market Maker or Floor Market Maker (M) **V** Away Broker Customer or Floor Broker Customer (C) **Allowed Values: MEMXOP** **1** Customer (C) **2** Professional Customer (C) **3** Broker Dealer (F) **4** Broker Dealer Customer (C) **5** Firm (F) **6** Market Maker (M) **7** Away Market Maker (M) **Allowed Values: MIAX** **1** Market Maker (M) **2** Away MM (M) **3** Broker Dealer (F) **4** Firm (F) **5** Pri Customer (C) **6** Non Pri Customer (C) **Allowed Values: MIAX Pearl** **1** Market Maker (M) **2** Away MM (M) **3** Broker Dealer (F) **4** Firm (F) **5** Pri Customer (C) **6** Non Pri Customer (C) **Allowed Values: MIAX Emerald** **1** Market Maker (M) **2** Away MM (M) **3** Broker Dealer (F) **4** Firm (F) **5** Pri Customer (C) **6** Non Pri Customer (C) **Allowed Values: MIAX SPHR** **1** Market Maker (M) **2** Away MM (M) **3** Broker Dealer (F) **4** Firm (F)

Field Name	Data Type	Description
exchOriginCode ***(continued)***		**5** Pri Customer (C) **6** Non Pri Customer (C) **Allowed Values: NASDAQ Options - NOBO, PHLX, NOM, ISE, GEMX, MRX** **1** Customer (C) **2** Firm (F) **3** Floor MM (M) **4** Off Floor MM (M) **5** Broker Dealer (C) **6** Professional Customer (C) **7** Proprietary Customer (C) **8** Retail Customer (C) **9** JBO (F) **10** Broker Dealer Firm (F)
executingFirm	Alphanumeric(8)	*Event(s): Simple Option Order Accepted Event, Option Order Modified Event, Option Trade Event, Options Order Restatement Event* The OCC number of the executing firm.
executionCodes	Name / Value Pairs	*Event(s): Order Trade Event, Order Fill Event, Trade Correction Event, Option Trade Event, Stock Leg Fill Event, Options Trade Correction Event* Codes that provide a way to augment executions with specific information about the execution. The Execution Codes field has the same formatting as Order Handling Instructions, where zero or more codes can be entered to provide additional execution information, like where a trade may have been executed on the floor. Each code is separated by a single pipe symbol (ASCII decimal 124, hex 7C). Codes which require a value will include that value immediately after the code Field Name and a single equal sign (ASCII decimal 61, hex 3D). All instructions that apply to the order are to be included. **Allowed Values: Boolean** *presence indicates truth* **NonMediaTrade** Presence of this instruction on an EOT event indicates that the details of this particular record either went to tape as part of a single-priced trade process, such as an opening or closing auction; or that the details of this event did not go to tape. The use of an EOF event implies that the details were not sent to tape, and this Name/Value pair should not be used on EOF events. **FLOOR** Presence of the value on an Options Trade (OT) event indicates that the trade occurred on the Exchange Floor in Open Outcry. Presence of the value on an Equity Trade (EOT) event indicates that the trade occurred on the Exchange Floor between a Designated Market Maker and/or a single or multiple Brokers. **PCTPX** Indicates that the execution price provided in the price field is specified as a percentage of a benchmark price. For example, price=94.5 indicates that the final dollar price will be 94.5% of a benchmark price. A trade correction (OTC or ETC) event with the final dollar price is required. **Allowed Values** **AUC** If the trade happened as part of an auction, this code identifies the auction by name (e.g., AUC=CROSS) **ASOF** The trade is being reported as- of another date. This option requires a Date value (e.g. ASOF=20171218).

Field Name	Data Type	Description
executionCodes ***(continued)***		**BulkTradeType** Value identifiying the aggregate print sent to tape or an opening, re-opening or closing trade as well as the underlying execution reports for all of the orders that executed as part of the single-priced trade event. This value must be reported for all single-priced trade events. Valid Values: **O** Opening single-priced trade **H** Re-opening single-priced trade **I** IPO single-priced trade **C** Closing single-priced trade **R** Continuous market trade with multiple parties on one or both sides **N** Not related to a single-priced trade event (this value is implied if this Name/Value pair is not provided) **BulkTradeID** Value that links together the aggregate print sent to tape or an opening, re-opening or closing trade and the underlying execution reports for all of the orders that executed as part of the single-priced trade event. **childOrderID** Requires value, e.g. childOrderID = 123456789 **CORR** Boolean. Indicates that the trade was entered as a correction to a previously reported trade. **MOOTLINK** Required for manual floor trades reported to FINRA CAT as MOOT events. This field contains the identifier that will be used for linkage to the IM MOOT event. **NOBUYID** Indicates that there is neither a quoteID nor an orderID associated with the buy side of the trade. **NOSELLID** Indicates that there is neither a quoteID nor an orderID associated with the sell side of the trade. **PCTP** Executions for FLEXPCT orders are reported, with the price as the final dollar value of the trade. However, the price was determined as a percentage execution. The original trade percentage value is reported using the PCTP execution code, which requires a Numeric(10,8) value, where 94.5% would be reported as PCTP=94.5. **PCTO** Executions for FLEXPCT trades are reported using the optionID of the percentage product. However, the final execution happens with a different optionID that is not percentage based. This final optionID is a Text<40> field, and is reported in the trade with the PCTO execution code (e.g., PCTO=OPTIONID1234). **PRVRSL** Boolean. Indicates that the trade was entered to reverse a partial quantity of a previously reported trade. **REFTRADEID** Required for trades marked as a reversal, partial reversal or correction of a previously reported trade, this field contains the trade being referenced. REFTRADEID must reference a previously reported trade, or a previously reported trade correction that has a matching tradeID. **REFTRDDATE** Required for trades marked as a reversal, partial reversal or correction of a previously reported trade, this field requires a Date value of the original trade date (e.g., REFTRDDATE =20210305) **RVRSL** Boolean. Indicates that the trade was entered to reverse a previously reported trade in its entirety. **Allowed Values Cboe Legacy (C1)** *active 3/29/2019 – 10/4/2019* **TradeType** This code requires a choice value (e.g., TradeType=N) where N

Field Name	Data Type	Description
executionCodes ***(continued)***		is a value from the following list: **B** Blocktrade **R** Regular Trade **F** Intermarket Sweep **L** No Print Linkage Trade **M** Manual Trade **P** Par Trade **X** Cross Product Leg Trade **S** Cross Product Cross Trade **I** Cross Product AIM Cross Trade **H** Handheld Trade **Q** Par to Market Maker Trade **1** Regular trade reversal **2** No Print Linkage Trade Reversal **3** No Print Linkage Trade Manual **T** Two-Day Trade **TradeSource** This code requires a choice value (e.g., TradeSource=PAR) where the value is one of the three following choices: **PAR** **System** **Manual** **FirmTradeRptTime** Shows the Firm Trade Report Time (applies to Block trade and manual trades, time the firm/market maker reports the floor trade), requires a timestamp (e.g., FirmTradeRptTime=20170108T023000.123456789). Note that the timestamp must be in the CAT timestamp format described in section 1.5 of the tech specs **FirmTradeTime** Shows the Firm Trade Time - applies to manual trades - Market Makers have an option to specify when they did the trade on the floor. Requires a timestamp (e.g., FirmTradeTime=20170108T023000.123456789). Note that the timestamp must be in the CAT timestamp format described in section 1.5 of the tech specs **TradeRptTime** Shows the Tape Report Time (when the system reports to OPRA i.e. when the GUI user hits the send button) applies to manual and block trades only. Requires a timestamp. (e.g., TradeRptTime=20170108T023000.123456789). Note that the timestamp must be in the CAT timestamp format described in section 1.5 of the tech specs **EndorseTime** (Floor only) In the case of a Names Later transaction, $TIME is the time this execution was endorsed by this side. If not specified, assume to be equal to FirmTradeRptTime.(e.g.EndoreTime= 111500.123456789.)This timestamp must be in the CAT time format described in section 1.5 of the tech specs **NamesLater** (Floor only) If present, this specifies that this side is reporting Names Later. EndorseTime will differ from FirmTradeRptTime. (e.g. NamesLater=Y) **BBOBP** CBOE BBO Bid Price at the time of the trade. Requires a price value. (e.g., BBOBP=12.25) **BBOBS** CBOE BBO Bid Size at the time of the trade. Requires an integer value. (e.g., BBOBS=400) **BBOAP** CBOE BBO ask price at the time of the trade. Requires a price

Field Name	Data Type	Description
executionCodes ***(continued)***		value. (e.g., BBOAP=12.50) **BBOAS** CBOE BBO ask size at the time of the trade. Requires an integer value. (e.g., BBOAS=200) **BDATE** Shows the business date. Requires a date value expressed as YYYYMMDD (e.g., BDATE=20170112). **FloorActivityType** Types of Floor Execution; Choice fields: **Unspecified** **TradeWithAllExecution** **TradeWithBookExecution** **SwapExecution** **COAExecution** **InCrowdExecution** **RepresentedInCrowd** **TradeInitiatedInCrowd** **TradeEndorsement** **Allowed Values: Cboe** **DACClosePrice** Closing price for the underlying. Accepts a price value expressed as ##.#### (e.g. 12.3456). **DeltaRefPrice** The value of the underlying as known by the submitter of the order. Accepts a price value expressed as XX.XXXX (e.g. 12.3456). **DeltaValue** The multiplier applied to the difference between the referencePrice and the closing price of the option's underlying value (specified per leg in the case of a complex order). Accepts a value from -1.0000 to 1.0000. **FirmTradeRptTime** Shows the Firm Trade Report Time (applies to Block trade and manual trades, time the firm/market maker reports the floor trade), requires a timestamp (e.g., FirmTradeRptTime=20170108T023000.123456789). Note that the timestamp must be in the CAT timestamp format described in section 1.5 of the tech specs **INTLIQ** Liquidity classification internal to Cboe. Requires a choice value (e.g., INTLIQ=X) from the following list: **A** added **R** removed **X** routed **B** both order washed/removed some liquidity then got booked **D** externally removed **c** conditionally added **C** auction **Q** options wait order **P** RemovedPending **SUBLIQ** Cboeinternal subliquidity indicator. This is filled in on executions once the code offering the best price to the member is selected. Requires a choice value (e.g., SUBLIQ=N) from the following list: **A** halt auction **b** AIM – Automated Improvement Mechanism **B** SUM (Options only – step up auctions mechanism)

Field Name	Data Type	Description
executionCodes *(continued)*		**c** Cboe Market Close **C** close auction **D** dark book **E** retail price improvement (BYX Equities: Retail Order vs. Retail Price Improving Order) **f** Floor Order **G** SetterNoSize **h** halt queued **H** hidden **I** hidden improved **J** joiner **k** BrokerPreferencing **K** hidden reserve (hidden portion of a reserve order) **m** hidden midpoint (US Equities: Hidden midpoint execution) **M** MiddayCross **n** CLNK **N** normal **O** open auction **o** open queued **P** IPO auction **p** Periodic Auction (applicable for Cboe-BYX only) **q** QCC (Options only - Qualified Contingent Cross) **R** bolt route **r** Persisted (GTC restatement) **s** SAM Auction **S** setter **T** dark Book IOC **u** ClosingCross **U** Turner **v** ClosingCrossBrokerPref **V** visible improved **x** Multilateral Compression Trade of Proprietary Product **y** Related Futures Cross (RFC) **z** Position Compression Cross (PCC) **TradeRptTime** Shows the Tape Report Time (when the system reports to OPRA i.e. when the GUI user hits the send button) applies to manual and block trades only. Requires a timestamp. (e.g., TradeRptTime=20170108T023000.123456789). Note that the timestamp must be in the CAT timestamp format described in section 1.5 of the tech specs **Allowed Values: BOX** **FLEX** Event is associated to a FLEX Option **TT** Indicates when the trade was done. Requires a choice value from the following list: **Opening** **MarketOperation** **ContinuousTrading**

Field Name	Data Type	Description	
executionCodes ***(continued)***			**GuaranteedAuction**
			SolicitationAuction
			FacilitationAuction
			ExecutedAway
			FloorTrade
		STI	Indicates the trade type. Requires a choice value from the following list:
			RegularTrade
			As-of-Trade
			Block Trade
			Late Trade
			Hidden Trade
			Price Volume Adjustment
			Exchange For Risk
			Basis Swap
			IsoInbound
			GdoTradeThrough
			PipSweep
			USContingent
			Pip
			Crossed
			FloorTrade
		SID	Indicate the Strategy id. Value associated will be blank or will contain the Strategy Identification in the format of Text(10).
		SOT	Indicates the strategy order type. Requires a choice value from the following list:
			IMPROVE **INITO** **MBF** **EXPOSED** **CROSS** **CONTINGENT**
		STID	Indicate the Strategy Trade Id. Value associated will be blank or will contain the Strategy Identification
		STT	Indicates the strategy trade type. Requires a choice value from the following list:
			Opening **MarketOperation** **ContinuousTrading** **GuaranteedAuction** **SolicitationAuction** **FacilitationAuction** **ExecutedAway** **FloorTrade**
		SV	Indicate the Strategy Verb. Value associated will be blank or will contain B (for Buy), S (for Sell). Note: allowed values included “Sell” or “Buy” as part of back processing only for trade dates 3/29/19 to 6/21/19. They were active between the processing dates of 7/26/19 to 8/30/19.
		Allowed Values: MIAX	
		AUC	Indicates an auction. Requires one of the values from the

Field Name	Data Type	Description
executionCodes ***(continued)***		following list: **1** Opening **2** Reopening **3** Closing **4** Routing **5** LiquidityRefresh **6** PairedPrime **7** CustomerCrossPrime **8** QualifiedContingentCrossPrime **9** LiquidityExposure **C** ImmediateUncrossing **I** IIPOpening **L** CLEP **P** RIPReEvalutionCross **R** RIPReEvalution **U** URIPAuctionOnArrival **Y** IIPOpeningCross **Allowed Values: MIAX Emerald** **AUC** Indicates an auction. Requires one of the values from the following list: **1** Opening **2** Reopening **3** Closing **6** PairedPrime **7** CustomerCrossPrime **8** QualifiedContingentCrossPrime **C** ImmediateUncrossing **I** IIPOpening **L** CLEP **P** RIPReEvalutionCross **R** RIPReEvalution **U** URIPAuctionOnArrival **Y** IIPOpeningCross **Allowed Values: CHX** **TradeType** Name value pair, which requires value to be one of the following choices from the following list: **CSP** CSS entered correspondent trades **AWA** Away Market Executions **CHX** ECHX Trade **MAN** Manual **DRP** Drop copy away market execution **NAM** Recovery required **RCV** Recovery of NAME/NAME trade **AWE** Away sent electronically thru CHX systems **AWM** Away sent manually thru CHX systems

Field Name	Data Type	Description
executionCodes ***(continued)***		**RPT** Allocation report **AWF** Away market trades cleared by CHX **VEN** Away market clearing flip - non-ORS **AAW** IB Alternative Away Market Execution **AOR** ORS Away market clearing flip **RPS** Riskless Principal Second Component Trade **SNAP** Sub-second Non-displayed Auction Process (SNAP) Trade **executionID** For OrderFill, this is the execution ID received from the routing vendor. The value is of type Text<40> **executionMarket** For OrderFill - requires a choice value from the following list: **XCHI** Chicago Stock Exchange **XNYS** New York Stock Exchange **XASE** American Stock Exchange **ARCX** NYSE ARCA **XBOS** Boston Stock Exchange **XPHL** Philadelphia Stock Exchange **XCIS** National Stock Exchange **XADF** FINRA ADF **XCBO** Chicago Board Options Exchange **XNAS** NASDAQ Stock Exchange **BATS** Cboe BZXStock Exchange **BATY** Cboe BYZ - Exchange, Inc. **EDGA** Cboe EDGA **EDGX** Cboe EDGX **IEXG** Investors Exchange **Allowed Values: NYSE Options** **Cabinet** **COA** Auction Type **Complex** **CUBE Auction** Part of back processing only for trade dates 3/29/19 to 6/21/19. This value was accepted between the processing dates of 7/26/19 to 8/30/19. . **CUBEAuction** **Flex** **Man** **Open Auction** Part of back processing only for trade dates 3/29/19 to 6/21/19. This value was accepted between the processing dates of 7/26/19 to 8/30/19. **OpenAuction** **Allowed Values: NYSE Equities** **CROSS** **Allowed Values: IEX** **A** Member adds liquidity against a Retail Liquidity Provider order **C** Closing Auction on IEX

Field Name	Data Type	Description
executionCodes ***(continued)***		**B** Tape B Security **D** Execution of displayed Continuous Book interest in a cross or auction **H** Halt Auction Opening on IEX **I** Continuous Trade on IEX **K** Discretionary Peg, Fixed Midpoint Peg, Midpoint Peg, or Primary Peg order removes displayed liquidity (Tape B) **L** Traded with Displayed Liquidity **M** Added **O** Opening Auction on IEX **P** IPO Auction Opening on IEX **Q** Removes liquidity during a crumbling quote **R** Retail order removes liquidity **S** Self Trade on IEX **T** Removed **X** Opening Match on IEX **Y** Post only order executes on entry **W** Resting order removes against Post Only order **Allowed Values: NASDAQ ISE, GEMX, MRX, NOBO** **liquidityCode** Name value pair, requires one of the following values from the following list: **0** None **1** Maker **2** Taker **4** Response **5** Hidden **6** OpeningRotation **7** Cross **8** FlashedOrder **9** FlashResponse **10** RoutedOut **11** TradeReport **12** ComboMakerAgainstCombo **13** ComboTakerAgainstCombo **14** ComboResponseAgainstCombo **15** ComboHiddenAgainstCombo **16** ComboOpeningRotation **17** ComboCross **18** ComboTakerAgainstRegular **19** RegularMakerAgainstCombo **20** ComboTakerAgainstIO **21** RegularTakerAgainstIO **22** IOMakerAgainstCombo **23** IOMakerAgainstRegular **24** RegularMakerAgainstIOParticipant **25** IOParticipantTakerAgainstRegular

Field Name	Data Type	Description		
executionCodes *(continued)*			**26**	BrokenPriceImprovement
			27	BrokenFacilitation
			28	BrokenSolicitation
			29	ComboBrokenPriceImprovement
			30	ComboBrokenFacilitation
			31	ComboBrokenSolicitation
			32	Block
			33	BlockResponse
			34	DirectedResponse
			35	Facilitation
			36	FacilitationResponse
			37	PriceImprovement
			38	PriceimprovementResponse
			39	Solicitation
			40	SolicitationResponse
			41	QualifiedContingentCross
			42	CustomerToCustomer
			43	ComboFacilitation
			44	ComboFacilitationResponse
			45	ComboPriceImprovement
			46	ComboPriceImprovementResponse
			47	ComboSolicitation
			48	ComboSolicitationResponse
			49	ComboQualifiedContingentCross
			50	ComboCustomerToCustomer
			51	SweepRoutedOut
			52	SweepTradeReport
			53	ComboTakerAgainstRegularThruNbbo
			54	ComboTakerAgainstIOThruNbbo
			55	SimpleExposureOrderInitiatorUponReceipt
			56	SimpleExposureOrderInitiator
			57	SimpleExposureOrderResponder
			OTHER	Other
		BuyMatchId	Unsigned value	
		SellMatchId	Unsigned value	
		AuctionId	Unsigned value	
		TradeSource	Name value pair, requires one of the following values from the following list:	
			0	AUTO_EXECUTION
			1	OPENING
			2	FLASH
			3	EXPOSURE
			4	BLOCK
			5	PIM
			6	PIM_COMBO
			7	FAC
			8	FAC_COMBO

Field Name	Data Type	Description
executionCodes ***(continued)***		**9** SOL **10** SOL_COMBO **11** CCC **12** CCC_COMBO **13** QCC **14** QCC_COMBO **15** MANUAL **16** NOS **17** OPENING_UNCROSS **18** UNCROSS **OTHER** OTHER **Allowed Values: NASDAQ – PHLX, NOM, NOBO** **TradeSource** Name value pair, requires one of the following values from the following list: **1** AUTOEX **2** DET **3** EBOOK **4** NOS **5** FBMS **6** SWEEP **7** QUOTE_M **8** CO_SWEEP **9** LEGGING **10** COMPLEX **11** OPENING **12** COLA **13** COCRA **14** PIXL_AUTO **15** PIXL_STOP **16** QCC **17** QCC_FBMS **FLEX** FLEX **OTHER** OTHER **BuyMatchId** Unsigned value **SellMatchId** Unsigned value **AuctionId** Unsigned value **Allowed Values: Nasdaq – NOBO** *in addition to values defined above* **TradeSource** Name value pair, requires one of the following values from the following list: **20** AUTO_EXECUTION **21** OPENING **23** EXPOSURE **24** BLOCK **25** PIM

Field Name	Data Type	Description
executionCodes *(continued)*		**26** PIM_COMBO **27** FAC **28** FAC_COMBO **29** SOL **30** SOL_COMBO **31** **CCC** **32** CCC_COMBO **33** QCC **34** QCC_COMBO **35** MANUAL **36** NOS **37** OPENING_UNCROSS 38 UNCROSS **Allowed Values: LTSE** **L** Continuous Trade on LTSE **S** Self Trade on LTSE **O** Opening Auction on LTSE **C** Closing Auction on LTSE **H** Halt Auction Opening on LTSE **N** IPO Auction Opening on LTSE
executionDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date the execution occurred.
executionPrice	Price	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Unit price of the trade.
executionQuantity	Unsigned	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Number of shares traded.
executionTimestamp	Timestamp	*Event(s): Order Trade Correction, Option Trade Correction* When a trade is reported, the time of the trade is reported as the eventTimestamp. The executionTimestamp is used in a correction event if the time of the trade needs to be changed. *Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the execution occurred.
exerciseStyle	American	*Reference Data: Option Series Dictionary Entry (OSDE)* Specifies the exercise style of the Option Series in Simple Option Series Dictionary Entry. **Allowed Values** **American** **European**
expirationDate	Date	*Reference Data: Option Series Dictionary Entry (OSDE)* The date an options contract will expire, taking the format: YYYYMMDD.

Field Name	Data Type	Description
explicitFeeFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates if a Clearing Price was entered. **Allowed Values** **Y** Explicit Fee Trade
fillID	Text (40)	*Event(s): Supplemental Trade Event (STE), Order Fill Event, Stock Leg Fill Event* A unique identifier for the transaction. The combination of reporter, date, symbol, side, and fillID should be unique.
finraContraControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Number used for interaction between TRFs and FINRA; populated only when trade is matched by comparison. Will be unique for a trade report date and market center.
finraControlDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Date of the current version of the trade.
finraControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Number of the current version of the trade.
finraTradeModifierLateCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* System Trade Modifier - Time Modifier - Updated by MPP Engine. **Allowed Values** **T** Executed Outside Normal Market Hours **U** Executed Outside Normal Market Hours and Reported Late **Z** Executed During Normal Market Hours and Reported Late
finraTradeModifierSroCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* System Trade Modifier SRO - Updated by MPP System. **Allowed Values** **B** Weighted Average Price for Trade Disseminated to CTA SIP **I** Odd Lot Trade **P** Prior Reference Price **V** Contingent Trade **W** Weighted Average Price for Trade Disseminated to UTP SIP **X** Exercise of OTC Option
finraTradeModifierThroughExemptTime	Time	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* System Trade Thru Exempt Modifier Time.
firmOriginalTrfControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Original Control Number provided by the TRF to the firm.
firmTradeModifierLateCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* System Trade Modifier - Time Modifiers (TradeModifier 3 in the FIX Spec). **Allowed Values** **T** Executed Outside Normal Market Hours **U** Executed Outside Normal Market Hours and Reported Late **Z** Executed During Normal Market Hours and Reported Late

Field Name	Data Type	Description
firmTradeModifierSettlementTypeCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* User Trade Modifier - Settlement Type (Settlement modifiers). **Allowed Values** **@** Regular settlement **C** Cash settlement **N** Next day settlement **R** Seller settlement
firmTradeModifierSroCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Further classification of the trade with regard to SRO required detail. This can either be entered by the firm or appended by the system. **Allowed Values** **1** Stop stock (regular trade) **A** Acquisition **B** Bunched Trade **D** Distribution **E** Automatic execution (system) **H** Intraday trade detail (system) **I** Odd lot **K** Rule 155 Amex/Rule 127 NYSE **M** Market Center close price (system) **O** Odd lot **P** Prior reference price **Q** Market center open price (system) **R** Away from market sale **S** Split trade **V** Contingent trade **W** Average price trade **X** Exercise of OTC option

Field Name	Data Type	Description
firmTradeModifierThroughExemptCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Further classification of the trade with regard to Trade Through Exemption. This is entered by the firm when it reports the trade. **Allowed Values** **2** NASD Self Help Indicator **3** Intermarket Sweep - Outbound **4** Derivatively Price **5** Market Center Reopen **6** Market Center Closing **7** Error Correction **8** Print Protection **9** Correct Consolidated Close Price as per Listing Market **F** Intermarket Sweep **J** NASD Subpenny Indicator **O** Market Center Open **V** NASD Contingent Indicator
firstTradeFinraContraControlDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Date of the first trade in a chain of corrections on the contra side trade report.
firstTradeFinraContraControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Number of the first trade in a chain of corrections on the contra side trade report.
firstTradeFinraControlDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Date of the first version of the trade.
firstTradeFinraControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Number of the first version of the trade.
floorBroker	Member Alias	*Event(s): Option Trade Event* The Member Alias of the executing floor broker.
groupID	Text (40)	*Reference Data: Complex Option Dictionary Entry (CODE)* An identifier supplied by the user/reporter.
haltActionCode	Choice	*Event(s): FINRA Halt/Resume (FHR)* Indicates the action being taken. **Allowed Values** **H** Trading Halt **Q** Quotation Resumption **T** Trading Resumption **X** Quotation and Trading Resumption
haltActionTimestamp	Timestamp	*Event(s): FINRA Halt/Resume (FHR)* The date/time the halt was initiated.

Field Name	Data Type	Description
haltMessageType	Choice	*Event(s): FINRA Halt/Resume (FHR)* Identifies the message format, in conjunction with the `messageCategory`. **Allowed Values** **2** Closing Trade Summary **A** General Administrative Message **H** Trading Action (Security) **M** Trading Action (Extraordinary Market)
haltReasonCode	Choice	*Event(s): FINRA Halt/Resume (FHR)* Indicates the reason for the halt/resume. **Allowed Values** **C11** Trade Halt Concluded by Other Regulatory Authority; Quotes/Trades to Resume **C13** Quote Only Resume for EMC and MWCB Quote **C14** Quote and Trade Resume for EMC and MWCB **CXL** Cancel **D1** Security Deleted from OTCE **H10** Halt - SEC Trading Suspension **H12** Halt - SEC Revocation **O1** Halt - Component/Derivative of Exchange-Listed Security **T3** Halt – News and Resumption Times **U1** Halt – Foreign Market/Regulatory **U2** Halt – Component/Derivative of Exchange-Listed Security **U3** Halt – Extraordinary Events **U4** Extraordinary Market Condition (EMC)Halt **U5** Market-wide Circuit Breaker Halt
handlingInstructions	Name / Value Pairs	*Event(s): Order Accepted Event, Order Route Event, Order Modified Event, Order Modify Route Event, Order Restatement Event, Simple Option Order Accepted Event, Complex Option Order Accepted Event, Complex Option Order Modified Event, Stock Leg Order Event, Option Order Modified Event, Stock Leg Modified Event, Option Route Event, Complex Option Route (OCOR), Modify Option Route Event, Options Order Restatement Event* The order handling instructions field is a way to provide multiple instruction codes in a somewhat flexible manner. This field will contain zero or more order instruction codes, each separated by a single pipe symbol (ASCII decimal 124, hex 7C). Codes which require a value will include that value immediately after the code Field Name and a single equal sign (ASCII decimal 61, hex 3D). All instructions that apply to the order are to be included. **Allowed Values: Boolean** *presence indicates truth* **AON** All or None **AUC** Auction Eligible **DNR** Do Not Route **FOK** Fill or Kill **IOC** Immediate or Cancel

Field Name	Data Type	Description	
handlingInstructions *(continued)*		**ISB**	Intermarket Sweep Book
		ISO	Intermarket Sweep
		NH	Not Held
		OPG	At the Opening
		PSO	Post Only
		WTP	Wash Trade Prevention
		Note: Some exchanges have special values to indicate handling of ISO orders. All ISO orders must be marked with the boolean ISO value. Thus, if an exchange denotes an ISO order with some custom attribute, it must also be marked with the common ISO value.	
		Allowed Values: Name Value Pairs	
		MIN	Minimum Quantity - requires an Integer value, representing he minimum quantity allowed to be executed in a single transaction (e.g., MIN=1000).
		WD	With Discretion Price - requires a Numeric value, representing the discretion price (e.g, WD=12.50)
		STP	Stop Price - requires a Numeric value representing the stop price (e.g., STP=17.95)
		XDATE	Expire Date - requires a Date value, representing the date that the order expires. The value must be in Date format (e.g., May 15, 2017 would be XDATE=20170515). The order expires at the close of the specified date.
		XTIME	Expire Time - requires a Time value, representing the time that the order expires. The value must in a valid Timestamp format.
		R2E	Route to Exchange - requires Exchange ID (e.g., R2E=G). The desired route destination is not the party receiving the actual route. The party receiving the route does not have discretion as to where to route the order. It must be routed to a specific exchange.
		R2M	Route to Industry Member - requires Member Alias (e.g., R2E=ABC123). The desired route destination is not the party receiving the actual route. The party receiving the route does not have discretion as to where to route the order. It must be routed to a specific industry member.
		R2O	Route to Other - requires Text(20) (e.g., R2O=Somebody). The desired route destination is not the party receiving the actual route. The party receiving the route does not have discretion as to where to route the order. It must be routed to an entity who is neither an exchange nor an industry member (i.e., the entity does not have a CAT reporting responsibility).
		Allowed Values: Cboe Legacy (C1) *active 3/29/2019 – 10/4/2019*	
		MIT	Market if touched, becomes a market order if the price is touched. Requires a price value (e.g, MIT=20.53).
		AucResp	A response to an auction, the remainder is canceled at the end of the auction. Requires a integer value representing the auction ID being responded to. (e.g., AucResp=1234).
		Reserve	Reserve, only a portion of the order is displayed. Requires an integer value representing quantity. (e.g., Reserve=300).
		PMM	Preferred market maker, requires a text (text, 10) value representing the acronym of the preferred market maker. (e.g.,

Field Name	Data Type	Description
handlingInstructions ***(continued)***		PMM=FRMA) **AIM** Automated Improvement Mechanism. Requires a choice value (e.g., AIM=AIM) selected from the following list **AIM** standard AIM **AIQ** QCC Primary Order **AIS** Sweep and AIM primary order **AIR** Re-route if cannot AIM primary order **ARE** Contra order to AIM. Requires a text (text 20) value representing the primary order ID. (e.g., ARE=AB54321) **AREOUT** Contra order to AIM where the user can opt out. Requires a text (text 20) value representing the primary order ID. (e.g., ARE=AB54321) **Designation** Order designation, requires a choice value (e.g., Designation=4) from the following list: **1** Tied Hedge **2** SPXCOMBO **3** Tied Hedge and Cash Spread **4** SPXCOMBO and Cash Spread **5** Cash Spread **UHI** User handling instruction, requires a choice value (e.g., UHI=4) from the following list: **1** Do Not Auction **2** Held **3** Solicited Order **4** Held and Solicited **5** Held and no COA **6** Electronic Only **7** Electronic Only and Solicited **8** Electronic Only and no COA **Allowed Values: Cboe** **ExecInst** Provides additional values for execution instructions that aren't already present in orderType or other handlingInstructions values. Requires a choice value (e.g., ExecInst=U) from the following list: **N** No special instructions **s** sweep **M** hidden peg to midpoint **L** alternative midpoint peg to less aggressive midpoint or 1 tick inside of NBBO **m** midpoint peg no lock hidden peg to midpoint but duck at or beyond limit **d** displayed peg order with discretion to the midpoint **g** AllOrNone **l** midpoint match (EDGX) **Q** market maker peg order **v** Dart dark route before outbound **w** DoNotDart opt of Dart **x** ImproveOnly Cboe only IOC that only matches better

Field Name	Data Type	Description	
handlingInstructions ***(continued)***			than NBBO
			y TAISO
			z DarkScan hit scan fast DLPs first
			t DarkScanWithoutDart
			r LateAuction late limit on open/close
			U route peg order
			u DartOnly route only to a dark venue
			F FastDart
			S SuperDart
			f ISO
			R PrimaryPeg
			h Minimum
			1 Not Held
			P MarketPeg
			X MidpointSwapOrder
			e Midpoint Discretionary Order with Quote Depletion Protection
		AutoMatchLimit	Auto Match any price improvement up to this price on a two-sided auction. Requires a PRICE datatype
		AutoMatchMkt	Auto Match any price improvement on a two-sided auction. Boolean – true if present
		LastPriority	The B/D does not want their full entitlement at the final auction price. Boolean – true if present
		RetailPriority	Retail orders are given priority. Boolean – true if present
		FloorTraderType	Type of Trader; Choice Field
			Unspecified
			PAROfficial
			PARBroker
			InCrowdMarketMaker
		AllowExposure	Expose auction order. Boolean – true if present
		WorkStationID	Work Station Identifier Name/Value Pair Alphanumeric(4)
		Reserve	Number of shares of a reserve order to display. Requires an UNSIGNED value
		ExtExecInst	Requires a choice value from the following list:
			N None
			T Retail Price Improving
			P Retail Order - Price Improvement Only
			R Retail Order
			S Retail Order NoFlagCLC
			X Retail Priority Order
			Y Retail Priority Order NoFlagCLC
		MaxRemovePct	The max percentage an order is allowed to remove before booking. Requires an Unsigned (e.g., MaxRemovePct=10)
		MaxRemovePercent	*The max percentage an order is allowed to remove before booking. Requires an Unsigned (e.g., MaxRemovePct=10). Part of back processing only for trade dates 3/29/2019 to 6/21/19. This value was accepted between the processing dates of 7/26/19 to 8/30/19.*
		AttributedOrder	Requires a choice value from the following list:

Field Name	Data Type	Description
handlingInstructions ***(continued)***		**N** None **Y** Attributed **R** Retail **C** AttributedClientIdOnly **Z** AttributedBoth **DisplayRange** This will be of type Unsigned, and is used for a "random replenishment" reserve order. The reload quantity is randomly selected using Reserve +/- displayRange e.g. Reserve of 1000, displayRange of 200, reload quantity will be randomly selected from 800, 900, 1000, 1100, or 1200. **Allowed Values: Cboe Equities** **TifMod** Supplemental time-in-force information. Requires a choice value (e.g., TifMod=1) from the following list: **1** include early and pre-market trading sessions **2** include pre-market session **3** include early, pre-, and post-market sessions **4** include pre-, and post-market sessions For session times, see the Hours of Operation section of the *Cboe US Equities Binary Order Entry ("BOE") Specification* and *Cboe US Equities FIX Specification* and documents available at https://www.cboe.com/us/equities/support/technical/. **Allowed Values: Cboe Options** **TifMod** Supplemental time-in-force information. Requires a choice value (e.g., TifMod=1) from the following list: **1** include pre-market session **5** GTH-Eligible (Options only) *3 Part of back processing only for trade dates 3/29/2019 to 6/21/19. This value was accepted between the processing dates of 7/26/19 to 8/30/19.* For session times, see the Hours of Operation section of the *Cboe Options Exchanges Binary Order Entry Specification ("BOE Specification")* and *US Options FIX Specification ("FIX Specification)* documents available at https://www.cboe.com/us/options/support/technical/. **Allowed Values: BOX** **EP** Requires Member Alias (e.g., EP=910). **IML** Indicate he Inter Market Linkage Behavior for the order. Requires a choice value from the following list: FLASH ROUTING NONE NBBO ISO CONTINGENT NOFLASH **PT** Indicate BOX Price Term for the order. Requires a choice value from the following list: PIP SOLICITATION

Field Name	Data Type	Description	
handlingInstructions ***(continued)***			FACILITATION CROSS DIRECTED PREF FLOOR
		OT	Indicate the order type for auction phase. Requires a choice value from the following list: IMPROVE INITO EXPOSED CROSS CONTINGENT MBF GTD Indicates Date in YYYYMMDD Format QT Requires a choice value from the following list: MINIMUM SURRENDER MIP AQ Indicate the additional quantity when QT is either MINIMUM or SURRENDER. Requires an unsigned integer value (e.g, AQ=1000) AP This will be field of type Price AT Requires a choice value from the following list: PIP SOLICITATION FACILITATION CROSS FIXED FLOOR
		AID	This will contain a "UNSIGNED" number that will allow BOX to track "Auction Phase Number" (e.g., AID=123456)
		Allowed Values: CHX (Legacy as of 11/2019)	
		ExecInst	Requires a choice value (e.g., ExecInst=f) from the following list: **5** Held **E** DNI - Do not increase **F** DNR - Do not reduce **K** Cancel on Trading Halt **X** TALG - Trade Along **y** Trade At Intermarket Sweep (TAISO) **q** Always Quote **I** Midpoint Cross **v** Stock-Option (for cross order only)
		TradeThruExemptReason	Requires a choice value (e.g., TradeThruExemptReason=2) from the following list: **1** Benchmark **2** QCT Qualified Contingent Trade

Field Name	Data Type	Description
handlingInstructions ***(continued)***		**3** Bonafide Error Indicator **PriceSliding** Requires a choice value (e.g., PriceSliding=L) from the following list: **L** CHX Only – Slide limit price on lock NBBO **S** CHX Only – Slide limit price on lock or cross NBBO **MatchTradePrevention** Requires a choice value (e.g., MatchTradePrevention=N) from the following list: **I** MTP Inactivate **N** MTP Cancel Newest **O** MTP Cancel Oldest **B** MTP Cancel Both **MTPSublevelInd** Requires a choice value (e.g., MTPSublevelInd=1) from the following list: [0-9,A-Z,a-z] **Allowed Values: LTSE** **PegO** Peg Offset, only on Primary Peg Orders. Requires a value for the offset +x.xx or -x.xx. (e.g. PegO=0.05, PegO=-0.05) **RML** Retail Midpoint Liquidity **RMO** Retail Midpoint Order **RP** Re-Price **RSV** Reserve **Allowed Values: NYSE Options** **ALO** Add Liquidity order **AON** All or None order **C2C** Customer to Customer Cross order **Cabinet** Cabinet order **ClearTheBook** Clear interest ahead of Outcry Trade **COA** Complex Order Auction order **ComplexOnly** Non-legging complex order **CUBEAUCF** (Legacy as of 10/23) **CUBEAUCPI** Price Improvement Auction **CUBEAUCS** Price Improvement Auction - All or None **Flex** FLEX order **FLEXPCT** FLEX Percentage order **FloorTrade** Outcry Trade **FloorTradeNamesLater** (Legacy as of 10/23) **FloorTradeNamesLaterAllocation** (Legacy as of 10/23) **IO** Imbalance Offset order **ISO** Intermarket Sweep Order **NOW** (Legacy as of 10/23) **NR** Non-Routable **ND** Non-Displayed (Legacy as of 10/23) **PNP** (Legacy as of 10/23) **PNP+** (Legacy as of 10/23) **PNPB** (Legacy as of 10/23) **PNPLO** (Legacy as of 10/23)

Field Name	Data Type	Description
handlingInstructions ***(continued)***		**QCC** Qualified Contingent Cross order **RoutableIOC** Routable IOC order **Stop** Stop Order (Stop=<px>) **StopLimit** Stop Limit Order (StopLimit=<px>) **Allowed Values: NYSE Equities** **355** Primary Market after 3:55 PM **945** Primary Market until 9:45 AM **945-355** Primary Market until 9:45 AND Primary Market after 3:55 **ALL** Trading session eligibility - ALL sessions applicable on the corresponding market **ALO** Add Liquidity order **AOC** Auction or Cancel **BrokerConfirm** Confirmation to trade a floor broker cross **CCO** Capital Commitment Order **CCO-PartialFill** Capital Commitment Order - Partial Fill Contra Side **ClosOffset** Closing Offset Order **CORE** Trading session eligibility - Core session eligible only **CORE_LATE** Trading session eligibility - Core & Late session eligible only **DIR** Routed to Primary market **DirectedTo_ALGO** Identifies if the inbound order has instructions to route the order to the specified Algo provider **DirectedTo_ATS** Routed to OneChronos **DLP** Limit DOrder **DMP** Midpoint DOrder **DPO** DOrder on Open or Close **DPP** Dark Primary Peg **IDO** Issuer Direct offering **ImblOffset** Imbalance Offset order **ISO** Intermarket Sweep Order **LPEG** Last Sale Peg order **MPEG** Market Peg order **MPL** Midpoint Passive Liquidity Order **NoIOI** No Route to IOI **NoMPL** (Legacy as of 8/2019) **NoMPL-IOI** (Legacy as of 8/2019) **Non-Display** Non Displayed Order **Non-Routable** Non Routable Order - DAY/GTC **NonRoutableIOC** Non Routable IOC Order **PO** Directed to Primary **POST** Trading session eligibility - Late session eligible only **PPEG** Primary Peg order **PRE** Trading session eligibility - Early session eligible only **PRE_CORE** Trading session eligibility - Early & Core session eligible only **QCT** Qualified Contingent Trade **Retail** Retail Order **RoutableIOC** Routable IOC order

Field Name	Data Type	Description	
handlingInstructions *(continued)*		**RPI**	Retail Provider
		Tracking	Tracking
		TradeAtISO	Trade-at ISO order
		Allowed Values: NOBO, PHLX, NOM, ISE, GEMX, MRX	
		Boolean Values	
		PostOnly	
		PostOnlyPrice	
		WAIT	
		AllowFlash	
		AllowExposure	
		DNR	
		DNTT	Do not trade through
		DNA	Do not Auction
		AO	Auction Only
		Name Value Pairs	
		DMM	STRING; DMM Name
		PMM	*STRING; PMM Name – Part of back processing only for trade dates 3/29/19 to 6/21/19. This value was accepted between the processing dates of 7/26/19 to 8/30/19.*
		DisplayWhen	For reserve orders, requires one of the following **1** Immediate **2** onExhaust
		RefreshMax	UNSIGNED; Contracts
		RefreshMin	UNSIGNED; Contracts
		InitDispContracts	UNSIGNED; Contracts [Initial Display Contracts for reserve orders]
		Reserve	*UNSIGNED; Contracts [Initial Display Contracts for reserve orders] – Part of back processing only for trade dates 3/29/19 to 6/21/19. This value was accepted between the processing dates of 7/26/19 to 8/30/19.*
		RoutingStrategy	Must be one of the following SRCH FIND SEEK
		RespAuctionId	UNSIGNED; auctionId
		MIN	UNSIGNED; Contracts
		OrderSource	Must be one of the following FIX OTTO SQF FBMS_FIX FBMS PRECISE_FIX QUO
		BrokerPct	NUMERIC<3,4>; Percentage
		EffectiveTime	TIME

Field Name	Data Type	Description
handlingInstructions *(continued)*		**StepUpPrice** PRICE **StepUpPriceType** Must be one of the following **1** Market **2** Limit **DMA** DMA Name [for route event], where 'DMA Name' can have values from the following list: CITI WEX MLGW GSG *GSW Part of back processing only for trade dates 3/29/19 to 6/21/19. This value was accepted between the processing dates of 7/26/19 to 8/30/19.* OTHER **DestExch** Dest Exch [for route event], where 'DestExch' can have values from the following list: **11** AMEX **12** BOXE **13** CBOE **14** EDGO **15** GMNI **16** ISEX **17** MCRY **18** MIAX **19** NYSE **20** MPRL **21** NSDQ **22** NOBO **23** CBC2 **24** PHLX **25** BATS **26** EMLD **27** MEMXOP **28** SPHR **1** BNY**2** CHBC **3** LBKI 4 FOGS OTHER OTHER **Allowed Values: ISE, GEMX, and MRX** *in addition to values defined above* **CrossType** Value must be one of the values from the following list: **1** None **2** Close **3** Open **4** PriceImp **5** QCC **6** Solicit

Field Name	Data Type	Description
handlingInstructions ***(continued)***		**7** Facilit **8** Flash **9** Block **10** Exposure **11** Cust OTHER **Allowed Values: PHLX, NOM, NOBO** *in addition to values defined above* **CrossType** Value must be one of the values from the following list:: **1** None **2** Close **3** Open **4** Complex **5** Open Complex **6** Close Complex **7** PIXL **8** QCC **9** SOLICIT **10** Complex PIXL **11** Complex SOLICIT OTHER **Allowed Values: Nasdaq – NOBO, MRX, GEMX** *in addition to values defined above* **CrossType** Value must be one of the values from the following list: **31** None **32** Close **33** Open **34** PriceImp **35** QCC **36** Solicit **37** Facilit **39** Block **40** Exposure **41** Cust **42** Volatility **RfaId** Value must be an integer. (e.g. RfaId=1234) **RfaInstruction** Value must be one of the values from the following list (e.g. RfaInstruction=B) **B** Booked **C** Cancel **Allowed Values: BX, PSX, NSDQ** **ChildCancelReason** Value must be one of the values from the following list: **1** User Requested Cancel **2** Immediate or Cancel order.

Field Name	Data Type	Description	
handlingInstructions ***(continued)***			**3** Timeout. The Time In Force for this order has expired **4** Supervisory. **5** This order cannot be executed because of a regulatory restriction **6** Self-Match Prevention. **7** System cancel. **8** Cross-canceled. Non-bookable cross orders that did not execute in the cross. **9** Order canceled due to insufficient quantity **10** This order cannot be executed because of Market Collars **11** Halted. The on-open order was canceled because the symbol remained halted after the opening cross-completed. **13** Closed. Any DAY order that was received after the closing cross is complete in a given symbol will receive this cancel reason. **15** Administrative cancel **16** Post Only Cancel. This Post Only order was canceled because it would have been price slid for NMS. **17** Post Only Cancel. This Post Only order was canceled because it would have been price slid due to a contra side displayed order on the book **18** Direct Listing with Capital Raise amt exceeded **19** Open Protection **20** Discretion **21** Cross Supervisory **22** Managed Orders **23** Cancel On Disconnect **24** User Modified **25** Oddlot **26** User Replaced **27** User Split **28** System Downtick **29** Market Collar **30** Late Cancel **31** Quantity Cancel **32** Forced Cancel **33** Reject Cancel ADMIN for an administrative cancel FEATURE in the service of a customer-requested feature OTHER
		Display	Value must be one of the values from the following list: **1** Attributable-Price to Display **2** Anonymous-Price to Comply **3** Non-Display **4** Post-Only **5** Imbalance-Only (for opening and closing cross only) **6** Mid-Point

Field Name	Data Type	Description	
handlingInstructions ***(continued)***			**7** Mid-Point Post Only
			8 Post-Only and Attributable – Price to Display
			9 Retail Order Type 1
			10 Retail Order Type 2
			11 Retail Price Improvement Order
			12 RoundLotOnly
			13 Latent
			14 HiddenFromReserve
			15 Conformant
			OTHER Other
		DLCR	Direct Listing with Capital Raise
		DMA	DMA Name [for route event], where 'DMA Name' can have following values:
			GSET
			MSCO
			OTHER
		EMOC	Extended Market On Close
		ExecBroker	Value must be one of the values from the following list:
			BCRT
			BCST
			BDRK
			BMOP
			BSCN
			BSKN
			BSKP
			BSTG
			BTFY
			DOTA
			DOTD
			DOTM
			DOTI
			MOPP
			TFTY
			SCAN
			SKIP
			SKNY
			SAVE
			QSAV
			QTFY
			DOTZ
			LIST
			CART
			SOLV
			QSLV
			ESCN
			MOPB
			RFTY

Field Name	Data Type	Description	
handlingInstructions ***(continued)***			QRTY
			INET
			ISAM
			ISBX
			ISBY
			ISBZ
			ISCX
			ISIX
			ISNA
			ISNX
			ISNY
			ISPA
			ISPX
			ISCN
			ISLT
			PCRT
			PMOP
			PSCN
			PSKN
			PSKP
			PSTG
			PTFY
			QCST
			QDRK
			STGY
			TFYB
			TFYX
			XCST
			XDRK
			ALL
			BNET
			ISCB
			ISLF
			ISMI
			ISMX
			ISNQ
			MIDP
			QNET
			SCAR
			XNET
			LSTY
			CUSTOM-RFTY
			OTHER
		ExecInst	Value must be one of the values from the following list:
			1 Midpoint Peg
			2 No Peg
			3 Market Peg

Field Name	Data Type	Description
handlingInstructions ***(continued)***		**4** Quoting Peg **5** Primary Peg **6** INAV pegging **7** means Intermarket Sweep Order (ISO) **8** means Trade-at Intermarket Sweep Order **9** means Reactive Trade Now **10** means Reactive Trade Now opt-out **MELO** for a Midpoint ELO order **RPI** for a Retail Price Improvement Program order **SUPL** for a Supplemental order **RSRV** = <Maxfloor> **XCTBL** Value must be one of the values from the following list: Y N **Allowed Values: MEMX** **PegO** Peg Offset, only on Primary Peg Orders. Requires a value for the offset +x.xx or -x.xx. (e.g. PegO=0.05, PegO=-0.05) **RML** Retail Midpoint Liquidity **RMO** Retail Midpoint Order **RP** Re-Price **RSV** Reserve **Allowed Values: MEMXOP** **PA** Price Adjustment **Allowed Values: MIAX PEARL Equities** **RouteOnce** Order will route upon arrival if marketable against away quotes and then, depending on time-in-force, will rest on the MIAX PEARL book. **ReRoutable** Order will route upon arrival if marketable against away quotes and then rest on the MIAX PEARL book. If an away market subsequently locks or crosses the order, the order will route again. **RoutingStrategy** Routing strategy. Requires a choice value from the following list (e.g., RoutingStrategy=2): **2** OrderProtection **3** PrimaryAuction **Allowed Values: MIAMI, EMLD** **AutoMatchLimit** Auto Match any price improvement up to this price on a two-sided auction. Requires a PRICE datatype. **AutoMatchMarket** Auto Match any price improvement on a two-sided auction. Boolean – true if present. **Allowed Values: IEX** **CxlPxBack** Provides instructions for behavior to enforce upon expiration of a set timer for D-Limit Orders. Requires a choice value (e.g., CxlPxBack=N) from the following list:

Field Name	Data Type	Description
handlingInstructions ***(continued)***		**N** No Action **C** Cancel Order **U** Re-price order to the inside NBB (Buys) and NBO (Sells) **DisplayRange** Specifies a quantity range for random replenishment of reserve orders. (e.g. DisplayRange=100) **Reserve** Quantity to display for reserve orders (Max Floor). (e.g. Reserve=500) **RoutingInst** Requires a choice value (e.g. RountingInst=1) from the following list: **0** None **1** Trade Now **2** Trade Now opt-out **SigVersion** Requires a choice value from the following list that specifies which version of the Signal applies to the order (e.g. SigVersion= SignalV5): **SignalV5** **SignalV6**
ID	Text (20)	*Reference Data: Member Dictionary Entry (MDE), Member Alias Detail Entry (MADE)* The CRD number of the firm.
initiator	Choice	*Event(s): Order Modified Event, Order Canceled Event,Option Quote Event, Option Quote Cancel Event, Option Order Modified Event, Complex Option Order Modified Event, Stock Leg Modified Event, Option Order Canceled Event* Indicates who initiated a cancel or modification request. If an order/quote is implicitly modified or canceled via an unsolicited action (e.g., peg order price change or cancelation due to timeout), then the initiator is the exchange itself. If an order/quote is modified or canceled as a result of an explicit request from the party that sent the order/quote, then the initiator is the firm/market maker that sent the explicit modify/cancel request. Thus, all explicit modify/cancel requests will have an initiator of either Firm or MarketMaker, as appropriate and all implicit, unsolicited modify/cancel actions will have an initiator of Exchange. **Allowed Values** **Firm** **Exchange** **MarketMaker**
intendedMarketCenter	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Intended Market Center. **Allowed Values** **D** ADF
isGloballyUnique	Boolean	*Event(s): Complex Option Accepted (OCOA), Complex Option Route (OCOR), Complex Option Internal Route (OCIR)* Indicates that the orderID is globally unique across all optionIDs for the exchange/date. This means that no other complex order can have the same orderID. Furthermore, leg events for this complex order must be reported with just the complexOrderID and not the complexOptionID.

Field Name	Data Type	Description
issueID	Integer	*Event(s): FINRA Halt/Resume (FHR)* Indicates the issue being halted/resumed.
issueSymbolId	Symbol	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Character symbol of the traded issue.
kind	Choice	*Reference Data: Option Series Dictionary Entry (OSDE), Complex Option Dictionary Entry (CODE)* Specifies if an option is a simple, complex, flex, or percentage denominated flex option. For the value FLEXPCT, the strike price and order prices of the option are in percentages. **Allowed Values** **Complex** **Standard** **Non-Standard** **FLEX** **FLEXPCT**
lastUpdateDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date the record was last updated.
lastUpdateTime	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the record was last updated.
leavesQty	Unsigned	*Event(s): Order Canceled Event, Order Trade Event, Order Fill Event, Order Cancel Route Event, Order Restatement Event, Option Order Canceled Event, Option Cancel Route Event, Option Trade Event, Stock Leg Fill Event, Options Order Restatement Event* The quantity remaining unfilled after the event. The meaning of this field is subjective depending on the event, refer to each individual event definition for more detail.
legType	Choice	*Reference Data: Complex Option Dictionary Entry (CODE)* For a Complex Option Dictionary Entry, this field defines the type of each leg. **Allowed Values** **Equity** **Index** **Option**

Field Name	Data Type	Description
liquidityCode	Choice	*Event(s): Order Trade Event, Option Trade Event* Included in the side trade details for options and equity trade events, represents whether a given side was adding or removing liquidity. **Allowed Values** **Added** **Removed** **RoutedOut** **Opening-ReopeningAuction** **ClosingAuction** **CrossOrderExecution** **Other** **Allowed Values: MIAX PEARL Equities** **RoutingOrderProtection** Liquidity protection indicator for routed or away trades. **Allowed Values: NASDAQ – BX, PSX, NSDQ** **AfterHoursClose** **Midpoint-ELO**
lockedCrossOverrideFlag	Boolean	*Event(s): Equity Best Bid and Offer (EBBO)* Identifies whether a quote should be considered valid even if it will lock or cross the market. 'True' indicates that the quote is still valid 'False' indicates that the quote is not valid if it locks or crosses.
lockedInFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Locked-in flag. **Allowed Values** **A** Automatic Give Up (trade report on another firm's behalf) **Q** Qualified Special Representative (trade sent to clearing on another firm's behalf) **Y** Locked-in Trade
lockedInTradeTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the locked-in trade report was received by the reporting facility.

Field Name	Data Type	Description
marketCenterId	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Reporting facility to which the trade was reported. **Allowed Values** **C** Nasdaq TRF Chicago **D** ADF-TRF **N** NYX-TRF **L** Nasdaq-TRF **O** OTC-TRF (ORF) *Event(s): Equity Best Bid and Offer (EBBO)* Display-Only Facility on which the quote was displayed. **Allowed Values** **ADF** ADF Quote Display Facility
marketCenterOriginatorID	Choice	*Event(s): FINRA Halt/Resume (FHR)* Identifies the market center or system that originated the halt/resume action. **Allowed Values** **E** Market Center Independent (Message Generated by Data Feed Handler) **F** OTC Bulletin Board (OTCBB) and Other OTC Security **U** OTC Bulletin Board (OTCBB) **u** Other OTC Security (OOTC)
marketMaker	Member Alias	*Event(s): Quote Event (OQ), Quote Cancel Event* *Reference Data: Market Maker Dictionary Entry (MMDE)* The Member Alias assigned by the SRO as identified in the Member Dictionary Entry (MDE) memberAliases field. In the case where a market maker has multiple users (e.g., acronyms used to differentiate users within the same MM), there would be a separate Member Alias given to each user or sub-account.
marketMakerStatus	Choice	*Reference Data: Market Maker Dictionary Entry (MMDE)* The status of the member/symbol for the reporting date. **Allowed Values** **Active** Market Maker becomes active in the symbol **Inactive** Market Maker has become inactive in the symbol
marketMakerType	Choice	*Reference Data: Market Maker Dictionary Entry (MMDE)* A list of exchange defined values for the Equity Market Maker distinguishing between types or designations of market makers. Below are the common allowed values that are available to all exchanges. **Allowed Values** **MM** Market Maker *(default value)* **DMM** Designated Market Maker **LMM** Lead Market Maker **SLP** Secondary Liquidity Provider **SLMM** Secondary Liquidity Market Maker

Field Name	Data Type	Description
mediaReportedFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies if the trade was media reported or not (could differ from the publishIndicatorCode for odd lot trades). **Allowed Values** **Y** Media Reported **N** Not Media Reported
memberAlias	Member Alias	*Reference Data: Member Alias Detail Entry (MADE)* The member alias for which the MADE record is being reported.
memberAliases	Array of Member Alias	*Reference Data: Member Dictionary Entry (MDE)* A list of member aliases for an SRO member.
messageCategory	Choice	*Event(s): FINRA Halt/Resume (FHR)* Identifies the message format, in conjunction with the `haltMessageType`. **Allowed Values** **A** Administrative Messages **C** System Control Messages **T** Trade Related Messages
messageSequenceNumber	Integer	*Event(s): FINRA Halt/Resume (FHR)* At the beginning of each operational cycle, this number will be set to 00000000 (for the Start of Day) for each data channel.
messageTimestamp	Timestamp	*Event(s): FINRA Halt/Resume (FHR)* The date/time of the corresponding action (halt/resume).
mktMkrSubAccount	Text (20)	*Event(s): Simple Option Order Accepted Event, Option Order Modified Event, Option Trade Event, Option Order Restatement Event, Post Trade Allocation Event* The sub-account for the market maker. This is a text field and will be treated as pass through data - not validated.
name	Text	*Reference Data: Member Alias Detail Entry (MADE)* The doing-business-as (DBA) name corresponding to the member alias.
nbbPrice	Price	*Event(s): Order Accepted, Order Route, Order Modified, Order Trade, Order Modify Route, Simple Option Order Accepted, Stock Leg Order, Option Order Modified, Stock Leg Modified, Option Route, Modify Option Route, Simple Option Trade* The national best bid price at the moment the event. If the event changes the NBBO, this is the national best bid price before the change effected by the event, in this sense, this field is always the national best bid price immediately before the event occurs. See this field in context of the event definitions for more info.
nbbQty	Unsigned	*Event(s): Order Accepted, Order Route, Order Modified, Order Trade, Order Modify Route, Simple Option Order Accepted, Stock Leg Order, Option Order Modified, Stock Leg Modified, Option Route, Modify Option Route, Simple Option Trade* The national best bid quantity at the moment the event. If the event changes the NBBO, this is the national best bid quantity before the change effected by the event, in this sense, this field is always the national best bid quantity immediately before the event occurs. See this field in context of the event definitions for more info.

Field Name	Data Type	Description
nboPrice	Price	*Event(s): Order Accepted, Order Route, Order Modified, Order Trade, Order Modify Route, Simple Option Order Accepted, Stock Leg Order, Option Order Modified, Stock Leg Modified, Option Route, Modify Option Route, Simple Option Trade* The national best offer price at the moment the event. If the event changes the NBBO, this is the national best offer price before the change effected by the event, in this sense, this field is always the national best offer price immediately before the event occurs. See this field in context of the event definitions for more info.
nboQty	Unsigned	*Event(s): Order Accepted, Order Route, Order Modified, Order Trade, Order Modify Route, Simple Option Order Accepted, Stock Leg Order, Option Order Modified, Stock Leg Modified, Option Route, Modify Option Route, Simple Option Trade* The national best offer quantity at the moment the event. If the event changes the NBBO, this is the national best offer quantity before the change effected by the event, in this sense, this field is always the national best offer quantity immediately before the event occurs. See this field in context of the event definitions for more info.
noLinkControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Provides a link (via Control Number) to previous No transaction.
note	Text (255)	*Event(s): Note (NOTE)* Free form text provided by the exchange to describe the notation of the event.
noteType	Choice	*Event(s): Note (NOTE)* For a note event, classifies the type of note. **Allowed Values** **MISC** **Allowed Values: BOX** **StateChanged** **Allowed Values: Cboe Legacy (C1) Only** *active 3/29/2019 – 10/4/2019* **CBOE:1** Order Route Event (When an order is routed between internal CBOE systems). The source and destination will indicate more details. **CBOE:2** Cross Order Route Event **CBOE:3** Auction Start **CBOE:4** Auction End **CBOE:5** PAR_BROKER_USED_MKT_DATA **CBOE:6** PAR_BROKER_MKT_DATA **CBOE:7** PAR_BROKER_LEG_MKT **CBOE:8** PAR_MANUAL_MARKET_DATA **Allowed Values: Cboe Options** *active beginning on 10/7/2019* **CBOE:1** Order Route Event (When an order is routed between internal CBOE systems). The source and destination will indicate more details. **CBOE:6** PAR_BROKER_MKT_DATA **CBOE:7** PAR_BROKER_LEG_MKT **CBOE:8** PAR_MANUAL_MARKET_DATA

Field Name	Data Type	Description
noteType ***(continued)***		**Allowed Values: NYSE Options** **Floor** **Allowed Values: NYSE Equities** **CrossingSession** **AOCNoParticipation** Auction request acceptance to open/close the stock without participation by MM **AOCNoParticipationRei** Auction request to open/close the stock without participation by MM is rejected
noWasLinkNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Link to first No transaction
oeMemoTx	Text (10)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Memo text entered by firm.
onlyOneQuote	Boolean	*Event(s): Quote Event (OQ), Quote Cancel Event* True if the system allows only one quote for the particular market maker; false otherwise.
openCloseIndicator	Choice	*Event(s): Simple Option Order Accepted, Options Modified, Post Trade Allocation, Options Restatement or sideDetail of Option Trade events. (When this field is present in the sideDetails of an options trade event, it is applicable only when the side of the trade is an order)* Indicates the position of the order. **Allowed Values:** **Open** **Close** **Unspecified**
optionID	Text (40)	*Reference Data: Option Series Dictionary Entry (OSDE), Complex Option Dictionary Entry (CODE)* *Event(s): All events for Options Exchanges, Note (NOTE), Supplemental Trade Event (STE), Reject Message Event (RME)* The unique ID assigned to this option by the reporter. None of any two simple/complex/flex options should receive the same ID.
optionsSymbol	Text (14)	*Reference Data: Option Series Dictionary Entry (OSDE)* The option class or symbol for the series (as known by OCC).
orderAttributes	Name/Value Pairs	*Event(s): Order Accepted, Order Modified, Order Restatement, Simple Option Order Accepted, Complex Option Order Accepted, Complex Option Order Modified, Stock Leg Order, Option Order Modified, Complex Order Modified, Stock Leg Modified, Option Order Restatement* The order attributes field is a way to provide attributes of an order that are not necessarily the same as handling instructions. For example, the rank price of an order, or the participant with the best bid. **Allowed Values** **childOrderID** Requires value, e.g. childOrderID = 123456789 **FBT** Floor Broker Trade; Boolean value where presence indicates that the event is the result of a Floor Broker Trade. This can be used by an exchange to report an OOA event when a floor trade is executed. Firms are not currently required to report the corresponding event. The result is linkage errors that cannot be

Field Name	Data Type	Description
orderAttributes *(continued)*		repaired. The presence of this flag will exclude the event from linkage feedback and error rate calculation until such time as the corresponding events are required to be submitted by the firm. **NBBPAR** Participant at the best bid - requires a Participant ID, representing the participant at the best bid (e.g, NBBPAR=Par1) **NBOPAR** Participant at the best offer - requires a Participant ID, representing the participant at the best bid (e.g, NBOPAR=Par1) **pairedOrderId** Requires Text(40). In addition to the standard Text data type restrictions, Participants should avoid using the "at symbol," @ (ASCII decimal 64, hex 40). Participant-provided value that that will be present on the OOA, OCOA, OOM and OCOM events that are part of a customer-submitted cross order. The pairedOrderId must uniquely identify the paired orders within the Trade Date and Exchange. **PCTPX** Indicates that the limit price provided in the price field of the order is specified as a percentage of a benchmark price. For example, price=94.5 indicates that the final dollar price of the trade should be 94.5% of a benchmark price. **replacedOrderDate** Used when a new order is entered to replace a previously entered erroneous order. Reference to a date of a previously reported order that has a matching orderID. Requires a Date value, e.g. replacedOrderDate =20210930 **replacedOrderID** Used when a new order is entered to replace a previously entered erroneous order. Reference to a previously reported order that has a matching orderID. Requires a Text(40) value, e.g. replacedOrderID = 123456789ABC **RNKP** Rank Price - requires a Price value, representing the price used to rank the order in the book (e.g., RNKP=10.25). **Allowed Values: Cboe Legacy (C1) Only** *active 3/29/2019 – 10/4/2019* **MPID** Market participant ID, requires an alphanumeric(8) value. (e.g., MPID=A12345) **MeetExchangeID** Meet Exchange ID, requires a text(8) value. (e.g., MeetExchangeID=B76543) **Branch** Branch ID, requires a alphanumeric(8) value. (e.g., Branch=ABCD5) **BranchSeqNbr** The branch sequence number, requires an integer(10) value. (e.g., BranchSeqNbr=500321) **CorrespFirm** The corresponding firm, requires an alphanumeric(8) value. (e.g., CorrespFirm=987765B) **UserID** The user ID. Requires a text(8) value. (e.g., UserID=4321A) **Extensions** Order Extensions. Requires a text(256) value. **NBBOProtection** Specifies if the order is NBBO protected. Requires a Boolean value from one of the following choices: true, false. (e.g., NBBOProtection=false). **Allowed Values: Cboe** **AckSubLiquidity** This is a subset of the SubLiquidity values. Better prices are offered (in some cases) if an order is at the NBBO. This tells the member on order entry if their order did that. Requires a choice value (e.g., AckSubLiquidity=N) from the following list: **B** Bolt **G** SetterNoSize

Field Name	Data Type	Description		
orderAttributes ***(continued)***			**J**	Joiner
			N	Normal
			r	Persisted (GTC restatement)
			S	Setter
			U	Turner
		AddLiquidityOnly	Values used for "Post Only" orders. Requires a choice value (e.g., AddLiquidityOnly=A) from the following list:	
			A	Add only, don't remove liquidity
			B	Bypass removing hidden peg
			R	Allow removal
			L	don't remove at limit
		AllowPriceSlide	Describes what to do with an order if it locks/crosses with the NBBO. Requires a choice value (e.g., AllowSidePrice=M) from the following list:	
			S	allow slide and nerf
			R	no nerf and no slide
			L	allow slide no nerf
			P	price adjust
			m	multiple price adjust
			M	slide nerf unnerf when possible
			H	hide not slide
			N	don't re-scrape book at limit
			D	Slide Price
			E	Slide Price but no Nerf
			X	Don't Slide Don't Reject
			C	Bolt but no Nerf
			K	Cancel Back
			B	Bolt
		AuctionType	Auction type, used for fee purposes. Requires a choice value (e.g., AuctionType=H) from the following list:	
			O	open
			C	close
			H	halt
			I	IPO
			N	none
			G	GTHOpen
			V	Volatility
			U	ClosingCross
			P	Position Compression Cross (PCC)
			R	Related Futures Cross (RFC)
		BookLiquidity	Signifies whether the order is being added to the book. Requires a choice value from the following list:	
			A	Booked
			R	Not Booked
			X	Routed
			B	Booked Remainder
			Q	Wait
			C	Auction

Field Name	Data Type	Description		
orderAttributes *(continued)*			**P**	RemovedPending
		DeltaRefPrice	The value of the underlying as known by the submitter of the order. (Optionally present on a DAC order). Accepts a price value.	
		DeltaValue	The multiplier applied to the difference between the referencePrice and the closing price of the option's underlying value (specified per leg in the case of a complex order). (Optionally present on a DAC order). Accepts a value from -1.0000 to 1.0000.	
		Display	Display. Requires a choice value (e.g., Display=V) from the following list:	
			V	visible
			I	invisible
		Executable	Further describes the status of an order if it is/ is not yet live or executable. Can be updated with a modify event. Requires a choice value (e.g. Executable=W) from the following list:	
			E	order is executable
			P	order is route pending
			W	order in a wait state
			O	open auction MOO/LOO/LLOO + pre-open RHO
			C	close auction MOC/LOC/LLOC
			U	queued
			T	order is stop pending
			S	suspended
			Q	non executable visible quote
			D	pending queued
			I	Periodic Auction
			A	Step Up
			b	BAM Auction
			c	COA (Options only - Complex Order Auction - order is not currently executable as auction is not complete)
			q	QCC
			f	FOA – Flex Order Auction
			s	SAMAuction
			u	Closing Cross
			F	Floor
			L	Floor Local
			p	Position Compression Cross (PCC)
			r	Related Futures Cross (RFC)
		FloorOrderID	Contains the orderID of the Floor Broker PAR order for external reference (e.g., FloorOrderID= 123456789ABC)	
		MODR	Modify reason, requires a choice value (e.g., MODR=+) from the following list: (Note that in this list the acceptable values are surrounded by quotes because the list contains non alphanumeric values)	
			'P'	peg adjustment
			'C'	Cboe Market Close
			'+'	price was un-slid
			'L'	liquidity flag was changed (resting order routed away or fully delivered)

Field Name	Data Type	Description		
orderAttributes *(continued)*			**'R'**	user reduce (no loss of priority)
			'D'	adjustment of discretion price ONLY no loss in priority (midpoint discretionary peg orders)
			'U'	user other
			'-'	an external NBBO change (sip) caused some sort of change in the order
			'^'	Reroute (order lifted from book to reroute)
			'B'	un-bolt OR bolt-expire
			'W'	wash
			'T'	wait order
			'!'	reload of displaySize and loss of priority
			'K'	working price slid back to display price due to another market locking our protected quote
			‘S’	stop order
			‘A’	order routed away due to ROOC e.g. a few minutes before an open/close/ipo/halt auction
			‘E’	sweep SWPA or SWPB order after route plan has been developed
			'@'	Trading At Last
			'X'	Executable Status
			'Y'	Recovery
			‘F’	Floor Order
			‘2’	Clearing Failure
			‘r’	FloorUserCanceled
			‘q’	FloorEquityLegMatch
			‘p’	Periodic Auction (applicable for Cboe-BYX only)
		PriceType	Types or Prices. Allowed Values:	
			Unspecified	
			Cabinet	
			SubCabinet	
			CashSpread	
			DeltaAdjust (for Delta Adjust at Close order types)	
		PWASH	Prevent wash, more information about wash prevention. Requires a choice value (e.g., PWASH=P) from the following list:	
			N	do not prevent (none)
			F	prevent same firm match
			C	prevent clearing firm match
			P	prevent port-owner match
			X	prevent cross-firm match (equities exchanges only)
		REJA	Reject action, provides further information on action if the order can't be executed on entry. Requires a choice value (e.g., REJA=W) from the following list:	
			O	outbound
			R	reject
			Z	BZX only
			J	BYX only
			N	NASDAQ only
			A	ARCA only

Field Name	Data Type	Description
orderAttributes *(continued)*		**C** NSX only
		M CHX only
		X PHLX only
		K BEX only
		E ISE only
		U AMEX only
		D EDGA only
		G EDGX only
		Y NYSE only
		T TRACO only
		L FLOW only
		W CBSX only
		V DATA only
		H CTWO only
		S NOBX only
		F MIAX only
		Q ICRS only
		g GMNI only
		r Dark Reject
		a Dark Auto
		x MEMX Only
		P Periodic
		t Wait
		p Primary Only
		b BXE Only
		c CXE Only
		q TRQX Only
		h XHFT Only
		l CboeSelect
		e PERL Only
		m MERC Only
		i IEX Only
		d EMLD Only
		l LTSE
		w SPHR Only
		REROUTE Reroute, specifies whether or not we can reroute an order (route it a second time after it has been booked), if the NBBO goes locked or crossed. Requires a choice value (e.g. REROUTE=N) from the following list:
		N none
		L onLock
		C onCross
		K onLockOddLot
		RESTA Resting action, specifies whether this order will go onto the Cboebook or be routed away to post on somebody else's book. Requires a choice value (e.g., RESTA=I) from the following list:
		I Integrated, will rest on the Cboe book (though may not be resting at the point of the OA if it is a routed order,

Field Name	Data Type	Description		
orderAttributes *(continued)*				may never rest if it is a routed IOC)
			A	PostAway, will rest on another exchange’s book, looking like a routed order that hasn’t come back to Cboe
			D	Dark
			E	Expose
			T	Stepup
			F	Floor
		ROUTESTRAT	The route strategy used internally in the Cboe system. Requires a choice value (e.g., ROUTESTRAT=O) from the following list:	
			O	default, let the router select the strategy
			F	failover strategy for use when the router has a NoQuote condition
			L	legacy (emulate the behavior of the old router)
			C	cycle (sequentially route walking depth of book)
			K	dark liquidity scan
			T	toggle (causes the router to cycle through various other strategies on a per-order basis)
			B	ParT (Parallel Top)
			S	ParD (Parallel Depth), exhaust price level before proceeding
			2	Par2D (Parallel Depth including multiple price levels)
			M	Slim (predefined set of markets, DRT and then ALL)
			m	SlimPlus (Slim, but send to BYX before scraping the local book)
			R	Trim, scrape local book on way in (predefined set of markets, DRT, and then another predefined set of markets)
			r	Trim, but don't scrape local book on way in
			P	Trim2
			p	Trim2, but don't scrape local book on way in
			Q	Trim3
			q	Trim 3, but don't scrape local book on way in
			G	MidPoint routing
			b	SWEEPB (Route to market centers to remove least amount of protected quote shares so order can post. No executions occur is order size too small to completely remove all protected quotes)
			i	Book + IOC/(Day effective 10/21/14) Nasdaq
			t	Book + DRT + IOC/(Day effective 10/17/14) NYSE
			x	Book + IOC/(Day effective 10/17/14) NYSE
			f	Book + IOC LavaFlow
			a	ISO Sweep of all protected markets (similar to CboeParallel T)
			o	ROBB
			c	ROCO
			l	ROUC
			Z	RMPT
			z	IOCM
			u	Dark lit

Field Name	Data Type	Description
orderAttributes *(continued)*		**W** Lit sweep **D** Directed **A** ALLB **n** CLNK **RTLM** Route to listing market, specifies whether the order can be routed to the opening auction, the closing auction, or both on the listing exchange. Requires a choice value (e.g., RTLM=O) from the following list: **N** none **O** only on the open **C** only on the close **B** both (on the open or close) **H** Halt **Allowed Values: Cboe – BYX** *in addition to values defined above* **CrossTradeFlag** Requires a choice value (e.g., CrossTradeFlag=1) from the following list: **1** PeriodicAuctionOnly **2** PeriodicAuctionEligible **LockOrderForAuction** Requires a choice value (e.g., lockOrderForAuction=T) from the following list: **F** False (Do not lock the order) **T** True (PeriodicAuctionEligible order is locked for cancel/modify once an action starts) **Allowed Values: BOX** **FLEX** Event is associated to a FLEX Option **ST** Requires a choice from the following list: **InOrderBook** **Executed** **Exposed** **ToOla** **Directed** **CancelPending** **TraderCancelled** **Eliminated** **EliminatedOutOfLimit** **EliminatedByCircuitBreaker** **EliminatedOnDisconnection** **EliminatedByMarketControl** **EliminatedDueToUnpricedLeg** **EliminatedDueToTradingRestriction** **CancelledBySupervisor** **Received** **EliminatedDueToTradeLimitExceeded** **EliminatedDueToTradeActivityLimitExceeded** **EliminatedDueToMaximumNbTriggersLimitExceeded**

Field Name	Data Type	Description
orderAttributes ***(continued)***		**EliminatedDueToDrillThroughProtection** **Allowed Values: CHX** (Legacy as of 11/2019) **SettlementType** Requires a choice value (e.g., SettlementType=0) from the following list: **0** REG - Regular Way **1** CASH - Cash **2** NXT - Next Day **3** T+2 - Trade Date + 2 **4** T+3 - Trade Date + 3 **5** T+4 - Trade Date + 4 **6** FUT - Future **7** WI - When and If Issued **8** SO - Sellers Option **9** T+5 - Trade Date + 5 **S** SLR - Settlement Days **FutureSettlementDate** Requires value (e.g., FutureSettlementDate=YYYYMMDD) when SettlementType is 6 or S. Value is a date in format YYYYMMDD. **FutureSettlementDays** Requires value (e.g., FutureSettlementDays=4) when settlementType is S. Value is an integer. It is the number of settlement days. **ExpireSeconds** Requires value (e.g., ExpireSeconds=3) when timeInForce is GFS. Value is an integer. It is the number seconds for the good-till-seconds order. **ExpireDate** Requires value (e.g., ExpireDate=YYYYMMDD) when timeInForce code is GTD. Value is an integer. It is the date for the good-till-date order. **PegDiff** Requires value (e.g., PegDiff=2) for SNAP Auction market peg order. Value is an integer. It is the number of ticks for the symbol. **CancelOnSNAPAuctionFlag** Requires value (e.g., CancelOnSNAPAuctionFlag=Y) for an order. **Y** When a SNAP Auction is invoked, the order will not participate in the SNAP Auction **N** When a SNAP Auction is invoked, the order will participate in the SNAP Auction **SNAPMinExecRequiredFlag** Requires value (e.g., SNAPMinExecRequiredFlag=Y) for a SNAP Auction order. **Y** Minimum SNAP Auction threshold required **N** Minimum SNAP Auction threshold not required **SNAPConvertToAOOFlag** Requires value (e.g., SNAPConvertToAOOFlag=Y) for a SNAP Auction order. **Y** Convert to SNAP Auction Only Order if a SNAP Auction has already started by another order. **N** Cancel Order if a SNAP Auction has already started by another order. **SNAPAOOOneAndDoneFlag** Requires value (e.g., SNAPAOOOneAndDoneFlag=Y) for a SNAP Auction order. **Y** SNAP Auction Only Order will only participate in one SNAP Auction, then it will be canceled.

Field Name	Data Type	Description
orderAttributes *(continued)*		**N** SNAP Auction Only Order will participate in every SNAP Auction. **CreationTimestamp** Requires value when the eventTimestamp is different from the creation timestamp. (e.g., CreationTimestamp=20180415T143055.123456789) **SNAPAuctionOrder** Requires a choice value (e.g., SNAPAuctionOrder=s) from the following list: **s** SNAP Auction Order. Order used to potentially initiate a SNAP Auction. **Allowed Values: LTSE** **R** Boolean; Presence indicates that an order is designated as a Retail Order. **RDM** Reserve Display Method **1** Initial **3** Randomized by Size (e.g. RDM=3) **RRT** Reserve Replenishment Time - Replenishment time: **1** Immediate **2** Random (e.g. RRT=1) **RPF** Re-Price Frequency: **0** Single Reprice **2** Continuous (e.g. RPF=0) **RBH** Re-price Behavior: **1** RepriceLockCancelCross **2** RepriceLockREpriceCross (e.g. RBH=1) **STP** Self Trade Prevention; Requires a choice value (e.g., STP=0) from the following list: **0** CancelNewest **1** CancelOldest **2** DecrementAndCancel **3** CancelBoth **4** CancelSmallest **Allowed Values: NYSE Options** **BOLD** Broadcast Order Liquidity Delivery **ClearTheBook** Clear interest ahead of Outcry Trade. Requires order id of outcry order (ClearTheBook=281474976714831) **ClientID** Identifier used for Cross-MPID Self Trade Prevention **Covered** Contra Side of CUBE/QCC/C2C **Exposed** Initiating/RFQ Side of CUBE. Initiating Side of QCC/C2C **PackageLinkID** Outcry order linked to another outcry order; Requires a value (PackageLinkID=281474976714831) **ProactiveIns** Proactive If Locked - actively route if displayed interest is locked by another exchange **PublishQuantity** Reserve Order; Requires a value (PublishQuantity=100) **Reserve** (Legacy 10/2023) **STP** (Legacy 10/2023) **STP-C** Self Trade Both orders **STP-D** STP - Cancel Decrement

Field Name	Data Type	Description	
orderAttributes *(continued)*		**STP-N**	Self Trade Newest order
		STP-O	Self Trade Oldest order
		Allowed Values: NYSE Equities	
		72DCross	Floor broker cross tradeable on NBBO
		Border	(Legacy as of 8/2019)
		BrokerOrder	Floor Broker Order
		ClientID	Internal identifier used for Cross-MPID Self Trade Prevention purposes. Requires a value (e.g., ClientID=AAA1)
		CROWD	(Legacy as of 8/2019)
		DMM-Manual	Manual Entered DMM activity
		dOrderAucPrice	Auction Limit for Dorder; Requires a value (dOrderAucPrice=100.00)
		MMID	Market Maker ID; Requires a value (MMID=CSLM)
		ProactiveIns	Proactive If Locked - actively route if displayed interest is locked by another exchange
		Qorder	Market Maker Q-Order (NYSE ARCA)
		Reserve	Reserve Order; Requires a value (Reserve=100)
		RMO	Retail order
		Sorder	(Legacy as of 8/2019)
		STP	(Legacy as of 8/2019)
		STP-C	Self Trade Both orders
		STP-D	Self Trade and decrement
		STP-N	Self Trade Newest order
		STP-O	Self Trade Oldest order
		YGOrder	Yielding order
		SelfTrade	Part of back processing only for trade dates 3/29/2019 to 6/21/19. This value was accepted between the processing dates of 7/26/19 to 8/30/19. (Legacy as of 8/2019)
		MinQty	Minimum Trade Size; Requires a value (MinQty=100)
		MFS	Minimum Fill Size; Requires a value (MFS=100)
		PriceOffset	Price Offset; Requires a value (PriceOffset=0.01)
		MinTriggerSize	<OppSideMinSizeTriggerValue>; Requires Unsigned value (e.g., MinTriggerSize=1000)
		MinPegSize	<MinPegSize>; Requires Unsigned value (e.g., MinPegSize=1000) (Legacy as of 8/2019)
		MaxDiscVol	<MaxDiscVol>; Requires Unsigned value (e.g., MaxDiscVol=1000) (Legacy as of 8/2019)
		CeilingFloorPrice	<Peg_Price> ; Requires Price value (e.g., CeilingFloorPrice=0.01) (Legacy as of 8/2019)
		DiscPriceRange	<disc_price_range>; Requires Price value (e.g., DiscPriceRange=0.01) (Legacy as of 8/2019)
		TypeOfInterest	Requires a choicevalue from the following list (Legacy as of 8/2019) **DOTR** **CO** **EQAA** **EQBB** **EQDA**

Field Name	Data Type	Description
orderAttributes ***(continued)***		**EQDB** **EQGA** **RQGB** **SQAA** **SQBB** **SQDA** **SQDB** **DSQCC** **SQDC** **Allowed Values: IEX** **RoutingStrategy** Allowed values from the following list: **u** Router **s** Router Basic **MinQtyInstruction** Allowed values from the following list: **C** Composite **M** Minimum Execution Size with Cancel Remaining **A** Minimum Execution Size with AON Remaining **AntiInternalizationGroupId** Used for wash trade prevention. Allowed any two alphanumeric characters or the two-character string "--". [A-Za-z0-9][A-Za-z0-9] Depending upon the value used, these will be used to identify orders which have elected to not trade with identically marked orders from the same firm. The lower case and upper case characters are two distinct values. For example, "a1" and "A1" will be two distinct values. “--" Represents free to trade with anyone. **AIQ** (Anti-Internalization Qualifier) Allowed Values from the following list: **CO** Cancel Older order (existing value) **CN** Cancel Newest Order **CB** Cancel Both Orders **CS** Cancel Smallest Orders **DL** Decrement Larger Order **CancelOrSlide** Requires a choice value (e.g. CancelOrSlide=0) from the following list: **0** Slide **3** Cancel order instead of slide **OverrideAIQDLO** Override the default AIQ DLO functionality. Boolean value where presence indicates override is turned on for the order and absence indicates override is turned off. **PegDifference** Accepts a Price value. **Allowed Values: NASDAQ – NOBO, PHLX, NOM, ISE, GEMX, MRX** **Persist** Boolean **PrimarySide** Boolean **PrivateReference** Text<20> **BrokerText** Text<6> **BranchSeqNum** Text<20> **Text** Text<64>

Field Name	Data Type	Description	
orderAttributes *(continued)*		**FloorBrk**	Text<6>
		Tag1AcctId	Text<32>
		tag1AcctId	Text<32> - Part of back processing only for trade dates 3/29/2019 to 6/21/19. This value was accepted between the processing dates of 7/26/19 to 8/30/19.
		CrossClOrderId	Text<64>
		CrossOrderId	Text<64>
		StortSaleInd	Value must be on of the following **1** SHORT SALE **2** SHORT SALE EXEMPT
		StockCapacity	Value must be one of the following **1** Agent **2** Principal **3** Riskless Principal
		Allowed Values: NASDAQ – BX, PSX, NSDQ	
		AIQ	Anti-Internalization Flag; AIQ values with lowercase letters represent anti-internalization at the organization level across MPIDs. Uppercase values represent anti-internalization at single MPID/non-organizational level. Value must be one of the following: **B** Cancel Both (single MPID level) **D** DecrementBoth (single MPID level) **N** Disabled (single MPID level) **O** Cancel Oldest (single MPID level) **W** Cancel Newest (single MPID level) **Y** DecrementBothNoDetails (single MPID level) **b** Cancel Both (organization level) **d** DecrementBoth (organization level) **n** Disabled (organization level) **o** Cancel Oldest (organization level) **w** Cancel Newest (organization level) **y** DecrementBothNoDetails (organization level)
		AIQGroupID	Order Identifier used to prevent two orders with the same identifier from executing against each other. Value must be 2-character alphanumeric.
		CrossType	Value must be one of the following **0** None **1** Open **2** Halt **3** Close **4** Pause **5** Supplemental **6** Retail **7** ELO **8** After-Hours
		CustomerType	Value must be one of the following **1** Retail Designated **2** Non Retail Designated

Field Name	Data Type	Description
orderAttributes *(continued)*		**LULDFLAG** Value must be one of the following **C** LULD Closing Cross Order **l** LULD Reprices *(note that this is a lowercase "L")* **L** LULD **PriorityUpdate** NMS price sliding for re-entered order. Boolean. Presence indicates that the system replaced an order that was originally hidden due to NMS lock/cross rules with a displayed order (at the same price) after movement of the NBBO: **Y** **N** **OrgID** Optional value to provide additional enhanced support for Self-Trade Prevention across MPIDs. Value can contain up to 4 alphanumeric characters. **OriginalChildOrderID** = <ORDERID> **PEGOFFSET** Peg offset; Requires a Price value. (Ie. PEGOFFSET=0.01, PEGOFFSET=-0.001) **Allowed Values: LTSE** **AntiInternalizationGroupId** Used for wash trade prevention. Allowed any two alphanumeric characters or the two-character string "--". Depending upon the value used, these will be used to identify orders which have elected to not trade with identically marked orders from the same firm. The alphanumeric characters are distinct values. "--" represents free to trade with anyone. **Allowed Values: MEMX** **R** Boolean; Presence indicates that an order is designated as a Retail Order. **RDM** Reserve Display Method **1** Initial **3** Randomized by Size (e.g. RDM=3) **RRT** Reserve Replenishment Time - Replenishment time: **1** Immediate **2** Random (e.g. RRT=1) **RPF** RePrice Frequency: **0** Single Reprice **2** Continuous (e.g. RPF=0) **RBH** Reprice Behavior: **1** RepriceLockCancelCross **2** RepriceLockREpriceCross (e.g. RBH=1) **STP** Self Trade Prevention; Requires a choice value (e.g., STP=0) from the following list: **0** CancelNewest **1** CancelOldest **2** DecrementAndCancel **3** CancelBoth **4** CancelSmallest **Allowed Values: MEMXOP**

Field Name	Data Type	Description
orderAttributes ***(continued)***		**PAF** Price adjust Frequency; Requires a choice value (e.g., PAF=0) from the following list: **0** Single Reprice **1** Continuous **PAB** Price adjust behavior; Requires a choice value (e.g., PAB=0) from the following list: **1** Reprice Lock Cancel Cross **2** Reprice Lock Reprice Cross **MTP** Reserve Replenishment Time - Replenishment time: **0** Cancel Newest **1** Cancel Oldest **3** Cancel Both **Allowed Values: MIAX PEARL Equities** **STP** Self Trade Protection; Requires a choice value (e.g., STP=1) from the following list: **1** N/A **2** STP Cancel Newest **3** STP Cancel Oldest **4** STP Cancel Both **5** STP Decrement and Cancel **Display** Requires a choice value (e.g., Display =1) from the following list: **1** All or part of the order can be displayed **0** The order can never be displayed **PriceSlide** Requires a choice value (e.g., PriceSlide =2) from the following list: **1** N/A **2** NoPriceSliding **3** Once **4** OnceButCancelIfCrossedAtEntry **5** MultipleTimes **Attributable** Requires a choice value (e.g., Attributable=3) from the following list: **1** NotAttributable **2** AttributedToFirmMpid **3** AttributedToRetail
orderID	Text (40)	*Event(s): Order Accepted, Route, Modified, Canceled, Trade (sideDetails), Fill, Cancel Route, Modify Route and Restatement events, Simple Option Order Accepted, Complex Option Order Accepted, Stock Leg Order, Option Route, Complex Option Route (OCOR), Option Order Modified, Complex Option Order Modified, Option Order Canceled, Modify Option Route, Option Cancel Route, Simple Option Trade, Stock Leg Fill, Option Order Restatement and Options Post Trade Allocation events, Note (NOTE)* The internal order ID assigned to the order by the exchange.
orderType	Choice	*Event(s): Order Accepted, Order Routed, Order Modified, Order Restatement, Order Modify Route, Simple Option Order Accepted, Complex Option Order Accepted, Stock Leg Order, Option Order Modified, Complex Option Order Modified, Option Route, Option Order Restatement, Modify Option Route events* The order type defines the type of order being placed, and must be exactly one of

Field Name	Data Type	Description
orderType *(continued)*		the permitted values. Some values are exchange specific. This document details the technical specifications for what is reported in this field, not necessarily how to determine what value to be included in each report. See the CAT website for exchange-specific guidance on how to determine which values to use for reporting specific orders. Note: An asterisk (*) indicates that the value represents a Limit Order. **Allowed Values:** **AMPEG** Alt Midpoint Peg - pegs to less aggressive of midpoint or 1 tick inside the NBBO **CAB** Cabinet **LMT** Limit* **LOB** Limit or Better* **LOC** Limit on Close* **LOO** Limit on Open* **MIT** Market If Touched **MKT** Market **MOC** Market on Close **MOO** Market on Open **MDPEG** Midpoint Discretionary Peg - a primary peg, but has discretion to the midpoint of the NBBO **MPEG** Midpoint Peg **MMPEG** Market Maker Peg - will peg at 8%, 20%, or 28% of the NBBO depending on symbol and time of day (follows the LULD bands). Designed to allow MMs to satisfy their quoting obligations without stub orders **PPEG** Primary Peg **RPEG** Market Peg **RTPEG** Route Peg - Non-displayed primary peg order that only interacts with orders that are about to be routed out with size <= peg order size **SOL** Solicitation **STL** Stop Limit* **STP** Stop **Allowed Values: NYSE Options** **AutoMatch** **LimitCross** * **Allowed Values: NYSE Equities** **LDPEG** **LimitCross** * **Peg** **Allowed Values: IEX** **CDPEG** CorporateDiscretionary; discretionary Peg marked as corporate buyback (10b-18). **DLMT** Discretionary Limit*

Field Name	Data Type	Description
orderType *(continued)*		**DPEG** Discretionary Peg **FMPEG** FixedMidpointPeg; MidpointPeg that does not re-price based on changes to the NBBO **OPEG** Offset Peg **RDPEG** RetailDiscretionary; discretionary Peg marked as retail order. **RLP** RetailLiquidityDiscretionary; discretionary Peg marked as retail liquidity provider. **RLPM** RetailLiquidityMidpoint **RMPEG** RetailMidpoint; midpoint Peg marked as retail order. **Allowed Values: MIAX PEARL Equities** **MidPointPegAvailWhenLocked** Order is available to trade when the market is locked **MidPointPegUnavailWhenLocked** Order is not available to trade when the market is locked **PrimaryPegAvailWhenLocked** Primary PEG order is available to trade when the market is locked. **PrimaryPegUnavailWhenLocked** Primary PEG order is not available to trade when the market is locked.
originalAskQuoteID	Text (40)	*Event(s): Option Quote (OQ)* When onlyOneQuote=False, this field must be populated when the ask from this record replaces a previously submitted ask. This field must not be provided for a ask that does not replace a previous ask, and it should never be populated for an bid. When onlyOneQuote=True no value is necessary for this field.
originalModifierCode	Text (4)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Four-byte trade modifier as entered by the firm.
originalOrderDate	Date	*Event(s): Order Restatement, Option Order Restatement* This field represents the most recent trading day for which the order was active. Note that this may not be the date when the order was originally accepted. If the order has been active for multiple trading days, this field must reference the most recent trading day when the order was active.
originalOrderID	Text (40)	*Event(s): Order Modified, Order Restatement, Option Order Modified Event, Complex Option Order Modified Event, Stock Leg Modified, Option Order Restatement* The most recent internal order ID before the modify / replacement created a new order ID.
originalQuoteID	Text (40)	*Event(s): Quote Event (OQ)* The most recent quoteID of the existing quote before being updated or replaced.
Participant ID	Text (40)	Valid Participant ID values. Note that participants will use their Participant ID as their Reporter ID. **Allowed Values** **BZX** Cboe BZX Equities **BZXOP** Cboe BZX Options **BYX** Cboe BYX Exchange **BOX** BOX Options Exchange **C2** Cboe C2 Options **CBOE** Cboe Exchange **CHX** NYSE CHX

Field Name	Data Type	Description
Participant ID ***(continued)***		**EDGA** Cboe EDGA Exchange **EDGX** Cboe EDGX Equities **EDGXOP** Cboe EDGX Options **FINRA** Financial Industry Regulatory Authority **GEMX** Nasdaq GEMX **MRX** Nasdaq MRX **ISE** Nasdaq ISE **IEX** Investor's Exchange **MIAMI** Miami International Securities Exchange **PEARL** MIAX PEARL **EMLD** MIAX Emerald **SPHR** MIAX Sapphire Options **BX** Nasdaq BX Equities **NOBO** Nasdaq BX Options **PHLX** Nasdaq PHLX Options **PSX** Nasdaq PHLX Equities **NSDQ** The NASDAQ Stock Market **NOM** Nasdaq Options Market **NSX** NYSE National **NYSE** The New York Stock Exchange **AMEROP** NYSE American Options **AMER** NYSE American **ARCAOP** NYSE ARCA Options **ARCA** NYSE ARCA Equities **LTSE** Long Term Stock Exchange **PEARLEQ** MIAX PEARL Equities **BSTX** Boston Security Token Exchange **MEMX** Members Exchange **MEMXOP** Members Options Exchange
positionTransferFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Special processing flag indicating that the transaction is for internal FINRA use only and should not be disseminated. **Allowed Values** **3** Section 3 Fees **A** Audit Trail Only **N** None **P** Position Transfer
previousTradeFinraContraControlDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Date of the previous trade in a chain of corrections on the contra side trade report.
previousTradeFinraContraControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Number of the previous trade in a chain of corrections on the contra side trade report.
previousTradeFinraControlDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* FINRA Control Date of the previous version of the trade.

Field Name	Data Type	Description
previousTradeFinraControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* FINRA Control Number of the previous version of the trade.
priceOverrideCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies if a price validation test was overridden when the trade was entered into ACT. (When trades are entered into ACT, they are validated for reasonableness against a Price Validation Table. The Price Override widens the validation range). **Allowed Values** **O** Subscriber Override* **V** Supervisory Contract Override **X** Supervisory Override *(default) Value set by the ACT System for all CQS Issues if not "X" or "V"
price	Price	*Event(s): Order Accepted, Route, Modified, Modify Route or Restatement events, Simple Option Order Accepted, Complex Option Order Accepted, Stock Leg Order, Option Order Modified, Complex Option Order Modified, Option Route, Complex Option Route (OCOR), Modify Option Route, Option Order Restatement* The limit price of the order. For a complex option, this is the net price of the order, which can be either positive, negative, or zero. Event(s): Order Trade, Order Fill, Trade Break, Trade Correction Trade/fill price of the trade/fill. Event(s): Post Trade Allocation The price of the allocation.
primaryDeliverable	Symbol	*Reference Data: Option Series Dictionary Entry (OSDE)* The symbol for the primary deliverable component of the option, in the symbology of the listing exchange for that symbol. Alternatively, if a symbol dictionary is provided, a valid alias could be used.
publishIndicatorCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies if the trade is media reportable or not (could differ from the mediaReportedFlag for odd lot trades). **Allowed Values** **Y** Media Report Eligible **N** Not Media Report Eligible
putCall	Choice	*Reference Data: Option Series Dictionary Entry (OSDE)* Specifies if this simple option or option leg is a put or call. **Allowed Values** **Put** **Call**
quantity	Unsigned	*Event(s): Order Accepted, Route, Modified, Canceled, Trade, Fill, Modify Route, Order Restatement events; Simple Option Order Accepted, Complex Option Order Accepted, Stock Leg Order, Option Order Modified, Complex Option Order Modified, Stock Leg Modified, Option Route, Complex Option Route (OCOR), Option Order Canceled, Simple Option Trade, Stock Leg Fill, Modify Option Route, Option Order Restatement events* The quantity of the order.
quoteCondition	Text (8)	*Event(s): Equity Best Bid and Offer Event (EBBO)* Indicator used to determine whether a quote is eligible to participate in the NBBO.

Field Name	Data Type	Description
quoteID	Text (40)	*Event(s): Note (NOTE), Equity Best Bid and Offer Event (EBBO), Options Quote, Quote Cancel, and Options Trade (sideDetails) events* The ID assigned to this quote by the exchange to uniquely identify the quote. For two-sided quote reporting where each side has its own quote ID, this will be the buy side quote ID.
quoteInstructions	Name/Value Pairs	*Event(s): Equity Best Bid and Offer Event (EBBO)* Represents any additional instructions or attributes for the quote. **Allowed Values** *Codes to be provided.*
ratio	Unsigned	*Reference Data: Complex Option Dictionary Entry (CODE)* The ratio quantity of a complex option leg, relative to other legs. Ratios must already be reduced to the smallest units possible.
reason	Text (255)	*Event(s): Trade Break, Trade Correction, Option Trade Break, Option Trade Correction, Post Trade Allocation* Free format text field, with reason for the trade break or correction.
recordLoadDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date the record was created.
recordUniqueIdentifier	Text (31)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* FINRA-assigned unique identifier for each Reported Trade record.
referenceNumber	Text (20)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* User-defined trade reference number.
referenceReportingFacility	Text (6)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Reference Reporting Facility.
rejectContext	Name/Value Pairs	*Event(s): Reject Message Event (RME)* A list of key/value pairs, providing machine parseable data for the notation. The attributes are not defined in the spec, and can be any values as long as they conform to the format for a list of name/value pairs. Any additional information can be provided in this field. For example, the entire rejected message in Tag=Value format.
rejectReason	Choice	*Event(s): Reject Message Event (RME)* Code representing the reason why the order was rejected. Codes are exchange-specific. *The following values will be effective in the Production environment on December 5, 2022, in conjunction with the availability of the Reject Message Event.* **Allowed Values: BOX** **Syntax** **Technical** **Business** **Regulation** **Unknown** **Allowed Values: CBOE** **+** Risk Management Trigger at EFID Group Level

Field Name	Data Type	Description
rejectReason *(continued)*		**4** Order cannot be entered, cancelled, or modified during cutoff time for volatility series **A** Admin **c** Only Closing transactions for this symbol allowed **D** Duplicate ID **d** Order Entry disallowed **F** Modified by a Floor Cancel **f** Risk Management Trigger at Firm Level **g** No odd lot IPOs **G** Ran out of liquidity to execute against **H** Halted **I** Tried to send order to DR site **J** Too late to cancel **K** Exceeded Order Rate Threshold **L** Order would lock or Cross NBBO **M** Exceeded Client Specified Maximum Order Size **m** Market Access Risk Limit Exceeded in Router **N** Ran out of liquidity to execute against **o** Exceeded maximum open orders permitted **O** Supplied Order ID doesnt match a known order **P** Cant modify an order that is routed away **q** Quote Unavailable **Q** Waiting for first trade before allowing executions **R** Route unavailable **s** Risk Management Trigger at Symbol Level **S** Short price violation **T** Order would have caused a trade through violation **u** Tried to rest through LULD bands **U** User **V** Execution would cause a Wash Trade **w** AddLiquidityOnly Order tried to unslide but would have resulted in removing liquidity **W** AddLiquidityOnly Order would have removed liquidity **x** Crossed Market Protection **X** Order Expired **y** Message Received During Replay **Y** Symbol not supported **z** Session End **Allowed Values: Cboe - CBOE, EDGXOP, C2, BZXOP** **qa** Admin **qc** InvalidCapacity **qd** CloseOnly **qe** InvalidOrderEntryOperatorId **qi** InvalidCtiCode **qm** InvalidWashMethod **qn** ExceededMaxNotionalValuePerOrder **qo** InvalidOpenClose **qr** InvalidRemove

Field Name	Data Type	Description
rejectReason *(continued)*		**qs** InvalidSide **qt** InvalidSendTime **qx** ExceededMaxSizePerOrder **qy** InReplay **qz** InvalidReduceSizeInstruction **qC** InvalidClearing **qD** InvalidWashId **qE** InvalidSessionEligibility **qF** NotEnabled **qI** IncorrectDataCenter **qK** OrderRateThreshold **qL** InvalidNumberOfQuoteLegs **qM** SymbolsNotOnSameMatchingEngine **qO** InvalidManualOrderIndicator **qP** InvalidPostingInstruction **qQ** InvalidQuoteUpdateId **qR** RiskRootIsNotTheSame **qS** SymbolNotFound **qU** SymbolRangeUnreachable **qW** InvalidWashPreventType **Allowed Values: IEX** BrokerCredit BrokerOption DuplicateOrder EntryTimeIsNotToday ExceededMaxNotionalPerOrder ExceededMaxSharesPerOrder ExchangeClosed InvalidAIQGroup InvalidAuctionOrder InvalidDisplayRange InvalidExecInst InvalidGoodTillTimeOrder InvalidMaxFloor InvalidMinQtyInstruction InvalidOddLot InvalidOrderQty InvalidPegDifference InvalidPrice InvalidTagCombinationForAIQ InvalidTagCombinationForCorporateDiscretionary InvalidTagCombinationForDirected InvalidTagCombinationForDiscretionaryLimit InvalidTagCombinationForDiscretionaryPeg InvalidTagCombinationForFOK InvalidTagCombinationForIOC InvalidTagCombinationForISO InvalidTagCombinationForMarketMakerPeg InvalidTagCombinationForMarketOrder InvalidTagCombinationForMarketPeg

Field Name	Data Type	Description
rejectReason ***(continued)***		InvalidTagCombinationForMidpointPeg InvalidTagCombinationForMinQty InvalidTagCombinationForOffsetPeg InvalidTagCombinationForPeggedOrder InvalidTagCombinationForPrimaryPeg InvalidTagCombinationForPostOnlyOrder InvalidTagCombinationOrSessionForOrder InvalidTargetForRouteDirected InvalidTradeNowInstruction ISONotAllowed MarketOrdersNotAllowed MissingLocate MPIDUnknownOrDisabled NoLiveOrderFound OnlyTestSymbolsAllowed OrderAlreadyInPendingStatus OrderExceedsLimit OrderHasNotChanged OrderInvalidInCurrentMarketSession OrderInvalidInPreMarketSession OrderLockedIn OrderParametersDoNotMatch PostMarketNotAllowed PreMarketNotAllowed RecoveryInProgress RetailOrdersNotAllowed RouteDirectedNotAllowed RoutingNotAllowed SymbolHalted SymbolNotActive SystemError TooLateToCancel TooLateToEnter UnexpectedExpireTime UnknownBroker UnknownSymbol UnsupportedAmendment UnsupportedOrderParameters **Allowed Values: LTSE** **001** UnknownSymbol **002** ExchangeClosed **003** SymbolHalted **004** NotionalRiskLimitExceeded **005** InvalidTagNumber **006** RequiredTagMissing **007** ShortOrShortExemptOrderMustSetLocateReqdFalse **008** TagSpecifiedWithoutValue **009** IncorrectValueForField

Field Name	Data Type	Description
rejectReason *(continued)*		**010** InvalidOrderQuantity **011** InvalidMinQuantOrder **012** InvalidPrice **013** InvalidISOOrder **014** InvalidClOrdID **015** IncorrectDataFormatForValue **016** InvalidSenderCompID **017** InvalidMsgType **018** ISONotAcceptedForThisOrder **019** OrderInvalidInCurrentMarketSession **020** OrderInputInvalidOnDropSession **021** InvalidTIF **022** OrderQuantityTooLarge **023** OrderNotionalTooLarge **024** UnknownOrderToReplace **025** UnknownOrderToCancel **026** InvalidFieldDuringCxlRepl **Allowed Values: LTSE** (*active starting on 7/29/2024)* **1001** InvalidSymbol **1002** ExchangeClosed **1003** OrderExceedsLimit **1006** DuplicateOrder **1018** InvalidPriceIncrement **1019** NoNBBOAvailable **1020** NotionalValueExceedsThreshold **1022** BlockSellShortRiskRuleViolated **1023** HardToBorrowSecurityRiskRuleViolated **1027** MaxNotionalValuePerOrderRiskRuleViolated **1099** Other **1100** MissingSymbol **1101** MissingLocateReqd **1102** InvalidLocateReqd **1103** MissingClOrdId **1104** InvalidClOrdId **1105** MissingSide **1106** InvalidSide **1107** MissingOrderQty **1108** InvalidOrderQty **1109** MissingOrderType **1110** InvalidOrderType **1111** MissingTimeInForce **1112** InvalidTimeInForce **1113** MissingOrderCapacity **1114** InvalidCapacity **1115** MissingExecInst **1116** MissingLimitPrice **1117** InvalidLimitPrice **1118** MissingMaxFloor **1119** InvalidMaxFloor **1120** MissingReserveReplenishAmountType **1121** InvalidReserveReplenishAmountType **1122** MissingReserveReplenishTimeType **1123** InvalidReserveReplenishTimeType **1124** MissingRandomReplenishmentValue **1125** InvalidRandomReplenishmentValue **1126** InvalidRandomReplenishValueForReserveType **1127** MissingRepriceFrequencyType **1128** InvalidRepriceFrequencyType **1129** MissingRepriceBehaviorType

Field Name	Data Type	Description
rejectReason *(continued)*		**1130** InvalidRepriceBehaviorType **1131** InvalidRepriceBehaviorForRepriceFrequency **1132** MissingCustomerCapacityType **1133** InvalidCustomerCapacityType **1134** MissingExpireTime **1135** InvalidExpireTime **1136** MissingPegType **1137** InvalidPegType **1138** InvalidModifierForOrderType **1139** InvalidModifiersCombination **1140** InvalidTradingSessionForOrderType **1141** InvalidTimeInForceForOrderType **1142** InvalidModifierForPegType **1143** InvalidMinQty **1144** InvalidExecInst **1145** InvalidMpid **1146** SymbolHaltedOrPaused **1147** BlockISORiskRuleViolated **1148** BlockSessionRiskRuleViolated **1149** BlockNonTestSymbolsRiskRuleViolated **1150** MaxSharesPerOrderRiskRuleBreach **1151** PricePercentCollarRiskRuleViolated **1152** PriceValueCollarRiskRuleViolated **1153** MaxADVPercentPerOrderRiskRuleBreach **1154** DailyGrossNotionalExposureRiskRuleBreach **1155** DailyNetNotionalExposureRiskRuleBreach **1156** MaxNumDuplicateOrdersRiskRuleBreach **1157** MaxOrderRateRiskRuleBreach **1158** RestrictedSecurityRiskRuleViolated **1159** InvalidSelfTradePreventionConfiguration **1160** InvalidSelfTradePreventionType **1161** InvalidRiskGroupId **1162** FirmDisabled **1163** MPIDDisabled **1164** AccountDisabled **1165** CannotTradeNonTestSymbol **1166** MissingFirm **1167** MissingAccount **1168** MissingMPID **1169** MissingRiskGroup **1170** ailyMarketOrderGrossNotionalExposureRiskRuleBreach **1171** DailyMarketOrderNetNotionalExposureRiskRuleBreach **1172** MissingDispMethodType **1173** MissingFirmRiskSetting **1174** InvalidAccountMPIDToFirm **1175** InvalidPegOffsetValue **1179** MissingSTPGroupId **1180** InvalidSTPGroupId **1181** InvalidLnkId **2001** UnknownOrder **2003** OrderAlreadyPendingCancelOrReplace **2006** DuplicateClOrdID **2018** InvalidPriceIncrement **2099** Cancel-Other **2100** MissingSymbol **2101** MissingLocate **2102** MissingClOrdID **2103** InvalidOrderQuantity **2104** InvalidSymbol **2105** InvalidLimitPrice

Field Name	Data Type	Description
rejectReason *(continued)*		**2107** SymbolHaltedOrPaused **2108** OrderSizeExceedsLimit **2109** ExceededMaxOrderNotionalAmt **2110** MissingOrigOrderIdentifiers **2111** AmbiguousOrigOrderIdentifiers **2112** OrigOrderSymbolNotMatchingRequestSymbol **2113** UnsupportedDisplayQuantityChange **2114** UnsupportedOrdTypeChange **2115** UnsupportedSideChange **2116** UnsupportedQuantityChange **2117** InvalidLocate **2118** ExchangeClosed **2119** BlockSessionRiskRuleViolated **2120** BlockSellShortRiskRuleViolated **2121** MaxSharesPerOrderRiskRuleBreach **2122** NoNBBOAvailable **2123** MaxNotionalValuePerOrderRiskRuleBreach **2124** MaxADVPercentPerOrderRiskRuleBreach **2125** PricePercentCollarRiskRuleViolated **2126** PriceValueCollarRiskRuleViolated **2127** HardToBorrowSecurityRiskRuleViolate **2128** InvalidSide **2129** InvalidOrdType **2130** InvalidClOrdID **2131** InvalidLnkId **3000** Other **3001** UnknownProduct **3002** UnknownSide **3003** UnknownGroupId **3004** HigherPriceLowerOrEqualToLowerPrice **3005** ProductMissingForPriceRestriction **3006** DuplicateClOrdID **3007** MalformedRequestMissingClOrdIdField **3008** InvalidCancelGroupId **3009** InvalidClOrdId **3010** InvalidLowerPrice **3011** InvalidHigherPrice **Allowed Values: MEMX** **1001** InvalidSymbol **1002** ExchangeClosed **1003** OrderExceedsLimit **1006** DuplicateOrder **1018** InvalidPriceIncrement **1019** NoNBBOAvailable **1020** NotionalValueExceedsThreshold **1022** BlockSellShortRiskRuleViolated **1023** HardToBorrowSecurityRiskRuleViolated **1027** MaxNotionalValuePerOrderRiskRuleViolated **1099** Other **1100** MissingSymbol **1101** MissingLocateReqd **1102** InvalidLocateReqd **1103** MissingClOrdId **1104** InvalidClOrdId **1105** MissingSide **1106** InvalidSide **1107** MissingOrderQty **1108** InvalidOrderQty **1109** MissingOrderType

Field Name	Data Type	Description
rejectReason *(continued)*		**1110** InvalidOrderType **1111** MissingTimeInForce **1112** InvalidTimeInForce **1113** MissingOrderCapacity **1114** InvalidCapacity **1115** MissingExecInst **1116** MissingLimitPrice **1117** InvalidLimitPrice **1118** MissingMaxFloor **1119** InvalidMaxFloor **1120** MissingReserveReplenishAmountType **1121** InvalidReserveReplenishAmountType **1122** MissingReserveReplenishTimeType **1123** InvalidReserveReplenishTimeType **1124** MissingRandomReplenishmentValue **1125** InvalidRandomReplenishmentValue **1126** InvalidRandomReplenishValueForReserveType **1127** MissingRepriceFrequencyType **1128** InvalidRepriceFrequencyType **1129** MissingRepriceBehaviorType **1130** InvalidRepriceBehaviorType **1131** InvalidRepriceBehaviorForRepriceFrequency **1132** MissingCustomerCapacityType **1133** InvalidCustomerCapacityType **1134** MissingExpireTime **1135** InvalidExpireTime **1136** MissingPegType **1137** InvalidPegType **1138** InvalidModifierForOrderType **1139** InvalidModifiersCombination **1140** InvalidTradingSessionForOrderType **1141** InvalidTimeInForceForOrderType **1142** InvalidModifierForPegType **1143** InvalidMinQty **1144** InvalidExecInst **1145** InvalidMpid **1146** SymbolHaltedOrPaused **1147** BlockISORiskRuleViolated **1148** BlockSessionRiskRuleViolated **1149** BlockNonTestSymbolsRiskRuleViolated **1150** MaxSharesPerOrderRiskRuleBreach **1151** PricePercentCollarRiskRuleViolated **1152** PriceValueCollarRiskRuleViolated **1153** MaxADVPercentPerOrderRiskRuleBreach **1154** DailyGrossNotionalExposureRiskRuleBreach **1155** DailyNetNotionalExposureRiskRuleBreach **1156** MaxNumDuplicateOrdersRiskRuleBreach **1157** MaxOrderRateRiskRuleBreach **1158** RestrictedSecurityRiskRuleViolated **1159** InvalidSelfTradePreventionConfiguration **1160** InvalidSelfTradePreventionType **1161** InvalidRiskGroupId **1162** FirmDisabled **1163** MPIDDisabled **1164** AccountDisabled **1165** CannotTradeNonTestSymbol **1166** MissingFirm **1167** MissingAccount **1168** MissingMPID **1169** MissingRiskGroup

Field Name	Data Type	Description
rejectReason *(continued)*		**1170** ailyMarketOrderGrossNotionalExposureRiskRuleBreach **1171** DailyMarketOrderNetNotionalExposureRiskRuleBreach **1172** MissingDispMethodType **1173** MissingFirmRiskSetting **1174** InvalidAccountMPIDToFirm **1175** InvalidPegOffsetValue **1179** MissingSTPGroupId **1180** InvalidSTPGroupId **1181** InvalidLnkId **2001** UnknownOrder **2003** OrderAlreadyPendingCancelOrReplace **2006** DuplicateClOrdID **2018** InvalidPriceIncrement **2099** Cancel-Other **2100** MissingSymbol **2101** MissingLocate **2102** MissingClOrdID **2103** InvalidOrderQuantity **2104** InvalidSymbol **2105** InvalidLimitPrice **2107** SymbolHaltedOrPaused **2108** OrderSizeExceedsLimit **2109** ExceededMaxOrderNotionalAmt **2110** MissingOrigOrderIdentifiers **2111** AmbiguousOrigOrderIdentifiers **2112** OrigOrderSymbolNotMatchingRequestSymbol **2113** UnsupportedDisplayQuantityChange **2114** UnsupportedOrdTypeChange **2115** UnsupportedSideChange **2116** UnsupportedQuantityChange **2117** InvalidLocate **2118** ExchangeClosed **2119** BlockSessionRiskRuleViolated **2120** BlockSellShortRiskRuleViolated **2121** MaxSharesPerOrderRiskRuleBreach **2122** NoNBBOAvailable **2123** MaxNotionalValuePerOrderRiskRuleBreach **2124** MaxADVPercentPerOrderRiskRuleBreach **2125** PricePercentCollarRiskRuleViolated **2126** PriceValueCollarRiskRuleViolated **2127** HardToBorrowSecurityRiskRuleViolate **2128** InvalidSide **2129** InvalidOrdType **2130** InvalidClOrdID **2131** InvalidLnkId **3000** Other **3001** UnknownProduct **3002** UnknownSide **3003** UnknownGroupId **3004** HigherPriceLowerOrEqualToLowerPrice **3005** ProductMissingForPriceRestriction **3006** DuplicateClOrdID **3007** MalformedRequestMissingClOrdIdField **3008** InvalidCancelGroupId **3009** InvalidClOrdId **3010** InvalidLowerPrice **3011** InvalidHigherPrice **Allowed Values: MEMXOP** **1001** UnknownSymbol

Field Name	Data Type	Description
rejectReason *(continued)*		**1002** ExchangeClosed **1003** DuplicateOrder **1016** PriceExceedsCurrentPriceBand **1018** InvalidPriceIncrement **1027** MaxNotionalValuePerOrderRiskRuleViolated **1099** Other **1100** MissingSymbol **1101** Invlaid Symbol **1102** MissingClOrdID **1103** InvalidClORdID **1104** MissingSide **1105** InvalidSide **1106** MissingOrderQty **1107** InvalidOrderQty **1108** MissingOrderType **1109** InvalidOrderType **1110** MissingTimeInForce **1111** InvalidTimeInForce **1112** MissingTradingCapacity **1113** InvalidTradingCapacity **1114** MIssingExecInst **1115** InvalidExecInst **1118** MissingLimitPrice **1119** InvalidLimitPrice **1120** MissingCustomerCapacity **1121** Invalid Custoemr Capacity **1124** MissingMatchTradePreventionType **1125** InvalidMatchTradePreventionType **1126** MissingCancelGroupID **1127** InvalidCancelGroupID **1128** MissingMTPGroupID **1129** InvalidMTPGroupID **1130** MissingLnkID **1131** InvalidLnkID **1134** MissingRiskGroupID **1135** InvalidRiskGroupID **1136** MissingEFID **1137** InvalidEFID **1138** MissingListSeqNo **1139** InvalidListSeqNo **1140** QuotesHaveDifferentUnderliers **1141** TwoSidedQuotesCross **1142** MissingPositionEffect **1143** InvalidPositionEffect **1144** MissingRepriceBehaviorType **1145** InvalidRepriceBehaviorType **1146** MissingRepriceFrequencyType **1147** InvalidRepriceFrequencyType **1148** MissingPartyRoleType **1149** InvalidPartyRoleType **1150** MissingPartyID **1151** InvalidPartyID **1152** MissingPartyIDSource **1153** InvalidPartyIDSource **1154** UnderlyingSymbolNotOpen **1155** MissingSendingTime **1156** InvalidSendingTime **1157** MissingSubAccount **1158** InvalidSubAccount **1159** MissingGiveUp

Field Name	Data Type	Description
rejectReason ***(continued)***		**1160** InvalidGiveUp **1161** MissingCMTA **1162** InvalidCMTA **1163** MissingActionableID **1164** InvalidActionableID **1165** MissingOptionalOCCData Added **1166** InvalidOptionalOCCData **1200** Halted **1201** FirmDisabled **1202** EFIDDisabled **1203** AccountDisabled **1204** InvalidModifierForOrderType **1205** InvalidTimeInForceForOrderType **1207** PostOnlyNotAllowed **1208** QuoteModifyRejected **1209** QuotingDisabled **1210** InvalidQuoteComponentCount **1300** NonTestSymbolsBlocked **1301** NotionalValueExceedsThreshold **1302** MaxNotionalValuePerOrderRiskRuleViolated **1303** BlockNonTestSymbolRiskRuleViolated **1304** MaxContractsPerOrderRiskRuleViolated **1305** NBBOWidthExceedsThreshold **1306** ExchangePriceValueCollar RiskRuleViolated **1307** ExecutionPriceHigherThanStrikePrice **1308** MarketBuyWhenNBOIsZero **1309** MarketSellWhenNBBGreaterThanThreshold **1310** QuotePriceOutsideExchangeThreshold **1311** QuoteModifyRejected **1312** NumContractsExecutedExceedsThresh old **1313** NotionalValueOfExecutionsExceedsThreshold **1314** CountOfExecutionsExceedsThreshold **1315** OutstandingPercentageThresholdExceeded **1316** TripsThresholdExceeded **1317** ISOOrdersNotAllowed **1318** MarketIsCrossed **1319** ActiveRiskBreach **1320** ManualRiskBreach **1321** GrossNotionalValueExceedsThreshold **1322** NetNotionalValueExceedsThreshold **1323** DuplicateOrderThresholdExceeded **1324** OrderRateThresholdExceeded **1325** MassCancelLockoutInEffect **1326** MarketOrderGrossNotionalValueExceedsThreshold **1327** MarketOrderNetNotionalValueExceedsThreshold **1328** LimitOrderFatFingerCheck **1329** BulkQuoteFatFingerCheck **2000** TooLateToCancel **2001** UnkownOrder **2003** OrderAlreadyPendingCancelOrReplace **2006** DuplicateClOrdID **2008** PriceExceedsCurrentPriceBand **2018** InvalidPriceIncrement **2099** Other **2100** MissingSymbol **2101** InvalidSymbol **2102** MissingClOrdID **2103** InvalidClOrdID **2104** MissingSide **2105** InvalidSide

Field Name	Data Type	Description
rejectReason *(continued)*		**2106** MissingOrderQty **2107** InvalidOrderQty **2108** MissingOrderType **2109** InvalidOrderType **2110** MissingLimitPrice **2111** InvalidLimitPrice **2112** MissingLnkID **2113** InvalidLnkID **2114** MissingListSeqNo **2115** InvalidListSeqNo **2116** MissingOrigClOrdID **2117** InvalidOrigClOrdID **2118** MissingOrderID **2119** InvalidOrderID **2120** MissingSendingTime **2121** InvalidSendingTime **2200** Halted **2201** FirmDisabled **2202** EFIDDisabled **2203** AccountDisabled **2204** UnsupportedOrdTypeChange **2205** UnsupportedSideChange **2206** SymbolMismatch **2207** OrigOrderIDMismatch **2300** NonTestSymbolsBlocked **2301** NotionalValueExceedsThreshold **2302** MaxNotionalValuePerOrderRiskRuleViolated **2303** BlockNonTestSymbolRiskRuleViolated **2304** MaxContractsPerOrderRiskRuleViolated **2305** NBBOWidthExceedsThreshold **2306** ExchangePriceValueCollar RiskRuleViolated **2307** ExecutionPriceHigherThanStrikePrice **2308** MarketBuyWhenNBOIsZero **2309** MarketSellWhenNBBGreaterThanThreshold **2310** QuotePriceOutsideExchangeThreshold **2312** NumContractsExecutedExceedsThreshold **2313** NotionalValueOfExecutionsExceedsThrreshold **2314** CountOfExecutionsExceedsThreshold **2315** OutstandingPercentageThresholdExceeded **2316** TripsThresholdExceeded **2317** ISOOrdersNotAllowed **2318** MarketIsCrossed **2319** ActiveRiskBreach **2320** ManualRiskBreach **2321** GrossNotionalValueExceedsThreshold **2322** NetNotionalValueExceedsThreshold **2323** DuplicateOrderThresholdExceeded **2324** OrderRateThresholdExceeded **2325** MassCancelLockoutInEffect **2326** MarketOrderGrossNotionalValueExceedsThreshold **2327** MarketOrderNetNotionalValueExceedsThreshold **2328** LimitOrderFatFingerCheck **2329** BulkQuoteFatFingerCheck **3000** Other **3001** UnknownProduct **3002** UnknownSide **3003** UnknownGroupId **3004** HigherPriceLowerOr EqualToLowerPrice **3005** ProductMissingFor PriceRestriction **3006** DuplicateClOrdID

Field Name	Data Type	Description
rejectReason ***(continued)***		**3007** MissingClOrdID **3008** InvalidCancelGroupId **3009** InvalidClOrdId **3010** InvalidLowerPrice **3011** InvalidHigherPrice **3012** InvalidOrigClOrdID **3013** InvalidCancelGroupID **3014** MissingMassCancelInst **3015** InvalidMassCancelInst **3017** MassCancelInProgress **3018** MissingSendingTime **3019** InvalidSendingTime **Allowed Values: MIAX Emerald Option** **EMLD_ERR_0001** InvalidRequest **EMLD_ERR_0002** UndefinedError **EMLD_ERR_0003** NotPermitted **EMLD_ERR_0004** InvalidAction **EMLD_ERR_0005** InvalidForMpid **EMLD_ERR_0006** InvalidEnumValue **EMLD_ERR_0007** InvalidState **EMLD_ERR_1000** InvalidUnderlying **EMLD_ERR_1001** InvalidProduct **EMLD_ERR_1002** InvalidTimeInForce **EMLD_ERR_1003** TimeInForceConflict **EMLD_ERR_1004** IncompatibleTifIso **EMLD_ERR_1005** InvalidSize **EMLD_ERR_1006** InvalidOpenClose **EMLD_ERR_1007** InvalidSide **EMLD_ERR_1008** InvalidPrice **EMLD_ERR_1009** InvalidOrdType **EMLD_ERR_1010** IncompatibleOrdTypeTIF **EMLD_ERR_1011** IncompatibleOrdTypeISO **EMLD_ERR_1012** InvalidAwayMarketRoutingPolicy **EMLD_ERR_1013** InvalidClearingAccount **EMLD_ERR_1014** InvalidMinSize **EMLD_ERR_1015** ManagedInterestOnContraOrderSide **EMLD_ERR_1016** InvalidQuoteType **EMLD_ERR_1017** InvalidOrderQty **EMLD_ERR_1018** InvalidParticipantType **EMLD_ERR_1019** InvalidCoveredOrUncovered **EMLD_ERR_1020** InvalidClOrdID **EMLD_ERR_1021** InvalidOrigClOrdID **EMLD_ERR_1022** InvalidWait **EMLD_ERR_1023** InvalidForAssignedMM **EMLD_ERR_1024** InvalidToChange **EMLD_ERR_1025** InvalidDFCStatus **EMLD_ERR_1026** NotInLOW **EMLD_ERR_1027** InvalidWhenRouting

Field Name	Data Type	Description	
rejectReason *(continued)*		**EMLD_ERR_1028**	InvalidOrderState
		EMLD_ERR_1029	DuplicateClOrdID
		EMLD_ERR_1030	DuplicateOrderID
		EMLD_ERR_1031	DuplicateClientMessageID
		EMLD_ERR_1032	InvalidTargetMessageID
		EMLD_ERR_1033	UnknownOrder
		EMLD_ERR_1034	UnknownMpid
		EMLD_ERR_1035	InvalidMpid
		EMLD_ERR_1036	InvalidFirmCode
		EMLD_ERR_1037	QuoteOutsideAcceptanceWindow
		EMLD_ERR_1038	NoSuchEvent
		EMLD_ERR_1039	RestrictedToClosing
		EMLD_ERR_1040	NonTradeable
		EMLD_ERR_1041	MMNotRegisteredForUnderlying
		EMLD_ERR_1042	InvalidTifAocWhenOptionIsNotInAuction
		EMLD_ERR_1043	InvalidTifOpgWhenOptionIsNotInOpening
		EMLD_ERR_1044	InvalidTiflocWhenOptionIsNotTrading
		EMLD_ERR_1045	InvalidTifFokWhenOptionIsNotTrading
		EMLD_ERR_1046	SystemClosedForTrading
		EMLD_ERR_1047	InvalidISOWhenOptionIsOpening
		EMLD_ERR_1048	NotPermittedPrice
		EMLD_ERR_1049	NotPermittedSide
		EMLD_ERR_1050	Accepted
		EMLD_ERR_1051	PreLQW
		EMLD_ERR_1052	InvalidDFC
		EMLD_ERR_1053	WrongCloud
		EMLD_ERR_1054	TooLateToCancel
		EMLD_ERR_1055	InvalidCancel
		EMLD_ERR_1056	MomTooWideForMarket
		EMLD_ERR_1057	MomLimitTooInferior
		EMLD_ERR_1058	InvalidOrderID
		EMLD_ERR_1059	InvalidMarketOrderForLuldUnderlying
		EMLD_ERR_1060	InvalidAttributableIndicator
		EMLD_ERR_1061	InvalidMvpTicks
		EMLD_ERR_1062	InvalidCancelOnDisconnect
		EMLD_ERR_1063	InvalidOrderClass
		EMLD_ERR_1064	InvalidPrimeRole
		EMLD_ERR_1065	InvalidPrimeStrategy
		EMLD_ERR_1066	InvalidNumContraItems
		EMLD_ERR_1067	InvalidPairedOriginValue
		EMLD_ERR_1068	InvalidAllocID
		EMLD_ERR_1069	InvalidAutoMatchOrdType
		EMLD_ERR_1070	InvalidAutoMatchPrice
		EMLD_ERR_1071	InvalidLastAction
		EMLD_ERR_1072	InvalidIndirectCancel
		EMLD_ERR_1073	WideMarket
		EMLD_ERR_1074	InvalidWhenPostOnlyOrdType

Field Name	Data Type	Description	
rejectReason *(continued)*		**EMLD_ERR_1075**	InvalidWhenPostOnlyTif
		EMLD_ERR_1100	PairedPrimeFunctionalityIsSuspended
		EMLD_ERR_1101	CustomerCrossPrimeFunctionalityIsSuspended
		EMLD_ERR_1102	OptionIsNotInFreeTrading
		EMLD_ERR_1103	OptionIsAboutToCloseTrading
		EMLD_ERR_1104	LockedOrCrossedNbbo
		EMLD_ERR_1105	CrossedNbbo
		EMLD_ERR_1106	AuctionPriceDoesntImproveNbboOnContraSide
		EMLD_ERR_1107	ManagedInterestOnAgencyOrderSide
		EMLD_ERR_1108	ManagedInterestOnAtLeastOneSideOfMbbo
		EMLD_ERR_1110	PriceLocksMbboWithPriorityCustomerInterest
		EMLD_ERR_1111	PriceLocksMbboWithOrderInterest
		EMLD_ERR_1112	CrossedSameMpid
		EMLD_ERR_1113	QualifiedContingentCrossPrimeFunctionalityIsSuspended
		EMLD_ERR_1114	SmallSizeWithPennyNbbo
		EMLD_ERR_1115	PriceIsWorseThanSameSideNbboForIsoPairedPrime
		EMLD_ERR_1116	PriceIsWorseThanSameSideNbboForIsoPairedPrime
		EMLD_ERR_1117	IsoPrimeIsDisabled
		EMLD_ERR_1119	PriceIsOutsideOfNbbo
		EMLD_ERR_2000	MassQuoteCancelInProgress
		EMLD_ERR_2001	UnderlyingHaltInProgress
		EMLD_ERR_2002	PAllQuotesCanceled
		EMLD_ERR_2003	Arm2MpidUnderlyingProtectionInEffect
		EMLD_ERR_2004	Arm2FirmProtectionInEffect
		EMLD_ERR_2005	OccKillSwitchProtectionInEffect
		EMLD_ERR_2006	NotRequested
		EMLD_ERR_3000	RiskProtectionInEffect
		EMLD_ERR_3001	InvalidAllowableEngagementPercentage
		EMLD_ERR_3002	InvalidCountingPeriod
		EMLD_ERR_3003	NoSuchArmSetting
		EMLD_ERR_3004	InvalidArmSettingSource
		EMLD_ERR_3005	InvalidArm2CountingPeriod
		EMLD_ERR_3006	InvalidArm2ThresholdCount
		EMLD_ERR_3007	NoSuchArm2Setting
		EMLD_ERR_3010	RpmBlockedMpid
		EMLD_ERR_3020	SspBlocked
		EMLD_ERR_3021	InvalidSspScope
		EMLD_ERR_3022	SspNotEnabledForMpid
		EMLD_ERR_4000	SystemStateMalformattedTime
		EMLD_ERR_4001	SystemStateTimeInPast
		EMLD_ERR_4500	ComplexInvalidStrategy
		EMLD_ERR_4501	ComplexTradingSuspendedForCloud
		EMLD_ERR_4502	ComplexOrderFeatureDisabledForUnderlying
		EMLD_ERR_4503	ComplexAppOrdersDisabledForUnderlying
		EMLD_ERR_4504	ComplexStrategyNotTradeable
		EMLD_ERR_4505	InvalidNumProductLegs
		EMLD_ERR_4506	InvalidComplexPriceProtection

Field Name	Data Type	Description	
rejectReason *(continued)*		**EMLD_ERR_4507**	InvalidAuctionOnArrival
		EMLD_ERR_4508	InvalidBookMatchOnly
		EMLD_ERR_4509	InvalidLegRefId
		EMLD_ERR_4510	ComplexVerticalSpreadPriceProtection
		EMLD_ERR_4511	ComplexCalendarSpreadPriceProtection
		EMLD_ERR_4512	OutsidePriceRangeForStrategy
		EMLD_ERR_4513	StrategyNotQuoteEligible
		EMLD_ERR_4514	CMomPricedThrough
		EMLD_ERR_4515	StrategyNotDefined
		EMLD_ERR_4516	ComplexMarketOrdersDisabledForUnderlying
		EMLD_ERR_4517	ComplexPrimeOrderFeatureDisabled
		EMLD_ERR_4518	ComplexC2cOrderFeatureDisabled
		EMLD_ERR_4519	ComplexQccOrderFeatureDisabled
		EMLD_ERR_4520	ComplexParityPriceProtection
		EMLD_ERR_4521	ComplexEnhancementsPhase1Disabled
		EMLD_ERR_4525	ComplexButterflySpreadPriceProtection
		EMLD_ERR_4526	DebitCreditProtection
		EMLD_ERR_5000	UnknownPurgeOriginator
		EMLD_ERR_5001	SpeedTestControl
		EMLD_ERR_5100	InvalidStockClearingAccount
		EMLD_ERR_5101	InvalidStockLegCapacityIndicator
		EMLD_ERR_5102	InvalidSellShortStockLeg
		EMLD_FCR_0000	TooLateToCancel
		EMLD_FCR_0001	UnknownOrder
		EMLD_FCR_0002	BrokerOption
		EMLD_FCR_0003	OrderAlreadyInPendingCancelOrPendingReplaceStatus
		EMLD_FOR_0000	BrokerOption
		EMLD_FOR_0001	UnknownSymbol
		EMLD_FOR_0002	ExchangeClosed
		EMLD_FOR_0003	OrderExceedsLimit
		EMLD_FOR_0004	TooLateToEnter
		EMLD_FOR_0005	UnknownOrder
		EMLD_FOR_0006	DuplicateOrder
		EMLD_FOR_0007	DuplicateOfAVerballyCommunicatedOrder
		EMLD_FOR_0008	StaleOrder
		EMLD_FOR_0011	UnsupportedOrderCharacteristic
		Allowed Values: MIAX MIAMI Option	
		MIAMI_ERR_0001	InvalidRequest
		MIAMI_ERR_0002	UndefinedError
		MIAMI_ERR_0003	NotPermitted
		MIAMI_ERR_0004	InvalidAction
		MIAMI_ERR_0005	InvalidForMpid
		MIAMI_ERR_0006	InvalidEnumValue
		MIAMI_ERR_0007	InvalidState
		MIAMI_ERR_1000	InvalidUnderlying
		MIAMI_ERR_1001	InvalidProduct

Field Name	Data Type	Description	
rejectReason *(continued)*		**MIAMI_ERR_1002**	InvalidTimeInForce
		MIAMI_ERR_1003	TimeInForceConflict
		MIAMI_ERR_1004	IncompatibleTifIso
		MIAMI_ERR_1005	InvalidSize
		MIAMI_ERR_1006	InvalidOpenClose
		MIAMI_ERR_1007	InvalidSide
		MIAMI_ERR_1008	InvalidPrice
		MIAMI_ERR_1009	InvalidOrdType
		MIAMI_ERR_1010	IncompatibleOrdTypeTIF
		MIAMI_ERR_1011	IncompatibleOrdTypeISO
		MIAMI_ERR_1012	InvalidAwayMarketRoutingPolicy
		MIAMI_ERR_1013	InvalidClearingAccount
		MIAMI_ERR_1014	InvalidMinSize
		MIAMI_ERR_1015	InvalidLiquidityType
		MIAMI_ERR_1016	InvalidQuoteType
		MIAMI_ERR_1017	InvalidOrderQty
		MIAMI_ERR_1018	InvalidParticipantType
		MIAMI_ERR_1019	InvalidCoveredOrUncovered
		MIAMI_ERR_1020	InvalidClOrdID
		MIAMI_ERR_1021	InvalidOrigClOrdID
		MIAMI_ERR_1022	InvalidWait
		MIAMI_ERR_1023	InvalidForAssignedMM
		MIAMI_ERR_1024	InvalidToChange
		MIAMI_ERR_1025	InvalidDFCStatus
		MIAMI_ERR_1026	NotInLOW
		MIAMI_ERR_1027	InvalidWhenRouting
		MIAMI_ERR_1028	InvalidOrderState
		MIAMI_ERR_1029	DuplicateClOrdID
		MIAMI_ERR_1030	DuplicateOrderID
		MIAMI_ERR_1031	DuplicateClientMessageID
		MIAMI_ERR_1032	InvalidTargetMessageID
		MIAMI_ERR_1033	UnknownOrder
		MIAMI_ERR_1034	UnknownMpid
		MIAMI_ERR_1035	InvalidMpid
		MIAMI_ERR_1036	InvalidFirmCode
		MIAMI_ERR_1037	QuoteOutsideAcceptanceWindow
		MIAMI_ERR_1038	NoSuchEvent
		MIAMI_ERR_1039	RestrictedToClosing
		MIAMI_ERR_1040	NonTradeable
		MIAMI_ERR_1041	MMNotRegisteredForUnderlying
		MIAMI_ERR_1042	InvalidTifAocWhenOptionIsNotInAuction
		MIAMI_ERR_1043	InvalidTifOpgWhenOptionIsNotInOpening
		MIAMI_ERR_1044	InvalidTiflocWhenOptionIsNotTrading
		MIAMI_ERR_1045	InvalidTifFokWhenOptionIsNotTrading
		MIAMI_ERR_1046	SystemClosedForTrading
		MIAMI_ERR_1047	InvalidISOWhenOptionIsOpening
		MIAMI_ERR_1048	NotPermittedPrice

Field Name	Data Type	Description	
rejectReason *(continued)*		**MIAMI_ERR_1049**	NotPermittedSide
		MIAMI_ERR_1050	Accepted
		MIAMI_ERR_1051	PreLQW
		MIAMI_ERR_1052	InvalidDFC
		MIAMI_ERR_1053	WrongCloud
		MIAMI_ERR_1054	TooLateToCancel
		MIAMI_ERR_1055	InvalidCancel
		MIAMI_ERR_1056	MomTooWideForMarket
		MIAMI_ERR_1057	MomLimitTooInferior
		MIAMI_ERR_1058	InvalidOrderID
		MIAMI_ERR_1059	InvalidMarketOrderForLuldUnderlying
		MIAMI_ERR_1060	InvalidAttributableIndicator
		MIAMI_ERR_1061	InvalidMvpTicks
		MIAMI_ERR_1062	InvalidCancelOnDisconnect
		MIAMI_ERR_1063	InvalidOrderClass
		MIAMI_ERR_1064	InvalidPrimeRole
		MIAMI_ERR_1065	InvalidPrimeStrategy
		MIAMI_ERR_1066	InvalidNumContraItems
		MIAMI_ERR_1067	InvalidPairedOriginValue
		MIAMI_ERR_1068	InvalidAllocID
		MIAMI_ERR_1069	InvalidAutoMatchOrdType
		MIAMI_ERR_1070	InvalidAutoMatchPrice
		MIAMI_ERR_1071	InvalidLastAction
		MIAMI_ERR_1072	InvalidIndirectCancel
		MIAMI_ERR_1073	WideMarket
		MIAMI_ERR_1080	InvalidTifSaoWhenOptionIsNotInSettlementAuction
		MIAMI_ERR_1100	PairedPrimeFunctionalityIsSuspended
		MIAMI_ERR_1101	CustomerCrossPrimeFunctionalityIsSuspended
		MIAMI_ERR_1102	OptionIsNotInFreeTrading
		MIAMI_ERR_1103	OptionIsAboutToCloseTrading
		MIAMI_ERR_1104	LockedOrCrossedNbbo
		MIAMI_ERR_1105	CrossedNbbo
		MIAMI_ERR_1106	AuctionPriceDoesntImproveNbboOnContraSide
		MIAMI_ERR_1107	ManagedInterestOnAgencyOrderSide
		MIAMI_ERR_1108	ManagedInterestOnAtLeastOneSideOfMbbo
		MIAMI_ERR_1110	PriceLocksMbboWithPriorityCustomerInterest
		MIAMI_ERR_1111	PriceLocksMbboWithOrderInterest
		MIAMI_ERR_1112	CrossedSameMpid
		MIAMI_ERR_1113	QualifiedContingentCrossPrimeFunctionalityIsSuspended
		MIAMI_ERR_1114	SmallSizeWithPennyNbbo
		MIAMI_ERR_1115	PriceIsWorseThanSameSideNbboForIsoPairedPrime
		MIAMI_ERR_1116	IsoPrimeIsDisabled
		MIAMI_ERR_1119	PriceIsOutsideOfNbbo
		MIAMI_ERR_2000	MassQuoteCancelInProgress
		MIAMI_ERR_2001	UnderlyingHaltInProgress
		MIAMI_ERR_2002	AllQuotesCanceled
		MIAMI_ERR_2003	Arm2MpidUnderlyingProtectionInEffect

Field Name	Data Type	Description	
rejectReason ***(continued)***		**MIAMI_ERR_2004**	Arm2FirmProtectionInEffect
		MIAMI_ERR_2005	OccKillSwitchProtectionInEffect
		MIAMI_ERR_2006	NotRequested
		MIAMI_ERR_3000	RiskProtectionInEffect
		MIAMI_ERR_3001	InvalidAllowableEngagementPercentage
		MIAMI_ERR_3002	InvalidCountingPeriod
		MIAMI_ERR_3003	NoSuchArmSetting
		MIAMI_ERR_3004	InvalidArmSettingSource
		MIAMI_ERR_3005	InvalidArm2CountingPeriod
		MIAMI_ERR_3006	InvalidArm2ThresholdCount
		MIAMI_ERR_3007	NoSuchArm2Setting
		MIAMI_ERR_3010	RpmBlockedMpid
		MIAMI_ERR_3020	SspBlocked
		MIAMI_ERR_3021	InvalidSspScope
		MIAMI_ERR_3022	SspNotEnabledForMpid
		MIAMI_ERR_4000	SystemStateMalformattedTime
		MIAMI_ERR_4001	SystemStateTimeInPast
		MIAMI_ERR_4500	ComplexInvalidStrategy
		MIAMI_ERR_4501	ComplexTradingSuspendedForCloud
		MIAMI_ERR_4502	ComplexOrderFeatureDisabledForUnderlying
		MIAMI_ERR_4503	ComplexAppOrdersDisabledForUnderlying
		MIAMI_ERR_4504	ComplexStrategyNotTradeable
		MIAMI_ERR_4505	InvalidNumProductLegs
		MIAMI_ERR_4506	InvalidComplexPriceProtection
		MIAMI_ERR_4507	InvalidAuctionOnArrival
		MIAMI_ERR_4508	InvalidBookMatchOnly
		MIAMI_ERR_4509	InvalidLegRefId
		MIAMI_ERR_4510	ComplexVerticalSpreadPriceProtection
		MIAMI_ERR_4511	ComplexCalendarSpreadPriceProtection
		MIAMI_ERR_4512	OutsidePriceRangeForStrategy
		MIAMI_ERR_4513	StrategyNotQuoteEligible
		MIAMI_ERR_4514	CMomPricedThrough
		MIAMI_ERR_4515	StrategyNotDefined
		MIAMI_ERR_4516	ComplexMarketOrdersDisabledForUnderlying
		MIAMI_ERR_4517	ComplexPrimeOrderFeatureDisabled
		MIAMI_ERR_4518	ComplexC2cOrderFeatureDisabled
		MIAMI_ERR_4519	ComplexQccOrderFeatureDisabled
		MIAMI_ERR_4520	ComplexParityPriceProtection
		MIAMI_ERR_4521	ComplexEnhancementsPhase1Disabled
		MIAMI_ERR_4522	RelatedFuturesCrossDisabled
		MIAMI_ERR_4523	RelatedFuturesCrossNotForProprietaryProduct
		MIAMI_ERR_4524	RelatedFuturesCrossForNonComboStrategy
		MIAMI_ERR_4525	ComplexButterflySpreadPriceProtection
		MIAMI_ERR_4526	DebitCreditProtection
		MIAMI_ERR_5000	UnknownPurgeOriginator
		MIAMI_ERR_5001	SpeedTestControl
		MIAMI_ERR_5100	InvalidStockClearingAccount

Field Name	Data Type	Description
rejectReason *(continued)*		**MIAMI_ERR_5101** InvalidStockLegCapacityIndicator **MIAMI_ERR_5102** InvalidSellShortStockLeg **MIAMI_ERR_5103** InvalidStockLegType **MIAMI_FCR_0000** TooLateToCancel **MIAMI_FCR_0001** UnknownOrder **MIAMI_FCR_0002** BrokerOption **MIAMI_FCR_0003** OrderAlreadyInPendingCancelOrPendingReplaceStatus **MIAMI_FOR_0000** BrokerOption **MIAMI_FOR_0001** UnknownSymbol **MIAMI_FOR_0002** ExchangeClosed **MIAMI_FOR_0003** OrderExceedsLimit **MIAMI_FOR_0004** TooLateToEnter **MIAMI_FOR_0005** UnknownOrder **MIAMI_FOR_0006** DuplicateOrder **MIAMI_FOR_0007** DuplicateOfAVerballyCommunicatedOrder **MIAMI_FOR_0008** StaleOrder **MIAMI_FOR_0011** UnsupportedOrderCharacteristic **Allowed Values: MIAX PEARL Equity** **PEARLEQ_COR_0000** Unknown **PEARLEQ_COR_0001** TooLateToCancel **PEARLEQ_COR_0002** UnknownOrder **PEARLEQ_COR_0003** BrokerOption **PEARLEQ_COR_0004** PreviousStillPending **PEARLEQ_COR_0005** RejectedByPrimaryListingMarket **PEARLEQ_CRD_0000** Unknown **PEARLEQ_CRD_0001** OrderNotFound **PEARLEQ_CRD_0002** NotReserveOrder **PEARLEQ_CRD_0003** SymbolNotHalted **PEARLEQ_FCR_0000** TooLateToCancel **PEARLEQ_FCR_0001** UnknownOrder **PEARLEQ_FCR_0002** BrokerOption **PEARLEQ_FCR_0003** OrderAlreadyInPendingCancelOrPendingReplaceStatus **PEARLEQ_FOR_0000** BrokerOption **PEARLEQ_FOR_0001** UnknownSymbol **PEARLEQ_FOR_0002** ExchangeClosed **PEARLEQ_FOR_0003** OrderExceedsLimit **PEARLEQ_FOR_0004** TooLateToEnter **PEARLEQ_FOR_0005** UnknownOrder **PEARLEQ_FOR_0006** DuplicateOrder **PEARLEQ_FOR_0007** DuplicateOfAVerballyCommunicatedOrder **PEARLEQ_FOR_0008** StaleOrder **PEARLEQ_FOR_0011** UnsupportedOrderCharacteristic **PEARLEQ_FRR_0000** Unknown **PEARLEQ_FRR_0001** UnknownSymbol **PEARLEQ_FRR_0002** ExchangeClosed **PEARLEQ_FRR_0003** OrderExceedsLimit

Field Name	Data Type	Description	
rejectReason ***(continued)***		**PEARLEQ_FRR_0004**	TooLateToEnter
		PEARLEQ_FRR_0005	DuplicateOrder
		PEARLEQ_FRR_0006	StaleOrder
		PEARLEQ_FRR_0007	UnsupportedOrderCharacteristic
		PEARLEQ_FRR_0008	BrokerOption
		PEARLEQ_FRR_0009	UnknownOrder
		PEARLEQ_FRR_0010	TooLateToCancel
		PEARLEQ_FRR_0011	CancelOrReplaceAlreadyPending
		PEARLEQ_FRR_0012	UnknownID
		PEARLEQ_FRR_0013	UnknownSecurity
		PEARLEQ_FRR_0014	UnsupportedMsgType
		PEARLEQ_FRR_0015	SystemNotAvailable
		PEARLEQ_FRR_0016	MatchingEngineNotAvailable
		PEARLEQ_FRR_0017	InvalidTag
		PEARLEQ_FRR_0018	TagNotDefinedForMessage
		PEARLEQ_FRR_0019	UndefinedTag
		PEARLEQ_FRR_0020	TagWithoutValue
		PEARLEQ_FRR_0021	ValueOutOfRange
		PEARLEQ_FRR_0022	IncorrectDataFormat
		PEARLEQ_FRR_0023	CompIDIssue
		PEARLEQ_FRR_0024	SendingTimeAccuracyIssue
		PEARLEQ_FRR_0025	InvalidMsgType
		PEARLEQ_FRR_0026	RequiredTagMissing
		PEARLEQ_FRR_0027	Other
		PEARLEQ_MRR_0000	Unknown
		PEARLEQ_MRR_0001	InvalidSymbol
		PEARLEQ_MRR_0002	CloudNotAvailable
		PEARLEQ_MRR_0003	InvalidMpid
		PEARLEQ_MRR_0004	NotPermittedMpid
		PEARLEQ_MRR_0005	InvalidClOrdId
		PEARLEQ_MRR_0006	InvalidTargetClOrdId
		PEARLEQ_MRR_0007	InvalidAccount
		PEARLEQ_MRR_0008	InvalidClearingAccount
		PEARLEQ_MRR_0009	RequestNotPermitted
		PEARLEQ_MRR_0010	MaxOrderSize
		PEARLEQ_MRR_0011	InvalidPrice
		PEARLEQ_MRR_0012	InvalidSize
		PEARLEQ_MRR_0013	ExceededMaxLimitNotionalValue
		PEARLEQ_MRR_0014	IsoOrdersNotAllowed
		PEARLEQ_MRR_0015	ShortSaleOrdersNotAllowed
		PEARLEQ_MRR_0016	DupOrderCheckRejected
		PEARLEQ_MRR_0017	CrmSessionBlocked
		PEARLEQ_MRR_0018	MfpSessionBlocked
		PEARLEQ_MRR_0019	InvalidSelfTradeProtectionGroup
		PEARLEQ_MRR_0020	ExceededTestSymbolThrottle
		PEARLEQ_MRR_0021	CrmNetNotionalSessionBlocked
		PEARLEQ_MRR_0022	InvalidCloud

Field Name	Data Type	Description	
rejectReason ***(continued)***		**PEARLEQ_MRR_0023**	MpidMaxOrderSize
		PEARLEQ_MRR_0024	MpidExceededMaxLimitNotionalValue
		PEARLEQ_MRR_0025	MpidIsoOrdersNotAllowed
		PEARLEQ_MRR_0026	MpidShortSaleOrdersNotAllowed
		PEARLEQ_MRR_0027	CrmGrossNotionalOpenOrderSessionBlocked
		PEARLEQ_MRR_0028	CrmNetNotionalOpenOrderSessionBlocked
		PEARLEQ_MRR_0029	SessionShortSaleExemptOrdersNotAllowed
		PEARLEQ_MRR_0030	MpidShortSaleExemptOrdersNotAllowed
		PEARLEQ_MRR_0031	MpidMarketOrderNotPermitted
		PEARLEQ_MRR_0032	SessionMarketOrderNotPermitted
		PEARLEQ_MRR_0033	MpidRestrictedSecurity
		PEARLEQ_MRR_0034	SessionRestrictedSecurity
		PEARLEQ_MRR_0035	RpmOrderRateSessionBlocked
		PEARLEQ_MRR_0036	ExceededSessionAdvThreshold
		PEARLEQ_MRR_0037	ExceededMpidAdvThreshold
		PEARLEQ_MRR_0038	MpidRestrictedShortSaleSecurity
		PEARLEQ_MRR_0039	SessionRestrictedShortSaleSecurity
		PEARLEQ_MRR_0040	MpidNonAgencyOrderNotAllowed
		PEARLEQ_MRR_0041	SessionNonAgencyOrderNotAllowed
		PEARLEQ_MRR_0042	BlockedBySessionOrderRateLimit
		PEARLEQ_MRR_0043	BlockedByMpidOrderRateLimit
		PEARLEQ_MRR_0044	BlockedByMpidSymbolOrderRateLimit
		PEARLEQ_MRR_0045	MassCancelInvalidPurgeGroup
		PEARLEQ_MRR_0046	InvalidLocateAccount
		PEARLEQ_MRR_0047	MpidConfigurationMissing
		PEARLEQ_MRR_0052	InvalidPriceIncrement
		PEARLEQ_MRR_0053	InvalidOrderID
		PEARLEQ_MRR_0054	NotPermittedLocateAccountSession
		PEARLEQ_MRR_0055	SessionBlockedHelpDesk
		PEARLEQ_MRR_0056	SessionBlockedMFPUI
		PEARLEQ_MRR_0057	SessionBlockedMFPAPI
		PEARLEQ_ORR_0000	Unknown
		PEARLEQ_ORR_0001	InvalidMpid
		PEARLEQ_ORR_0002	InvalidSymbol
		PEARLEQ_ORR_0003	DuplicateClOrdId
		PEARLEQ_ORR_0004	InvalidClOrdId
		PEARLEQ_ORR_0005	UnknownOrigClOrdId
		PEARLEQ_ORR_0006	InvalidOrigClOrdId
		PEARLEQ_ORR_0007	InvalidSize
		PEARLEQ_ORR_0008	NotInLow
		PEARLEQ_ORR_0009	UnexpectedError
		PEARLEQ_ORR_0014	InvalidCancelReasonValue
		PEARLEQ_ORR_0015	InvalidPriceValue
		PEARLEQ_ORR_0016	InvalidPriceIncrement
		PEARLEQ_ORR_0019	OrderMpidMismatch
		PEARLEQ_ORR_0021	DuplicateOrderId
		PEARLEQ_ORR_0022	InvalidOrderType

Field Name	Data Type	Description	
rejectReason *(continued)*		**PEARLEQ_ORR_0023**	InvalidRequestOrigin
		PEARLEQ_ORR_0024	InvalidSellType
		PEARLEQ_ORR_0025	CancelAlreadyPending
		PEARLEQ_ORR_0026	InvalidWhenRouting
		PEARLEQ_ORR_0027	ReplaceAlreadyPending
		PEARLEQ_ORR_0028	InvalidOrderCapacity
		PEARLEQ_ORR_0029	InvalidTimeInForce
		PEARLEQ_ORR_0030	InvalidRoutingInstruction
		PEARLEQ_ORR_0031	InvalidSelfTradeProtectionLevel
		PEARLEQ_ORR_0032	InvalidSelfTradeProtectionInstruction
		PEARLEQ_ORR_0033	InvalidPriceSlidingAndRePriceFrequency
		PEARLEQ_ORR_0034	InvalidUsePriceSlidingAndRePriceFrequency
		PEARLEQ_ORR_0035	InvalidUsePostOnly
		PEARLEQ_ORR_0036	InvalidUseISO
		PEARLEQ_ORR_0038	InvalidUseDisplayed
		PEARLEQ_ORR_0040	InvalidAvailableWhenLocked
		PEARLEQ_ORR_0041	InvalidUseAvailableWhenLocked
		PEARLEQ_ORR_0042	MarketOrderPriceProtectionTriggered
		PEARLEQ_ORR_0043	InvalidRoutingStrategy
		PEARLEQ_ORR_0044	InvalidUseRoutingStrategy
		PEARLEQ_ORR_0045	InvalidUseRoutingInstruction
		PEARLEQ_ORR_0046	InvalidAttributable
		PEARLEQ_ORR_0048	InvalidUseLocateRequired
		PEARLEQ_ORR_0049	InvalidSelfTradeProtectionGroup
		PEARLEQ_ORR_0050	InvalidUseSelfTradeProtectionGroup
		PEARLEQ_ORR_0051	InvalidUseSelfTradeProtectionInstruction
		PEARLEQ_ORR_0052	InvalidUseMarketOrderPriceProtection
		PEARLEQ_ORR_0053	InvalidUseMarketOrderTradingCollarCustomValue
		PEARLEQ_ORR_0054	InvalidUseLimitOrderPriceProtection
		PEARLEQ_ORR_0055	LimitOrderPriceProtectionTriggered
		PEARLEQ_ORR_0056	InvalidForCurrentSymbolTradingStatus
		PEARLEQ_ORR_0057	IpoDayPrimaryExchangeNotOpenYet
		PEARLEQ_ORR_0058	InvalidUseMinQty
		PEARLEQ_ORR_0059	InvalidChangeToMinQty
		PEARLEQ_ORR_0060	InvalidChangeMaxFloorQty
		PEARLEQ_ORR_0061	InvalidMaxFloorQty
		PEARLEQ_ORR_0062	InvalidDisplayRangeQty
		PEARLEQ_ORR_0063	InvalidUseOrderType
		PEARLEQ_ORR_0064	FeatureDisabled
		PEARLEQ_ORR_0065	InvalidUseAttributable
		PEARLEQ_ORR_0066	InvalidPACPrimaryListingMarket
		PEARLEQ_ORR_0067	TooLateForPacOrder
		PEARLEQ_ORR_0068	PacOrdersNotAllowedWhileBlacklisted
		PEARLEQ_ORR_0069	UnknownOrderId
		PEARLEQ_ORR_0070	CancelByOrderIdNotAllowed
		PEARLEQ_ORR_0071	PacBlacklistCancelNotApplicable
		PEARLEQ_ORR_0072	ExceededMaxNotionalValue

Field Name	Data Type	Description
rejectReason ***(continued)***		**PEARLEQ_ORR_0073** LimitPriceMoreAggressiveThanMarketImpactCollar **PEARLEQ_ORR_0074** InvalidPurgeGroup **PEARLEQ_ORR_0075** InvalidPegOffset **PEARLEQ_ORR_0078** InvalidUseCancelIfNotNbboSetter **PEARLEQ_ORR_0079** SessionTypeMismatch **PEARLEQ_ORR_0080** NotPermittedLocateAccountMpid **PEARLEQ_ORR_0081** SessionBlockedByHelpDesk **PEARLEQ_ORR_0082** SessionBlockedByMFPUI **PEARLEQ_ORR_0083** SessionBlockedByMFPAPI **PEARLEQ_ORR_0084** MpidOrFirmBlockedByHelpDesk **PEARLEQ_ORR_0085** MpidOrFirmBlockedByMFPUI **PEARLEQ_ORR_0086** MpidOrFirmBlockedByMFPAPI **PEARLEQ_ORR_0087** PurgeGroupMpidOrFirmBlockedByHelpDesk **PEARLEQ_ORR_0088** PurgeGroupMpidOrFirmBlockedByMFPUI **PEARLEQ_ORR_0089** PurgeGroupMpidOrFirmBlockedByMFPAPI **PEARLEQ_ORR_0100** BlockedByCrmTradeGrossNotionalFirm **PEARLEQ_ORR_0101** BlockedByCrmTradeGrossNotionalMpid **PEARLEQ_ORR_0102** BlockedByCrmTradeGrossNotionalSession **PEARLEQ_ORR_0103** BlockedByHelpDeskOrMfpMpidOrFirm **PEARLEQ_ORR_0104** BlockedByHelpDeskOrMfpSession **PEARLEQ_ORR_0105** BlockedByUserPurgePort **PEARLEQ_ORR_0106** BlockedByCrmTradeNetNotionalFirm **PEARLEQ_ORR_0107** BlockedByCrmTradeNetNotionalMpid **PEARLEQ_ORR_0108** BlockedByCrmTradeNetNotionalSession **PEARLEQ_ORR_0109** BlockedByCrmOpenOrderGrossNotionalFirm **PEARLEQ_ORR_0110** BlockedByCrmOpenOrderGrossNotionalMpid **PEARLEQ_ORR_0111** BlockedByCrmOpenOrderGrossNotionalSession **PEARLEQ_ORR_0112** BlockedByCrmOpenOrderNetNotionalFirm **PEARLEQ_ORR_0113** BlockedByCrmOpenOrderNetNotionalMpid **PEARLEQ_ORR_0114** BlockedByCrmOpenOrderNetNotionalSession **PEARLEQ_ORR_0115** BlockedByRpmOrderRateFirm **PEARLEQ_ORR_0116** BlockedByRpmOrderRateMpid **PEARLEQ_ORR_0117** BlockedByRpmOrderRateSession **PEARLEQ_ORR_0118** PurgeGroupBlockedByHelpDeskOrMfpMpidOrFirm **PEARLEQ_ORR_0119** PurgeGroupBlockedByUserPurgePort **PEARLEQ_ORR_0120** BlockedByCrmTradeAndOpenOrderGrossNotionalFirm **PEARLEQ_ORR_0121** BlockedByCrmTradeAndOpenOrderGrossNotionalMpid **PEARLEQ_ORR_0122** BlockedByCrmTradeAndOpenOrderGross NotionalSession **PEARLEQ_ORR_0123** BlockedByCrmTradeAndOpenOrderNetNotionalFirm **PEARLEQ_ORR_0124** BlockedByCrmTradeAndOpenOrderNetNotionalMpid **PEARLEQ_ORR_0125** BlockedByCrmTradeAndOpenOrderNetNotionalSession **Allowed Values: MIAMI PEARL Option** **PEARL_ERR_0000** Accepted **PEARL_ERR_0001** InvalidRequest **PEARL_ERR_0002** UndefinedError **PEARL_ERR_0003** NotPermitted

Field Name	Data Type	Description	
rejectReason *(continued)*		**PEARL_ERR_0004**	InvalidAction
		PEARL_ERR_0005	InvalidForMpid
		PEARL_ERR_0007	InvalidState
		PEARL_ERR_0008	InvalidOptionKind
		PEARL_ERR_0009	InvalidBulkBinaryOrderItemRequestType
		PEARL_ERR_1000	InvalidUnderlying
		PEARL_ERR_1001	InvalidProduct
		PEARL_ERR_1002	InvalidTimeInForce
		PEARL_ERR_1004	IncompatibleTifIso
		PEARL_ERR_1006	InvalidOpenClose
		PEARL_ERR_1007	InvalidSide
		PEARL_ERR_1008	InvalidPrice
		PEARL_ERR_1009	InvalidOrdType
		PEARL_ERR_1011	IncompatibleOrdTypeISO
		PEARL_ERR_1012	InvalidOrderHandlingInstruction
		PEARL_ERR_1013	InvalidClearingAccount
		PEARL_ERR_1014	InvalidAccount
		PEARL_ERR_1017	InvalidOrderQty
		PEARL_ERR_1018	InvalidParticipantType
		PEARL_ERR_1019	InvalidCoveredOrUncovered
		PEARL_ERR_1020	InvalidClOrdID
		PEARL_ERR_1021	InvalidOrigClOrdID
		PEARL_ERR_1024	InvalidToChange
		PEARL_ERR_1025	InvalidLastAction
		PEARL_ERR_1026	NotInLOW
		PEARL_ERR_1027	InvalidWhenRouting
		PEARL_ERR_1028	InvalidOrderState
		PEARL_ERR_1029	DuplicateClOrdID
		PEARL_ERR_1030	DuplicateOrderID
		PEARL_ERR_1033	UnknownOrder
		PEARL_ERR_1034	UnknownMpid
		PEARL_ERR_1035	InvalidMpid
		PEARL_ERR_1036	InvalidFirmCode
		PEARL_ERR_1039	RestrictedToClosing
		PEARL_ERR_1040	NonTradeable
		PEARL_ERR_1044	InvalidTiflocWhenOptionIsNotTrading
		PEARL_ERR_1046	SystemClosedForTrading
		PEARL_ERR_1047	InvalidISOWhenOptionIsNot Trading
		PEARL_ERR_1051	WrongCloud
		PEARL_ERR_1053	InvalidCancel
		PEARL_ERR_1054	IncompatibleTifPostOnly
		PEARL_ERR_1056	InvalidAutoReplace
		PEARL_ERR_1057	IncompatibleOrdTypePostOnly
		PEARL_ERR_1058	DuplicateClientOrderId
		PEARL_ERR_1059	InvalidTargetClientOrderId
		PEARL_ERR_1060	InvalidClientOrderId
		PEARL_ERR_1061	AutoReplaceNothingToCancel
		PEARL_ERR_1062	InvalidCmta
		PEARL_ERR_1063	SspBlocked
		PEARL_ERR_1064	SspNotEnabledForMpid
		PEARL_ERR_1070	MomTooWideForMarket
		PEARL_ERR_1071	MomTooWideForSellMarketZeroBid
		PEARL_ERR_1072	MomBuyLimitTooAggressive
		PEARL_ERR_1073	MomSellLimitTooAggressive
		PEARL_ERR_1074	MomMaxOpenContracts
		PEARL_ERR_1075	MomMaxOpenOrders
		PEARL_ERR_1076	InvalidOrderID
		PEARL_ERR_1077	InvalidMarketOrderForLuldUnderlying
		PEARL_ERR_1078	InvalidMvpTicks
		PEARL_ERR_1079	InvalidCancelOnDisconnect

Field Name	Data Type	Description
rejectReason *(continued)*		**PEARL_ERR_1080** InvalidAdditionalText **PEARL_ERR_1081** MaxPutPriceViolation **PEARL_ERR_2001** SpeedTestControl **PEARL_ERR_3000** RiskProtectionInEffect **PEARL_ERR_3001** InvalidAllowableEngagementPercentage **PEARL_ERR_3002** InvalidCountingPeriod **PEARL_ERR_3003** NoSuchArmSetting **PEARL_ERR_3004** InvalidArmSettingSource **PEARL_ERR_3005** InvalidArm2CountingPeriod **PEARL_ERR_3006** InvalidArm2ThresholdCount **PEARL_ERR_3007** NoSuchArm2Setting **PEARL_ERR_3010** RpmBlockedMpid **PEARL_ERR_3012** AllBinaryOrdersCanceled **PEARL_ERR_3013** Arm2MpidUnderlyingProtectionInEffect **PEARL_ERR_3014** Arm2FirmProtectionInEffect **PEARL_ERR_3015** OccKillSwitchProtectionInEffect **PEARL_ERR_4000** SystemStateMalformattedTime **PEARL_ERR_4001** SystemStateTimeInPast **PEARL_FOR_0000** BrokerOption **PEARL_FOR_0001** UnknownSymbol **PEARL_FOR_0002** ExchangeClosed **PEARL_FOR_0003** OrderExceedsLimit **PEARL_FOR_0004** TooLateToEnter **PEARL_FOR_0005** UnknownOrder **PEARL_FOR_0006** DuplicateOrder **PEARL_FOR_0007** DuplicateOfAVerballyCommunicatedOrder **PEARL_FOR_0008** StaleOrder **PEARL_FOR_0011** UnsupportedOrderCharacteristic **PEARL_FCR_0000** TooLateToCancel **PEARL_FCR_0001** UnknownOrder **PEARL_FCR_0002** BrokerOption **PEARL_FCR_0003** OrderAlreadyInPendingCancelOrPendingReplaceStatus **Allowed Values: MIAX SPHR** **SPHR_ERR_0000** Accepted **SPHR_ERR_0001** InvalidRequest **SPHR_ERR_0002** UndefinedError **SPHR_ERR_0003** NotPermitted **SPHR_ERR_0004** InvalidAction **SPHR_ERR_0005** InvalidForMpid **SPHR_ERR_0007** InvalidState **SPHR_ERR_0008** InvalidOptionKind **SPHR_ERR_0009** InvalidBulkBinaryOrderItemRequestType **SPHR_ERR_1000** InvalidUnderlying **SPHR_ERR_1001** InvalidProduct **SPHR_ERR_1002** InvalidTimeInForce **SPHR_ERR_1004** IncompatibleTifIso **SPHR_ERR_1005** InvalidSize **SPHR_ERR_1006** InvalidOpenClose **SPHR_ERR_1007** InvalidSide **SPHR_ERR_1008** InvalidPrice **SPHR_ERR_1009** InvalidOrdType **SPHR_ERR_1011** IncompatibleOrdTypeISO **SPHR_ERR_1012** InvalidOrderHandlingInstruction **SPHR_ERR_1013** InvalidClearingAccount **SPHR_ERR_1014** InvalidAccount **SPHR_ERR_1017** InvalidOrderQty **SPHR_ERR_1018** InvalidParticipantType **SPHR_ERR_1019** InvalidCoveredOrUncovered **SPHR_ERR_1020** InvalidClOrdID

Field Name	Data Type	Description	
rejectReason ***(continued)***		**SPHR_ERR_1021**	InvalidOrigClOrdID
		SPHR_ERR_1024	InvalidToChange
		SPHR_ERR_1025	InvalidLastAction
		SPHR_ERR_1026	NotInLOW
		SPHR_ERR_1027	InvalidWhenRouting
		SPHR_ERR_1028	InvalidOrderState
		SPHR_ERR_1029	DuplicateClOrdID
		SPHR_ERR_1030	DuplicateOrderID
		SPHR_ERR_1033	UnknownOrder
		SPHR_ERR_1034	UnknownMpid
		SPHR_ERR_1035	InvalidMpid
		SPHR_ERR_1036	InvalidFirmCode
		SPHR_ERR_1039	RestrictedToClosing
		SPHR_ERR_1040	NonTradeable
		SPHR_ERR_1044	InvalidTiflocWhenOptionIsNotTrading
		SPHR_ERR_1046	SystemClosedForTrading
		SPHR_ERR_1047	InvalidISOWhenOptionIsNotTrading
		SPHR_ERR_1048	InvalidFixOrderClass
		SPHR_ERR_1051	WrongCloud
		SPHR_ERR_1053	InvalidCancel
		SPHR_ERR_1054	WideMarket
		SPHR_ERR_1055	IncompatibleTifPostOnly
		SPHR_ERR_1056	InvalidAutoReplace
		SPHR_ERR_1057	IncompatibleOrdTypePostOnly
		SPHR_ERR_1058	DuplicateClientOrderId
		SPHR_ERR_1059	InvalidTargetClientOrderId
		SPHR_ERR_1060	InvalidClientOrderId
		SPHR_ERR_1061	AutoReplaceNothingToCancel
		SPHR_ERR_1062	InvalidCmta
		SPHR_ERR_1063	SspBlocked
		SPHR_ERR_1064	SspNotEnabledForMpid
		SPHR_ERR_1065	InvalidIndirectCancel
		SPHR_ERR_1066	InvalidSspScope
		SPHR_ERR_1067	InvalidMinSize
		SPHR_ERR_1068	InvalidNumContraItems
		SPHR_ERR_1069	InvalidAllocID
		SPHR_ERR_1070	MomTooWideForMarket
		SPHR_ERR_1071	MomTooWideForSellMarketZeroBid
		SPHR_ERR_1072	MomBuyLimitTooAggressive
		SPHR_ERR_1073	MomSellLimitTooAggressive
		SPHR_ERR_1074	MomMaxOpenContracts
		SPHR_ERR_1075	MomMaxOpenOrders
		SPHR_ERR_1076	InvalidOrderID
		SPHR_ERR_1077	InvalidMarketOrderForLuldUnderlying
		SPHR_ERR_1078	InvalidMvpTicks
		SPHR_ERR_1079	InvalidCancelOnDisconnect
		SPHR_ERR_1080	InvalidAdditionalText
		SPHR_ERR_1081	MaxPutPriceViolation
		SPHR_ERR_2001	SpeedTestControl
		SPHR_ERR_2102	OptionIsNotInFreeTrading
		SPHR_ERR_2105	CrossedNbbo
		SPHR_ERR_2108	ManagedInterestOnAtLeastOneSideOfMbbo
		SPHR_ERR_2110	PriceLocksMbboWithPriorityCustomerInterest
		SPHR_ERR_2119	PriceIsOutsideOfNbbo
		SPHR_ERR_3000	RiskProtectionInEffect
		SPHR_ERR_3001	InvalidAllowableEngagementPercentage
		SPHR_ERR_3002	InvalidCountingPeriod
		SPHR_ERR_3003	NoSuchArmSetting
		SPHR_ERR_3004	InvalidArmSettingSource
		SPHR_ERR_3005	InvalidArm2CountingPeriod

Field Name	Data Type	Description
rejectReason *(continued)*		**SPHR_ERR_3006** InvalidArm2ThresholdCount **SPHR_ERR_3007** NoSuchArm2Setting **SPHR_ERR_3010** RpmBlockedMpid **SPHR_ERR_3012** AllBinaryOrdersCanceled **SPHR_ERR_3013** Arm2MpidUnderlyingProtectionInEffect **SPHR_ERR_3014** Arm2FirmProtectionInEffect **SPHR_ERR_3015** OccKillSwitchProtectionInEffect **SPHR_ERR_4000** SystemStateMalformattedTime **SPHR_ERR_4001** SystemStateTimeInPast **SPHR_ERR_4500** ComplexInvalidStrategy **SPHR_ERR_4502** ComplexOrderFeatureDisabledForUnderlying **SPHR_ERR_4504** ComplexStrategyNotTradeable **SPHR_ERR_4505** InvalidNumProductLegs **SPHR_ERR_4506** InvalidComplexPriceProtection **SPHR_ERR_4509** InvalidLegRefId **SPHR_ERR_4510** ComplexVerticalSpreadPriceProtection **SPHR_ERR_4511** ComplexCalendarSpreadPriceProtection **SPHR_ERR_4512** OutsidePriceRangeForStrategy **SPHR_ERR_4514** CMomPricedThrough **SPHR_ERR_4515** StrategyNotDefined **SPHR_ERR_4520** ComplexParityPriceProtection **SPHR_ERR_4525** ComplexButterflySpreadPriceProtection **SPHR_ERR_4526** DebitCreditProtection **SPHR_ERR_4527** ComplexBinaryOrderInvalidTif **SPHR_ERR_5100** InvalidStockClearingAccount **SPHR_ERR_5101** InvalidStockLegCapacityIndicator **SPHR_ERR_5102** InvalidSellShortStockLeg **SPHR_ERR_5103** InvalidStockLegType **SPHR_FOR_0000** BrokerOption **SPHR_FOR_0001** UnknownSymbol **SPHR_FOR_0002** ExchangeClosed **SPHR_FOR_0003** OrderExceedsLimit **SPHR_FOR_0004** TooLateToEnter **SPHR_FOR_0005** UnknownOrder **SPHR_FOR_0006** DuplicateOrder **SPHR_FOR_0007** DuplicateOfAVerballyCommunicatedOrder **SPHR_FOR_0008** StaleOrder **SPHR_FOR_0011** UnsupportedOrderCharacteristic **SPHR_FCR_0000** TooLateToCancel **SPHR_FCR_0001** UnknownOrder **SPHR_FCR_0002** BrokerOption **SPHR_FCR_0003** OrderAlreadyInPendingCancelOrPendingReplaceStatus **Allowed Values: NASDAQ Equities** **0** none **1** quoteUnavailable **2** destinationClosed **3** invalidDisplay **4** invalidMaxFloor **5** invalidPegType **6** fatFinger **7** halted **8** isoNotAllowed **9** invalidSide **10** processingError **11** cancelPending **12** firmNotAuthorized **13** invalidMinQuantity **14** noClosingRefPrice

Field Name	Data Type	Description
rejectReason ***(continued)***		**15** other **16** cancelNotAllowed **17** peggingNotAllowed **18** crossedMarket **19** invalidQuantity **20** invalidCrossOrder **21** replaceNotAllowed **22** routingNotAllowed **23** invalidSymbol **24** test **25** lateLOCTooAggresive **26** retailNotAllowed **27** invalidMPPO **28** invalidDestination **29** invalidPrice **30** sharesExceedThreshold **31** maxNotionalExceeded **32** aggExposureExceeded **33** marketImpact **34** riskRestrictedStock **35** riskShortSellRestricted **36** riskOrderTypeRestricted **37** riskExceedsADV **38** riskFatFinger **39** riskLocateRequired **40** riskSymbolMessageRates **41** riskMessageRates **42** riskDuplicateRates **43** orderDead **44** clearingOutsideTimes **45** prmRejectAll **46** prmEasyBorrow **47** prmSymbolRestricted **48** prmIsoRestricted **49** prmOddLotRestricted **50** prmMidpointRestricted **51** prmPreMktRestricted **52** prmPostMktRestricted **53** prmShortSaleRestricted **54** prmOnOpenRestricted **55** prmOnCloseRestricted **56** prmTwoSidedQuoteRequired **57** prmShares **58** prmValue **59** overExposed **60** invalidMidpointPostOnlyPrice **61** UNKNOWN **62** INVALID_PARAMETERS **63** BAD_QUOTE **64** DEST_NOT_AVAILABLE **65** POSS_DUPE_ORDER **66** BAD_QUOTE_2 **67** RISK_EXPOSURE_OPEN **68** FIRM_LOCKED **69** RISK_EXPOSURE_EXEC **70** RISK_EXPOSURE_NOTIONAL **71** FIXORDER_BROKER_OPTION **72** FIXORDER_UNKNOWN_SYMBOL **73** FIXORDER_EXCHANGE_CLOSED **74** FIXORDER_EXCEEDS_LIMIT

Field Name	Data Type	Description
rejectReason *(continued)*		**75** FIXORDER_TOO_LATE **76** FIXORDER_UNKNOWN **77** FIXORDER_DUPLICATE **78** FIXORDER_DUPLICATE_VERBAL **79** FIXORDER_STALE **80** FIXCANCEL_TOO_LATE **81** FIXEDITING_FATAL_ERROR **Allowed Values: NASDAQ – PHLX, NOM** **1** AUTOPURGE **2** POD **3** FIRM **4** REASSIGN **5** HALT **6** AIQ **7** MANUPURGE **8** OPENPURGE **9** REPRICE **10** SUSPEND **11** LIQUIDITY TAKER **12** RAPID FIRE VOL **13** ZAP DELETE **14** KILLSWITCH AUTO **15** KILLSWITCH CMD LINE **16** KILLSWITCH TRADEINFO **17** notPermitted **18** badStopPrice **19** systemClosed **20** invalidType **21** invalidClearing **22** halt **23** invalidTime **24** invalidCross **25** invalidMpid **26** invalidMinSize **27** alreadyOpened **28** restrictedSymbol **29** closeCross **30** invalidSymbol **31** testmode **32** tiedToStockNotAllowed **33** invalidSize **34** limitTooDeep **35** systemError **36** invalidAttribute **37** suspend **38** notFreeTrading **39** nbboTooWide **40** changeContractsNoOrder **41** changeContractsInvalid **42** reentry **43** killswitch_reentry **44** postOnlyReprice **45** undLULD **46** invalidPreOpenIoc **47** userCancel **48** ioc **49** timeout **50** unsolictedOutReentry

Field Name	Data Type	Description
rejectReason *(continued)*		**51** routeRequest **52** staleOrder **53** sppLimit **54** auctionInProgress **55** engineCancel **56** tooLateToAct **57** noAuction **58** invalidTIF **59** aonNotAllowed **60** bboCross **61** purge **62** orderExpired **63** aiq **64** cnbboLimit **65** noBbo **66** mktOrder **67** treasuryOptionsNotAllowed **68** openingCancel **69** executionNotPossible **70** badCapacity **71** optionNotOpen **72** openDelay **73** liquidityTaker **74** killSwitch **75** adminCancel **76** systemCancel **77** brokerOption **78** invalidCrossSurrender **79** cod **80** eodCancel **85** actionNotAllowed **86** CXLDOWN **87** doNotUse **88** featureNotSupported **89** halted **90** instrumentClosed **91** instrumentState **92** invalidALO **93** invalidAuctionType **94** invalidCapacity **95** invalidDisclosureMask **96** invalidDisplay **97** invalidDntt **98** invalidFirm **99** invalidFlashInst **100** invalidISO **101** invalidInstrType **102** invalidKillAction **103** invalidMsgType **104** invalidOpenClose **105** invalidOrderType **106** invalidPersist **107** invalidPrice **108** invalidProduct **109** invalidReserveInfo **110** invalidScope **111** invalidShortSaleInd **112** invalidSide **113** MASS_CANCEL **114** none

Field Name	Data Type	Description
rejectReason *(continued)*		**115** orderNotFound **116** prefNotAllowed **117** missingClearingAccount **118** invalidStrategy **119** undReentry **120** invalidSelfReplenishVolume **OTHER** OTHER **Allowed Values: NASDAQ – ISE, GEMX** *(GEMX - Legacy active until 11/30/2023)* **1** AUTOPURGE **2** POD **3** FIRM **4** REASSIGN **5** AIQ **6** MANUPURGE **7** OPENPURGE **8** REPRICE **9** LIQUIDITY TAKER **10** RAPID FIRE VOL **11** ZAP DELETE **12** KILLSWITCH AUTO **13** KILLSWITCH CMD LINE **14** KILLSWITCH TRADEINFO **15** KILLSWITCH USER **16** notPermitted **17** invalidStopPrice **18** invalidDisplay **19** invalidType **20** invalidFirm **21** invalidClearing **22** halt **23** invalidTime **24** invalidCross **25** invalidMpid **26** invalidMinSize **27** alreadyOpened **28** restrictedSymbol **29** closeCross **30** invalidSymbol **31** tiedToStockNotAllowed **32** invalidSize **33** systemError **34** invalidAttribute **35** nbboTooWide **36** changeContractsNoOrder **37** changeContractsInvalid **38** reentry **39** killswitchReentry **40** undLULD **41** ioc **42** timeout **43** unsolictedOutReentry **44** routeRequest **45** sppLimit **46** invalidTIF **47** bboCross **48** cnbboLimit **49** noBbo **50** mktOrder

Field Name	Data Type	Description
rejectReason *(continued)*		**51** treasuryOptionNotAllowed **52** openingCancel **53** executionNotPossible **54** invalidCapacity **55** optionNotOpen **56** killswitchPurge **57** systemCancel **58** brokerOption **59** invalidSide **60** invalidSpread **61** invalidAuctionType **62** invalidFormat **63** frozen **64** requestPending **65** cancelUp **66** cancelDown **67** postOnlyTaker **68** invalidState **69** tooManyAuctions **70** invalidAuctionParams **71** rejectedReplace **72** massCancel **73** invalidReprice **74** price **75** size **76** nbboLimit **77** impliedExec **78** tooManyImplieds **79** complexInstrExists **80** exceededMaxComplexInstr **81** firmExceededMaxComplexInstr **82** invalidPtaContracts **83** invalidMatchId **84** invalidTradeId **85** invalidCrossId **86** invalidClientId **87** dnttNotAllowed **88** instrumentClosed **89** atrLimitReached **90** invalidISO **91** invalidStepupPrice **92** threeTickLimitReached **93** pending **94** pennyNbboRestriction **95** invalidDntt **96** invalidInstrType **97** invalidOrderType **98** invalidALO **99** invalidFlashInst **100** invalidPrefParty **101** invalidReserveInfo **102** invalidPersist **103** invalidShortSaleInd **104** invalidProduct **105** invalidScope **106** invalidOpenClose **107** invalidToken **108** invalidKillAction **109** invalidLegCount **110** invalidLegType

Field Name	Data Type	Description
rejectReason *(continued)*		**111** invalidLegRatio **112** invalidCrossType **113** prefNotAllowed **114** orderNotFound **115** actionNotAllowed **116** instrumentState **117** qccNotAllowed **118** qccWithStockNetPriceNotAllowed **119** qccWithMultiOptLegNotAllowed **120** invalidDestination **121** maxRoutesAttempted **122** destinationNotAvailable **123** minQtyNotSatisfied **124** sorRespTimeout **125** invalidAllocSplits **126** qccWithStockPriceNotAllowed **127** tooManyStockTradeAttempts **128** notTob **129** cod **130** poolExhausted **131** eodCancel **132** CLOSEPURGE **133** PRICE_LIMIT **134** ORDER_SIZE **161** none **162** ADMIN_CANCEL **163** BAD_STOP_PRICE **164** SYSTEM_CLOSED **165** CANCEL_ON_DISCONNECT **166** INVALID_MAX_FLOOR **167** HALTED **168** PROCESS_ERROR **169** KILL_SWITCH **170** FIRM_NOT_AUTHORIZED **171** STALE_ORDER **172** INVALID_ROUTING_INST **173** MARKET_IS_OPEN **174** RESTRICTED_SYMBOL **175** INVALID_SYMBOL **176** TEST_MODE **177** USER_CANCEL **178** INVALID_PRICE **179** TIED_TO_STOCK_INVALID **180** THRESHOLD_EXCEEDED **181** SPP_LIMIT **182** AUCTION_IN_PROGRESS **183** LIMIT_TOO_DEEP **184** RESERVED1 **185** RESERVED2 **186** FEATURE_NOT_SUPPORTED **187** SYSTEM_ERROR **188** ENGINE_CANCEL **189** TOO_LATE_TO_ACT **190** NO_AUCTION **191** INVALID_ATTRIBUTE **192** DO_NOT_USE **193** INVALID_TIF **194** AON_NOT_ALLOWED **195** SYSTEM_CLOSED2 **196** NBBO_CROSSED

Field Name	Data Type	Description
rejectReason *(continued)*		**197** PURGE **198** INVALID_PRICE2 **199** NOT_FREE_TRADING **200** NBBO_TOO_WIDE **201** REENTRY_NO_ORDER **202** REENTRY_SYSTEM_ERROR **203** REENTRY_REQUIRED **204** ORDER_EXPIRED **205** AIQ_CANCEL **206** LIQUIDITY_TAKER **207** POST_ONLY_REPRICE **208** LULD **209** SUSPEND **210** OPEN_DELAY **211** INVALID_PREOPEN_IOC **212** unAuthorizedGiveup **213** INVALID_PREOPEN_IOC **214** invalidTriggerId **215** invalidAccount **216** invalidAccountNoKill **217** invalidAccountFirm **218** beforeGtc **219** afterNothingDone **220** invalidRoutingStrategy **221** invalidTargetFirm **222** time **223** minReserveOrderNotFullfilled **224** closingCancel **225** portRateBreached **226** invalidTraderId **227** stopOrderMissingPreviousTradePrice **228** stopPriceOnlyAllowedForStopOrder **229** firmSuspended **230** traderSuspended **231** portSuspended **232** invalidInvestmentDecision **233** invalidExecutionDecision **234** invalidDea **235** invalidPartyRoleQualifier **236** instrumentExpired **237** invalidBrokerPct **238** invalidExecutionSourceCode **239** prmGroupBlocked **240** prmLimitsMissing **241** prmGroupProductBlocked **242** prmMaxOrderVolume **243** prmMaxOrderValue **244** maxOrderVolume **245** maxOrderValue **246** invalidPrmGroup **247** prmProductOpenOrderVol **248** prmProductOpenDelta **249** prmProductOpenVega **250** prmProductTradedVol **251** prmProductTradedDelta **252** prmProductTradedVega **253** prmProductTotalVol **254** prmProductTotalDelta **255** prmProductTotalVega **256** firmExceededMaxQuoteRequest

Field Name	Data Type	Description
rejectReason *(continued)*		**257** circuitBreaker
		258 quoteRequestInProgress
		259 invalidEvent
		260 invalidMatchEventId
		261 rfaReentry
		262 invalidRfaInstruction
		263 rfaInstructionWithRfaId
		264 invalidPrmLimit
		265 invalidPrmActionBlock
		266 prmGroupUnblocked
		267 prmProductUnblocked
		268 free_10001
		269 orej_system_error
		270 orej_duplicate_order_id
		271 orej_invalid_time_for_acceptance
		272 orej_not_open_for_trading
		273 orej_unacceptable_volume
		274 orej_invalid_auction_response_attribute
		275 orej_limit_too_far_below_bid
		276 orej_limit_too_far_above_ask
		277 orej_giveup_override_not_allowed
		278 orej_aon_replace_not_allowed
		279 orej_opg_after_opening
		280 orej_off_floor_acct_not_allowed
		281 orej_invalid_volume
		282 orej_mkt_is_invalid
		283 orej_fok_is_invalid
		284 orej_auction_response_not_allowed
		285 orej_post_only_reprice
		286 free_10019
		287 free_10020
		288 free_10021
		289 orej_invalid_limit_price
		290 orej_invalid_stop_price
		291 orej_buy_stop_lteq_bid
		292 orej_sell_stop_gteq_ask
		293 free_10026
		294 orej_mm_must_be_limit
		295 orej_firm_must_be_limit
		296 orej_bd_must_be_limit
		297 free_10030
		298 orej_aon_not_allowed_for_mm
		299 orej_aon_not_allowed_for_firm
		300 orej_aon_not_allowed_for_bd
		301 free_10034
		302 free_10035
		303 free_10036
		304 free_10037
		305 free_10038
		306 orej_missing_account_id
		307 free_10040
		308 free_10041
		309 orej_restricted_option
		310 orej_invalid_open_close
		311 orej_mm_only
		312 orej_must_be_straight_cancel
		313 orej_target_not_found
		314 orej_target_cancel_pending
		315 orej_target_filled
		316 orej_target_cancelled

Field Name	Data Type	Description
rejectReason ***(continued)***		**317** free_10050 **318** orej_target_not_open **319** free_10052 **320** orej_cancel_buy_sell_mismatch **321** orej_cancel_symbol_mismatch **322** orej_repl_symbol_mismatch **323** orej_cancel_volume_mismatch **324** orej_cancel_price_mismatch **325** orej_cancel_origin_mismatch **326** orej_cancel_mm_mismatch **327** free_10060 **328** free_10061 **329** free_10062 **330** orej_cancel_bad_leaves_volume **331** free_10064 **332** orej_missing_mm_badge **333** free_10066 **334** free_10067 **335** orej_mm_badge_not_allowed **336** free_10069 **337** orej_broker_option **338** orej_stale_order **339** orej_listed_routing_only **340** orej_in_trading_halt **341** free_10074 **342** free_10075 **343** orej_unknown_clearing_firm **344** orej_mar_too_many_routes **345** orej_mar_duplicate_order **346** orej_mar_exch_direct_not_allowed **347** orej_mar_exch_direct_cust_only **348** orej_luld **349** orej_suspend **350** orej_killswitch **351** orej_liquidity_taker **352** free_10085 **353** free_10086 **354** free_10087 **355** free_10088 **356** orej_tltc **357** free_10090 **358** orej_purge **359** free_10092 **360** orej_aiq **361** orej_reentry_required **362** orej_nbbo_too_wide **363** orej_invalid_msg_type **364** orej_required_tag_missing **365** free_10098 **366** free_10099 **367** free_10100 **368** orej_invalid_firm **369** orej_invalid_cross_surrender **370** orej_invalid_br_seqno **371** orej_invalid_side **372** orej_invalid_kind **373** orej_off_floor_req_exch **374** orej_off_floor_req_multacc **375** orej_invalid_multacc **376** orej_off_floor_req_multiacc

Field Name	Data Type	Description
rejectReason ***(continued)***		**377** orej_invalid_strike_price **378** orej_invalid_order_type **379** orej_invalid_cust_firm **380** free_10113 **381** orej_invalid_send_time **382** orej_invalid_tif **383** free_10116 **384** orej_invalid_aon **385** orej_iso_aon_is_invalid **386** orej_opg_co_not_allowed **387** orej_opg_iso_not_allowed **388** orej_invalid_qualifier **389** free_10122 **390** orej_invalid_orig_mkt **391** orej_invalid_option_symbol **392** orej_cancel_cmta_mismatch **393** orej_cancel_supp_mismatch **394** orej_cancel_crosstype_mismatch **395** orej_cancel_openclose_mismatch **396** orej_cancel_execbroker_mismatch **397** orej_cancel_fbnum_mismatch **398** orej_supp_id_too_long **399** orej_invalid_mm_badge **400** free_10133 **401** free_10134 **402** free_10135 **403** free_10136 **404** free_10137 **405** free_10138 **406** free_10139 **407** free_10140 **408** orej_invalid_strategy **409** orej_invalid_leg_ratio **410** orej_duplicate_leg_ref_id **411** orej_invalid_num_legs **412** orej_dup_leg_symbol **413** orej_invalid_non_conforming_ratio **414** orej_price_violates_spp_limit **415** orej_feature_not_supported **416** free_10149 **417** orej_open_delay **418** orej_preopen_ioc **419** orej_iso_must_be_limit **420** orej_invalid_security_type **421** free_10154 **422** orej_invalid_cl_order_id **423** orej_invalid_orig_cl_order_id **424** orej_invalid_ifi **425** orej_invalid_exec_inst **426** orej_invalid_route_inst **427** orej_iso_opg_is_invalid **428** orej_poss_dup **429** free_10162 **430** free_10163 **431** orej_invalid_exp **432** orej_invalid_leg_ref_id **433** orej_cancel_clearing_mismatch **434** orej_iso_not_allowed **435** orej_invalid_handling_inst **436** orej_opg_stop_limit_not_allowed

Field Name	Data Type	Description
rejectReason *(continued)*		**437** orej_auction_eligibility_mismatch **438** orej_cannot_change_stop_class **439** orej_exp_day_invalid **440** orej_invalid_prin_agency **441** orej_invalid_stock_leg **442** orej_auction_in_progress **443** orej_invalid_nwt_price **444** orej_invalid_auction_id **445** orej_invalid_cross_specs **446** orej_straight_cxl_not_allowed **447** orej_cxl_replace_not_allowed **448** orej_invalid_num_orders **449** orej_order_ids_same **450** orej_must_improve_price **451** orej_msg_too_late_to_process **452** orej_no_auction **453** orej_nbbo_crossed **454** orej_attribute_mismatch **455** orej_symbol_not_open **456** orej_exch_direct_must_be_limit **457** orej_invalid_max_floor **458** orej_invalid_min_quantity **459** orej_invalid_underlying **460** orej_invalid_risk_request **461** orej_wait_iso_not_allowed **462** orej_opg_aon_not_allowed **463** orej_buy_market_order **464** orej_bbo_invalid **465** free_10198 **466** orej_reserve_not_allowed **467** orej_postonly_not_allowed **468** orej_invalid_floor_brk **469** orej_invalid_priv_ref **470** orej_invalid_effective_time **471** orej_invalid_good_til_date **472** orej_invalid_cross_client_order_id **473** orej_invalid_num_sides **474** orej_invalid_display_when **475** orej_invalid_price_prot_scope **476** orej_invalid_auction_inst **477** orej_invalid_stepup_price **478** orej_invalid_stepup_price_type **479** orej_invalid_spec_order_type **480** orej_invalid_exposure **481** orej_invalid_broker_pct **482** orej_invalid_price_delta **483** orej_must_be_limit **484** orej_must_be_routable **485** orej_must_persist **486** orej_must_be_aon **487** orej_opg_stop_not_allowed **488** orej_reserve_modification_invalid **489** orej_invalid_entitlement_req_id **490** orej_invalid_no_party_entitlements **491** orej_invalid_list_update_action **492** orej_invalid_no_party_details **493** orej_invalid_party_detail_id **494** orej_invalid_party_detail_role **495** orej_invalid_id_source **496** orej_invalid_security_id

Field Name	Data Type	Description
rejectReason *(continued)*		**497** orej_invalid_alloc_id **498** orej_invalid_alloc_trans_type **499** orej_invalid_trade_date **500** orej_invalid_no_allocs **501** orej_invalid_alloc_shares **502** orej_invalid_no_execs **503** orej_invalid_exec_id **504** orej_exec_broker_required **505** orej_invalid_shares **506** orej_invalid_display_range **507** orej_postonly_replace **508** orej_invalid_maturity_date **509** orej_invalid_security_exchange **510** orej_too_many_auctions **511** orej_mar_cust_limit_qty **512** orej_mar_cust_limit_notional **513** orej_mar_cust_limit_agg_qty **514** orej_mar_cust_limit_agg_notional **515** orej_invalid_match_id **516** orej_invalid_pta_account **517** orej_invalid_pta_contracts **518** orej_invalid_client_id **519** orej_preferencing_not_allowed **520** orej_invalid_stock_leg_giveup **521** orej_invalid_contra_side_short_sell **522** orej_pta_not_allowed **523** orej_qcc_invalid_stock_ratio **524** orej_cancel_strategy_mismatch **525** orej_destination_not_available **526** orej_invalid_underlying_price **527** orej_invalid_underlying_qty **528** orej_invalid_rfp_id **529** orej_invalid_root_parties **530** away_status_New **531** away_status_PartiallyFilled **532** away_status_Filled **533** away_status_Done **534** away_status_Canceled **535** away_status_Replaced **536** away_status_PendingCancel **537** away_status_Stopped **538** away_status_Rejected **539** away_status_Suspended **540** away_status_PendingNew **541** away_status_Calculated **542** away_status_Expired **543** away_status_Accepted **544** away_status_PendingReplace **545** away_status_Restated **546** away_status_Trade **547** away_status_TradeCancel **548** away_status_TradeCorrect **549** alloc_status_Accepted **550** alloc_status_BlockLevelReject **551** alloc_status_PartialAccept **552** alloc_status_NotYetProcessed **553** invalidTimeOfAgreement **554** invalidTradeReportId **555** invalidTradeReportRefId **556** invalidAgencyCross

Field Name	Data Type	Description
rejectReason *(continued)*		**557** invalidHandlingInstr **558** invalidEqualLeg **559** invalidMinBlockTradeSize **560** invalidDeferralThreshold **561** invalidTradePublishIndicator **562** invalidMaximumTradeReportSize **563** invalidTradeType **564** flexInstrExists **565** invalidCircuitBreakerId **566** invalidPriceProtectionTableCode **567** invalidStrikePrice **568** invalidExpirationDate **OTHER** OTHER **Allowed Values: NASDAQ – NOBO, MRX, GEMX** **1** AUTOPURGE **2** POD **3** FIRM **4** REASSIGN **5** HALT **6** AIQ **7** MANUPURGE **8** OPENPURGE **9** REPRICE **10** SUSPEND **11** LIQUIDITY TAKER **12** RAPID FIRE VOL **13** ZAP DELETE **14** KILLSWITCH AUTO **15** KILLSWITCH CMD LINE **16** KILLSWITCH TRADEINFO **1017** KILLSWITCH USER **1018** notPermitted **1019** badStopPrice **1020** systemClosed **1021** invalidDisplay **1022** invalidType **1023** invalidFirm **1024** invalidClearing **1025** halt **1026** invalidTime **1027** invalidCross **1028** invalidMpid **1029** invalidMinSize **1030** alreadyOpened **1031** restrictedSymbol **1032** closeCross **1033** invalidSymbol **1034** testmode **1035** invalidPrice **1036** tiedToStockNotAllowed **1037** invalidSize **1038** limitTooDeep **1039** featureNotSupported **1040** systemError **1041** invalidAttribute **1042** suspend **1043** notFreeTrading **1044** nbboTooWide **1045** changeContractsNoOrder

Field Name	Data Type	Description
rejectReason *(continued)*		**1046** changeContractsInvalid **1047** reentry **1048** killswitch_reentry **1049** postOnlyReprice **1050** undLULD **1051** invalidPreOpenIoc **1052** userCancel **1053** ioc **1054** timeout **1055** unsolictedOutReentry **1056** routeRequest **1057** staleOrder **1058** sppLimit **1059** auctionInProgress **1060** engineCancel **1061** tooLateToAct **1062** noAuction **1063** invalidTIF **1064** aonNotAllowed **1065** bboCross **1066** purge **1067** orderExpired **1068** aiq **1069** cnbboLimit **1070** noBbo **1071** mktOrder **1072** treasuryOptionsNotAllowed **1073** openingCancel **1074** executionNotPossible **1075** invalidCapacity **1076** optionNotOpen **1077** openDelay **1078** liquidityTaker **1079** killSwitchPurge **1080** adminCancel **1081** systemCancel **1082** brokerOption **1083** invalidSide **1084** invalidSpread **1085** invalidAuctionType **1086** invalidFormat **1087** frozen **1088** requestPending **1089** cancelUp **1090** cancelDown **1091** postOnlyTaker **1092** invalidState **1093** tooManyAuctions **1094** invalidAuctionParams **1095** rejectedReplace **1096** massCancel **1097** invalidReprice **1098** price **1099** size **1100** nbboLimit **1101** impliedExec **1102** tooManyImplieds **1103** complexInstrExists **1104** exceededMaxComplexInstr **1105** firmExceededMaxComplexInstr

Field Name	Data Type	Description
rejectReason *(continued)*		**1106** invalidPtaContracts **1107** invalidMatchId **1108** invalidTradeId **1109** invalidCrossId **1110** invalidClientId **1111** dnttNotAllowed **1112** instrumentClosed **1113** atrLimitReached **1114** invalidISO **1115** invalidStepupPrice **1116** threeTickLimitReached **1117** pending **1118** pennyNbboRestriction **1119** invalidDntt **1120** invalidInstrType **1121** invalidOrderType **1122** invalidALO **1123** invalidFlashInst **1124** invalidPrefParty **1125** invalidReserveInfo **1126** invalidPersist **1127** invalidShortSaleInd **1128** invalidProduct **1129** invalidScope **1130** invalidOpenClose **1131** invalidToken **1132** invalidKillAction **1133** invalidLegCount **1134** invalidLegType **1135** invalidLegRatio **1136** invalidCrossType **1137** prefNotAllowed **1138** orderNotFound **1139** actionNotAllowed **1140** instrumentState **1141** qccNotAllowed **1142** qccWithStockNetPriceNotAllowed **1143** qccWithMultiOptLegNotAllowed **1144** invalidDestination **1145** maxRoutesAttempted **1146** destinationNotAvailable **1147** minQtyNotSatisfied **1148** sorRespTimeout **1149** invalidAllocSplits **1150** qccWithStockPriceNotAllowed **1151** tooManyStockTradeAttempts **1152** notTob **1153** cod **1154** poolExhaused **1155** eodCancel **1156** unAuthorizedGiveup **1157** invalidTriggerId **1158** invalidAccount **1159** invalidAccountNoKill **1160** invalidAccountFirm **1161** beforeGtc **1162** afterNothingDone **1163** invalidRoutingStrategy **1164** invalidTargetFirm **1165** time

Field Name	Data Type	Description
rejectReason *(continued)*		**1166** minReserveOrderNotFullfilled **1167** closingCancel **1168** portRateBreached **1169** invalidTraderId **1170** stopOrderMissingPreviousTradePrice **1171** stopPriceOnlyAllowedForStopOrder **1172** firmSuspended **1173** traderSuspended **1174** portSuspended **1175** invalidInvestmentDecision **1176** invalidExecutionDecision **1177** invalidDea **1178** invalidPartyRoleQualifier **1179** instrumentExpired **1180** invalidBrokerPct **1181** invalidExecutionSourceCode **1182** prmGroupBlocked **1183** prmLimitsMissing **1184** prmGroupProductBlocked **1185** prmMaxOrderVolume **1186** prmMaxOrderValue **1187** maxOrderVolume **1188** maxOrderValue **1189** invalidPrmGroup **1190** prmProductOpenOrderVol **1191** prmProductOpenDelta **1192** prmProductOpenVega **1193** prmProductTradedVol **1194** prmProductTradedDelta **1195** prmProductTradedVega **1196** prmProductTotalVol **1197** prmProductTotalDelta **1198** prmProductTotalVega **1199** firmExceededMaxQuoteRequest **1200** circuitBreaker **1201** quoteRequestInProgress **1202** invalidEvent **1203** invalidMatchEventId **1204** rfaReentry **1205** invalidRfaInstruction **1206** rfaInstructionWithRfaId **1207** tobRepriced **1208** invalidPrmLimit **1209** invalidPrmActionBlock **1210** prmGroupUnblocked **1211** prmProductUnblocked **1212** missingClearingAccount **1213** free_10001 **1214** orej_system_error **1215** orej_duplicate_order_id **1216** orej_invalid_time_for_acceptance **1217** orej_not_open_for_trading **1218** orej_unacceptable_volume **1219** orej_invalid_auction_response_attribute **1220** orej_limit_too_far_below_bid **1221** orej_limit_too_far_above_ask **1222** orej_giveup_override_not_allowed **1223** orej_aon_replace_not_allowed **1224** orej_opg_after_opening **1225** orej_off_floor_acct_not_allowed

Field Name	Data Type	Description
rejectReason *(continued)*		**1226** orej_invalid_volume **1227** orej_mkt_is_invalid **1228** orej_fok_is_invalid **1229** orej_auction_response_not_allowed **1230** orej_post_only_reprice **1231** free_10019 **1232** free_10020 **1233** free_10021 **1234** orej_invalid_limit_price **1235** orej_invalid_stop_price **1236** orej_buy_stop_lteq_bid **1237** orej_sell_stop_gteq_ask **1238** free_10026 **1239** orej_mm_must_be_limit **1240** orej_firm_must_be_limit **1241** orej_bd_must_be_limit **1242** free_10030 **1243** orej_aon_not_allowed_for_mm **1244** orej_aon_not_allowed_for_firm **1245** orej_aon_not_allowed_for_bd **1246** free_10034 **1247** free_10035 **1248** free_10036 **1249** free_10037 **1250** free_10038 **1251** orej_missing_account_id **1252** free_10040 **1253** free_10041 **1254** orej_restricted_option **1255** orej_invalid_open_close **1256** orej_mm_only **1257** orej_must_be_straight_cancel **1258** orej_target_not_found **1259** orej_target_cancel_pending **1260** orej_target_filled **1261** orej_target_cancelled **1262** free_10050 **1263** orej_target_not_open **1264** free_10052 **1265** orej_cancel_buy_sell_mismatch **1266** orej_cancel_symbol_mismatch **1267** orej_repl_symbol_mismatch **1268** orej_cancel_volume_mismatch **1269** orej_cancel_price_mismatch **1270** orej_cancel_origin_mismatch **1271** orej_cancel_mm_mismatch **1272** free_10060 **1273** free_10061 **1274** free_10062 **1275** orej_cancel_bad_leaves_volume **1276** free_10064 **1277** orej_missing_mm_badge **1278** free_10066 **1279** free_10067 **1280** orej_mm_badge_not_allowed **1281** free_10069 **1282** orej_broker_option **1283** orej_stale_order **1284** orej_listed_routing_only **1285** orej_in_trading_halt

Field Name	Data Type	Description
rejectReason *(continued)*		**1286** free_10074
		1287 free_10075
		1288 orej_unknown_clearing_firm
		1289 orej_mar_too_many_routes
		1290 orej_mar_duplicate_order
		1291 orej_mar_exch_direct_not_allowed
		1292 orej_mar_exch_direct_cust_only
		1293 orej_luld
		1294 orej_suspend
		1295 orej_killswitch
		1296 orej_liquidity_taker
		1297 free_10085
		1298 free_10086
		1299 free_10087
		1300 free_10088
		1301 orej_tltc
		1302 free_10090
		1303 orej_purge
		1304 free_10092
		1305 orej_aiq
		1306 orej_reentry_required
		1307 orej_nbbo_too_wide
		1308 orej_invalid_msg_type
		1309 orej_required_tag_missing
		1310 free_10098
		1311 free_10099
		1312 free_10100
		1313 orej_invalid_firm
		1314 orej_invalid_cross_surrender
		1315 orej_invalid_br_seqno
		1316 orej_invalid_side
		1317 orej_invalid_kind
		1318 orej_off_floor_req_exch
		1319 orej_off_floor_req_multacc
		1320 orej_invalid_multacc
		1321 orej_off_floor_req_multiacc
		1322 orej_invalid_strike_price
		1323 orej_invalid_order_type
		1324 orej_invalid_cust_firm
		1325 free_10113
		1326 orej_invalid_send_time
		1327 orej_invalid_tif
		1328 free_10116
		1329 orej_invalid_aon
		1330 orej_iso_aon_is_invalid
		1331 orej_opg_co_not_allowed
		1332 orej_opg_iso_not_allowed
		1333 orej_invalid_qualifier
		1334 free_10122
		1335 orej_invalid_orig_mkt
		1336 orej_invalid_option_symbol
		1337 orej_cancel_cmta_mismatch
		1338 orej_cancel_supp_mismatch
		1339 orej_cancel_crosstype_mismatch
		1340 orej_cancel_openclose_mismatch
		1341 orej_cancel_execbroker_mismatch
		1342 orej_cancel_fbnum_mismatch
		1343 orej_supp_id_too_long
		1344 orej_invalid_mm_badge
		1345 free_10133

Field Name	Data Type	Description
rejectReason *(continued)*		**1346** free_10134 **1347** free_10135 **1348** free_10136 **1349** free_10137 **1350** free_10138 **1351** free_10139 **1352** free_10140 **1353** orej_invalid_strategy **1354** orej_invalid_leg_ratio **1355** orej_duplicate_leg_ref_id **1356** orej_invalid_num_legs **1357** free_10145 **1358** orej_invalid_non_conforming_ratio **1359** orej_price_violates_spp_limit **1360** orej_feature_not_supported **1361** free_10149 **1362** orej_open_delay **1363** orej_preopen_ioc **1364** orej_iso_must_be_limit **1365** orej_invalid_security_type **1366** free_10154 **1367** orej_invalid_cl_order_id **1368** orej_invalid_orig_cl_order_id **1369** orej_invalid_ifi **1370** orej_invalid_exec_inst **1371** orej_invalid_route_inst **1372** orej_iso_opg_is_invalid **1373** orej_poss_dup **1374** free_10162 **1375** free_10163 **1376** orej_invalid_exp **1377** orej_invalid_leg_ref_id **1378** orej_cancel_clearing_mismatch **1379** orej_iso_not_allowed **1380** orej_invalid_handling_inst **1381** orej_opg_stop_limit_not_allowed **1382** orej_auction_eligibility_mismatch **1383** orej_cannot_change_stop_class **1384** orej_exp_day_invalid **1385** orej_invalid_prin_agency **1386** orej_invalid_stock_leg **1387** orej_auction_in_progress **1388** orej_invalid_nwt_price **1389** orej_invalid_auction_id **1390** orej_invalid_cross_specs **1391** orej_straight_cxl_not_allowed **1392** orej_cxl_replace_not_allowed **1393** orej_invalid_num_orders **1394** orej_order_ids_same **1395** orej_must_improve_price **1396** orej_msg_too_late_to_process **1397** orej_no_auction **1398** orej_nbbo_crossed **1399** orej_attribute_mismatch **1400** orej_symbol_not_open **1401** orej_exch_direct_must_be_limit **1402** orej_invalid_max_floor **1403** orej_invalid_min_quantity **1404** orej_invalid_underlying **1405** orej_invalid_risk_request

Field Name	Data Type	Description
rejectReason *(continued)*		**1406** orej_wait_iso_not_allowed **1407** orej_opg_aon_not_allowed **1408** orej_buy_market_order **1409** orej_bbo_invalid **1410** free_10198 **1411** orej_reserve_not_allowed **1412** orej_postonly_not_allowed **1413** orej_invalid_floor_brk **1414** orej_invalid_priv_ref **1415** orej_invalid_effective_time **1416** orej_invalid_good_til_date **1417** orej_invalid_cross_client_order_id **1418** orej_invalid_num_sides **1419** orej_invalid_display_when **1420** orej_invalid_price_prot_scope **1421** orej_invalid_auction_inst **1422** orej_invalid_stepup_price **1423** orej_invalid_stepup_price_type **1424** orej_invalid_spec_order_type **1425** orej_invalid_exposure **1426** orej_invalid_broker_pct **1427** orej_invalid_price_delta **1428** orej_must_be_limit **1429** orej_must_be_routable **1430** orej_must_persist **1431** orej_must_be_aon **1432** orej_opg_stop_not_allowed **1433** orej_reserve_modification_invalid **1434** orej_invalid_entitlement_req_id **1435** orej_invalid_no_party_entitlements **1436** orej_invalid_list_update_action **1437** orej_invalid_no_party_details **1438** orej_invalid_party_detail_id **1439** orej_invalid_party_detail_role **1440** orej_invalid_id_source **1441** orej_invalid_security_id **1442** orej_invalid_alloc_id **1443** orej_invalid_alloc_trans_type **1444** orej_invalid_trade_date **1445** orej_invalid_no_allocs **1446** orej_invalid_alloc_shares **1447** orej_invalid_no_execs **1448** orej_invalid_exec id **1449** orej_exec_broker_required **1450** orej_invalid_shares **1451** orej_invalid_display_range **1452** orej_postonly_replace **1453** orej_invalid_maturity_date **1454** orej_invalid_security_exchange **1455** orej_too_many_auctions **1456** orej_mar_cust_limit_qty **1457** orej_mar_cust_limit_notional **1458** orej_mar_cust_limit_agg_qty **1459** orej_mar_cust_limit_agg_notional **1460** orej_invalid_match_id **1461** orej_invalid_pta_account **1462** orej_invalid_pta_contracts **1463** orej_invalid_client_id **1464** orej_preferencing_not_allowed **1465** orej_invalid_stock_leg_giveup

Field Name	Data Type	Description
rejectReason *(continued)*		**1466** orej_invalid_contra_side_short_sell **1467** orej_pta_not_allowed **1468** orej_qcc_invalid_stock_ratio **1469** orej_cancel_strategy_mismatch **1470** orej_destination_not_available **1471** orej_invalid_underlying_price **1472** orej_invalid_underlying_qty **1473** orej_invalid_rfp_id **1474** orej_invalid_root_parties **1475** away_status_New **1476** away_status_PartiallyFilled **1477** away_status_Filled **1478** away_status_Done **1479** away_status_Canceled **1480** away_status_Replaced **1481** away_status_PendingCancel **1482** away_status_Stopped **1483** away_status_Rejected **1484** away_status_Suspended **1485** away_status_PendingNew **1486** away_status_Calculated **1487** away_status_Expired **1488** away_status_Accepted **1489** away_status_PendingReplace **1490** away_status_Restated **1491** away_status_Trade **1492** away_status_TradeCancel **1493** away_status_TradeCorrect **1494** alloc_status_Accepted **1495** alloc_status_BlockLevelReject **1496** alloc_status_PartialAccept **1497** alloc_status_NotYetProcessed **1498** invalidTimeOfAgreement **1499** invalidTradeReportId **1500** invalidTradeReportRefId **1501** invalidAgencyCross **1502** invalidHandlingInstr **1503** invalidEqualLeg **1504** invalidMinBlockTradeSize **1505** invalidDeferralThreshold **1506** invalidTradePublishIndicator **1507** invalidMaximumTradeReportSize **1508** invalidTradeType **1509** flexInstrExists **1510** invalidCircuitBreakerId **1511** invalidPriceProtectionTableCode **1512** invalidStrikePrice **1513** invalidExpirationDate **1514** REJ_NO_ERROR **1515** REJ_OPEN_ORDER_VALUE **1516** REJ_OPEN_ORDER_TOTAL_VALUE **1517** REJ_TRADE_VALUE **1518** REJ_TRADE_TOTAL_VALUE **1519** REJ_ORDER_RATE **1520** REJ_REPEATED_ORDER_GEN **1521** invalidStrategy **1522** undReentry **1523** invalidSelfReplenishVolume **OTHER** Other

Field Name	Data Type	Description
rejectReason *(continued)*		**Allowed Values: NYSE – AMER, ARCA, CHX, NSX, NYSE, ARCAOP, AMEROP** **0** **1** Invalid SenderCompID **2** Invalid SenderSubID **3** Invalid SendingTime **4** Invalid TargetCompID **5** Invalid TargetSubId **6** Invalid OnBehalfOfCompID **7** Invalid OnBehalfOfSubID **8** Invalid DeliverToCompID **9** Invalid DeliverToSubID **10** Invalid Account **11** Invalid ClOrdID **12** Invalid ExecInst **13** Invalid IDSource **14** Invalid OrderQty **15** Invalid OrdType **16** Invalid Price **17** Invalid Order Capacity **18** Invalid Security ID **19** Invalid Side **20** Invalid Symbol/Series **21** Invalid Text **22** Invalid TimeInForce **23** Invalid Settlement Type **24** Invalid FutSettDate **25** Invalid SymbolSfx **26** Invalid Open/Close **27** Invalid StopPx **28** Invalid Client ID **29** Invalid MinQty **30** Invalid MaxFloor **31** Invalid LocateReqd **32** Invalid ExpireTime **33** Invalid SecurityType **34** Invalid MaturityMonthYear **35** Invalid PutOrCall **36** Invalid StrikePrice **37** Invalid CoveredOrUncovered **38** Invalid CustomerOrFirm **39** Invalid MaturityDay **40** Invalid PegDifference **41** Invalid SellersDays **42** Invalid TradingSessionID **43** Invalid NoTradingSessions **44** Invalid DiscretionInst **45** Invalid DiscretionOffset **46** Invalid PriceType **47** Invalid ClearingFirm **48** Invalid ClearingAccount **49** Invalid PartyID **50** Invalid Optional Data **51** Invalid CrossID **52** Invalid StrategyIndicator **53** Invalid TradeID **54** Invalid NoSelfTrade **55** Invalid CAPStrategy **56** Invalid SpecialOrdType **57** Invalid RoutingInst **58** Invalid OffsetPrice

Field Name	Data Type	Description
rejectReason *(continued)*		**59** Invalid ExtendedExecInst **60** Invalid IntroducingBadgeID **61** Invalid BillTo **62** Invalid ParentFirmClOrdID **63** Invalid ParentFirmExchangeOrdID **64** Invalid ParentFirm **65** Invalid InterestType **66** Invalid DisplayInd **67** Invalid PegInd **68** Invalid CeilingFloorPrice **69** Invalid MinPegQty **70** Invalid DOrderAuctionPrice **71** Invalid DiscMaxVol **72** Invalid DicsRouteInd **73** Invalid MinimumTriggerVol (MTV) **74** Invalid Attributed Quote **75** Invalid Proactive If Locked **76** System not available **77** System full (MENG_RATE_EXCEEDED) **78** Throttle Reject **79** Symbol/Series Halted **80** No symbol/series permission **81** Price Too Far Outside **82** MWCB Halt **83** Market Closed **84** Symbol/Series Closed **85** LULD Cancel Instruction **86** No Price Slide Inst During SSR **87** Invalid StockLegGiveUp **88** Invalid NoLegs **89** Invalid LegPositionEffect **90** Invalid LegSymbol **91** Invalid LegCFICode **92** Invalid LegMaturityDate **93** Invalid LegStrikePrice **94** Invalid LegContractMultiplier **95** Invalid LegRatioQty **96** Invalid LegSide **97** Invalid LegRefID **98** Unsupported Order Type **99** UROUT **100** Primary Market Not Available **101** No NBBO/PBBO for Peg **102** No Market for Market Order **103** Marketable Price **104** Done for Day **105** Credit Limit Violation **106** Cancel Remaining IOC **107** Too Late to Cancel **108** Invalid PossResend **109** Cancel Pending **110** Symbol/Series already opened **111** Firm Bulk Cancel **112** OnBehalfOfCompID Blocked **113** ClearingFirm Blocked **114** Cancel/Replace Pending **115** Modify Pending **116** Cannot Flip Imbalance **117** Cannot Increase Imbalance **118** Pending Cancel - Imbalance Freeze

Field Name	Data Type	Description
rejectReason *(continued)*		**119** Pending Replace - Imbalance Freeze **120** Pending Modify - Imbalance Freeze **121** Pending Cancel - Routed Interest **122** Pending Replace - Routed Interest **123** Pending Modify - Routed Interest **124** Pending - Auction Running **125** Duplicative Order Check **126** Cancelled by Exchange **127** New Order **128** Fill **129** Partial Fill **130** Reduced **131** Replaced **132** No Market for Cross **133** STP Cancel* **134** Invalid PossDupe **135** TPID Blocked **136** Invalid Bulk Cancel **137** Pending Bulk Cancel **138** Symbol/Series Not Open **139** Symbol/Series Suspended **140** Symbol IPO Halt **141** Invld Inst During Imbalance Freeze **142** Invld Inst After Cutoff Time **143** Cancelled by Primary Market **144** Pending - Imbalance Freeze **145** No RLP Permission **146** Invalid Instruction for IOC's **147** System full (CG_RATE_EXCEEDED) **148** Pending Cancel - Auction Running **149** Pending Modify - Auction Running **150** Pending Replace - Auction Running **151** Invld Inst for Pending Order **152** SSH Price below NBB on ISO **153** IOC Received while Auction Running **154** Pending - Session Transition **155** Pend Cancel - Session Transition **156** Pend Modify - Session Transition **157** Pend Replace - Session Transition **158** Invalid For Tick Pilot **159** Invalid MMID **160** Invalid MPID **161** Invalid CancelInsteadOfReprice **162** Invalid RetailIndicator **163** SenderCompID Not Active **164** MPID Blocked **165** Invalid Timestamp **166** Invalid Permission for SenderCompID **167** Invalid UserData **168** Pillar Risk Mitigation **169** No Last Sale for Peg **170** Symbol Pending Closing Auction **171** Extreme Closing Order Imbalance **172** Invalid Multi-Message **173** Invalid Request Targeting Manual SenderCompID **174** Invalid DMMRejectReason **175** Pending - Pending Auction State **176** Pending Cancel – Pending Auction State **177** Pending Modify – Pending Auction State **178** Pending Replace – Pending Auction State

Field Name	Data Type	Description
rejectReason ***(continued)***		**179** Cancelled - DMM Manual Order Re-price **180** Too early to open **181** Too early to close **182** Symbol not frozen by Auction Request **183** Symbol is frozen by DBK GUI **184** Too Late - Auction Running **185** Too Late - Symbol Transition **186** No Eligible Crossing Interest **187** Book is locked/crossed **188** No interest exists **189** Imbalance too large **190** SSH price below SSR filing price **191** Symbol Already Closed **192** Opening template opened **193** Closing template opened **194** PRIN entered on auction template **195** Mandatory Indication submitted **196** Pending Crowd exists **197** No consolidated last sale **198** Paired qty exceeds max trade parameter **199** LULD or MWCB timer running **200** Locking/Crossing Away Quote **201** Cross Not Eligible **202** Reserved for future use **203** Pending Acceptance (for Ack on order arrival) **204** Rejected Cancel by DMM **205** Cancelled by DMM **206** Price is outside allowable range **207** Auction Validation In Progress **208** Invalid ManualActionID **209** Invalid AuctionSellIndicator **210** Invalid IntradaySellShortQty **211** Indication Template is Open **212** No Prev Closing Price **213** Cancelled due to Trading Collars **214** Underlying is in LULD State **215** Invalid ManualResponseType **216** No NYSE last sale **217** Symbol Direct Listing State **218** DMM GUI Reduction in Manual Order Qty **219** Cross Eligible **220** Dry Run **221** Unsupported by BrokerPlex **222** Cross Blocked by BBO/PBBO **223** Risk - Single Ord Max Qty **224** Risk - Single Ord Max Notional **225** Risk - Gross Credit Breach **226** Risk - Kill Switch **227** Invalid RouteToBroker **228** Spread Too Wide **229** Expire FOK **230** Locks displayed interest **231** MMQuote Price lock/cross contra side NBBO **232** Invalid Leg Symbol **233** Duplicate Leg Symbol **234** Symbol Not In Underlying **235** Leg ratios not in the most reduced form **236** Option leg ratio too high **237** Stock leg ratio too low **238** Complex series already exists

Field Name	Data Type	Description
rejectReason *(continued)*		**239** Number of legs incorrect
		240 Cancel - does not set NBBO inst
		241 Cancel - avoid reprice inst
		242 Allow reprice once, then cancel inst
		243 Invalid AllocationPct
		244 InvalidMMQuoteType
		245 Invalid MMSentTime
		246 Series Expired
		247 Invalid GiveUpMMID
		248 Invalid NoSides
		249 Pending TO Acceptance
		250 Symbol Exercise Style Not Same
		251 Invalid BulkAction
		252 Invalid CancelScope
		253 No Legal Width Spread
		254 Invalid TargetCancelMPID
		255 Invalid TargetCancelSender CompID
		256 Clear the Book Prev. Entered
		257 Timer Expired - API Allocation
		258 Invalid Number of Quotes
		259 Invalid OrderID
		260 Risk Control Event
		261 TO Rejected
		262 Cancelled - Corporate Action
		263 Too Late to Replace
		264 Invalid RefDelta
		265 Invalid StockPrice
		266 Reserved for future use
		267 Invalid TiedToStock
		268 Too Late to Allocate
		269 Customer Interest on BBO
		270 IDO Cannot be Modified
		271 Cross Outside BBO
		272 Pending TO Approval
		273 Invalid LegOpenClose
		274 Invalid RiskControlType
		275 Invalid RiskControlActivation
		276 Invalid BreachActionRequest
		277 Invalid IOCAttribution
		278 Invalid RiskActionType
		279 Invalid USDLimit
		280 IDO Already Exists
		281 Invalid TimeLimit
		282 Invalid PercentageLimit
		283 Invalid CountLimit
		284 Risk - Roll Transact Breach
		285 Risk - Roll Vol Breach
		286 Risk - Roll Pct Breach
		287 Risk - Roll GRMP Breach
		288 Risk - MM Arbitrage Breach
		289 Risk - MM Intrinsic Breach
		290 IDO Must Exist
		291 Invalid Allocation tag <insert invalid tag number>
		292 Invalid Trade tag <insert invalid tag number>
		293 Arbitrage Check
		294 Intrinsic Value Check
		295 Credit Debit Check
		296 Invalid Risk User
		297 Invalid Risk Entity
		298 Invalid Flex Series Already Exists

Field Name	Data Type	Description
rejectReason ***(continued)***		**299** Invalid Flex Series Key **300** Invalid Clear the Book **301** Invalid StockQty **302** Invalid PackageLinkID **303** Complex Max Series Breach **304** Invalid Flex EOD tag <insert invalid tag number> **305** Invalid PercentagePrice **306** Invalid PercentageStrike **307** Cancel Remaining GTX **308** COA Not Running **309** Invalid OrdStatus **310** MMID Blocked **311** Invalid CATIMID **312** Risk - Rej ISO **313** Risk - Rej Mkt Orders **314** Risk - Rej MOO MOC **315** Risk - Rej Early Trading **316** Risk - Rej Late Trading **317** Risk - Rej Restricted Symbol **318** Risk - Rej Sell Short for Symbol **319** Risk - Rej Sell Short Exempt for Symbol **320** Risk - Rej Ord Max Qty Symbol ADV **321** Risk - Max Duplicative Ord **322** Risk - Require LocateBroker **323** Invalid RiskRangeID **324** Invalid RiskMinimumValue **325** Invalid PriceScale **326** Invalid - Max Risk Symbols Exceeded **327** Invalid - Risk Settings Incomplete **328** Invalid AuctionID **329** GroupID Blocked **330** Pending FLEX Price Msg **331** Risk – Ord Rate Threshold **332** Routing Venue Not Available **333** Cancelled by Routing Venue **334** Invalid TargetStrategy **335** Cancel - IOC crossing away quote **400** Repriced **800** Broker Reject **906** System full (CGA_RATE_EXCEEDED) **999** Unknown Issue Encountered **Allowed Values: NYSE – AMEROP** (*valid until 10/31/2023*) **MMQ Reason Codes:** **1** System Unavailable **3** Invalid Sequence Number **4** Invalid Series Index **5** Series Non-Active **6** Invalid Value **7** Client Session Already Active **8** Client Session Disabled **9** Invalid MMID **10** Invalid Series **11** Underlying Mismatch **12** Invalid Price **13** Invalid Size **14** Unknown Underlying **15** Invalid Market Maker for Underlying **16** Invalid Market Maker **17** Logon Read Failed

Field Name	Data Type	Description
rejectReason *(continued)*		**18** No Logon Message **19** User Name Not Found **20** Risk Mitigation Limit Exceeded **21** Invalid Quote Count **22** Invalid Underlying Symbol **23** Invalid Side **24** Invalid Price Increment **25** Series Halted **26** MM is not Active **27** Duplicate Quote ID **28** Internal Error **29** Market is Closed **30** Risk Mitigation is Above Allowed Value **31** Disconnect Takedown **32** Series Closed **33** Underlying Group Mismatch **34** Market Not Open **36** Unsupported Risk Mitigation Type **37** Global Breach **39** Too Executable **40** MMLO Quote Locking/Crossing **41** Market Maker Quote Locking/Crossing Away **42** Feature Not Available **43** Self-Trade Prevention **44** Invalid Value **45** Too Aggressive for Re-Pricing **46** Invalid Display Price **Order Reject Reason Codes:** **100** Miscellaneous **101** Unsupported Feature **102** Exchange closed **103** Invalid Data **104** Too Late to Enter **105** Unknown Order Sender **106** Invalid Data for Outcry **107** Rejected Due to Risk Mitigation **Cancel Reject Reason Codes:** **200** Too Late to Cancel **201** Unknown Order for cancel/Invalid Data for Replacement **202** Miscellaneous
relatedMarketCenterId	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* For the non-tape "riskless" leg of a riskless principal transaction, the facility or market where the first leg of the transaction was reported. **Allowed Values** **0** ADF/ORF **1** Nasdaq TRF **2** FINRA/Nasdaq TRF Chicago **3** NYSE TRF **A** NYSE American, LLC **B** Nasdaq BX **C** NYSE National, Inc **E** MEMX Exchange **F** Foreign Mkt

Field Name	Data Type	Description
		G Cboe BYX Exchange, Inc. **H** Cboe BZX Exchange, Inc. **I** International Securities Exchange **J** Cboe EDGA Exchange, Inc. **K** Cboe EDGX Exchange, Inc. **L** LTSE Exchange **M** NYSE Chicago **N** New York Stock Exchange, LLC (NYSE) **O** Unknown Market Center **P** NYSE Arca, Inc. **Q** Nasdaq Stock Market, LLC **U** Unspecified Mult Mkt Trades **V** Investors' Exchange, LLC. (IEX) **W** CBoe Stock Exchange, Inc. **X** Nasdaq PSX LLC **Y** MIAX Pearl Exchange
reportedShareQuantity	Unsigned	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Number of shares traded as reported to the SIP.
reportedSideCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Side of the trade (buy/sell/cross) from the perspective of the firm with the reporting obligation. **Allowed Values** **B** Buy Side **S** Sell Side **X** Crossed Trade
reportedUnitPrice	Price	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Unit price of the trade as reported to the SIP.
reporter	Reporter ID	*Event(s): Note (NOTE), Self-Help Declaration (SHD)* *Reference Data: Market Maker Dictionary Entry (MMDE), Member Dictionary Entry (MDE), Member Alias Detail Entry (MADE), Option Series Dictionary Entry (OSDE), Complex Option Dictionary Entry (CODE)* Reporter ID of the entity reporting the events or reference data.
reportingExecutingMpid	Member Alias	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* MPID of the executing party.
reportingObligationFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies if the reporting-side firm had the reporting obligation for the trade under FINRA trade reporting rules. **Allowed Values** **Y** Reporting Firm Has Reporting Obligation
reportingSideBranchSequenceIdentifier	Text (20)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Branch/sequence number of the reporting-side firm.

Field Name	Data Type	Description
reportingSideCapacityCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Capacity of the reporting-side firm. **Allowed Values** **A** Agency **P** Principal **R** Riskless Principal
reportingSideClearingNumber	Unsigned	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Clearing number of the firm that cleared the trade for the reporting-side firm.
reportingSideMemoText	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Provides a link (via Control Number) to the original trade report, when a subsequent report is submitted to reallocate some of the trade volume to a different capacity. This is a free-form text field; participants can enter any information in this field.
reportingSideMpid	Member Alias	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* MPID of the firm with the reporting obligation.
reportingSideShortSaleCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies a short sale by the executing firm and indicates the type of short. **Allowed Values** **SS** Short Sale **SX** Short Sale Exempt
reportingSubmittingEntityId	Text (4)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates the entity that initiated the submission. For a FINRA-initiated submission on behalf of the firm, this will be 'FNRA'. Otherwise, for a firm-initiated submission, it will be the firm MPID. For NC TRF, NQ TRF and NY TRF, this is always NQTC, NQTR or NYTR. For ADF and ORF it is the MPID of the submitting firm.
reportTypeCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies whether this is a No/Was report. **Allowed Values** **N** No **W** Was
result	Choice	*Event(s): Order Route, Order Cancel Route, Order Modify Route; Option Route, Complex Option Route (OCOR), Modify Option Route, Option Cancel Route* The result of the Route, Cancel Route or Modify Route request communicated to the exchange. **Allowed Values** **ACK** Acknowledged **REJ** Rejected **NR** No Response **UNSOL** Unsolicited: only valid for an unsolicited cancel route

Field Name	Data Type	Description
resultTimestamp	Timestamp	*Event(s): Order Route, Order Cancel Route, Order Modify Route; Option Route, Complex Option Route (OCOR), Modify Option Route, Option Cancel Route* The date/time the result of Route, Modify Route, or Cancel Route request was received.
retransmissionRequester	Text (2)	*Event(s): FINRA Halt/Resume (FHR)* Indicates if the message is an original transmission or retransmission. If the message is a retransmission, this field indicates the two-character retransmission identifier of the intended data recipient.
reversalFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates that the trade report is reversal transaction. **Allowed Values** **Y** Reversal **N** Not a Reversal
revokedTimestamp	Timestamp	*Event(s): Self-Help Declaration (SHD)* Date and time the self-help was revoked. If self-help is not revoked by the end of the day, this field may be left unreported or can be set to the closing time. However, another self-help event must be reported for the next day.
routedOrderID	Text (40)	*Event(s): Order Accepted, Order Modified, Simple Option Order Accepted, Complex Option Order Accepted, Stock Leg Order, Option Order Modified, Complex Option Order Modified, Stock Leg Modified* The ID assigned to this order by the routing firm when submitting the order to the exchange. *Event(s): Equity Order Modified, Equity Order Adjusted, Option Order Modified, Option Order Adjusted* For the return of unexecuted liquidity previously routed away, the exchange-assigned ID used to route the order away. *Event(s): Order Modify Route (EOR), Modify Option Route (OOMR), Complex Option Route (OCOR)* The routedOrderID as represented in the original or most recent Route/Modify Route message sent to the routing broker. *Event(s): Reject Message Event (RME)* The routedOrderID as represented in the message that was rejected. *Event(s): Equity Best Bid and Offer Event (EBBO)* The quote ID that the firm used in the API message when they sent the quote to the display only facility.

Field Name	Data Type	Description
routedOriginalOrderID	Text (40)	*Event(s): Order Modified, Option Order Modified, Complex Option Order Modified, Stock Leg Modified* The routedOrderID for the order, as sent by the routing broker in the original route message, or the most recent modify message (in FIX OrigClOrdId, in OUCH Existing Order Token). *Event(s): Order Modify Route, Modify Option Route events* The routedOrderID as represented in the original or most recent Route/Modify Route message sent to the routing broker. *Event(s): Reject Message Event (RME)* The ID for the order being modified, as sent by the routing broker in the original route message, or the most recent modify message (in FIX OrigClOrdID, in OUCH Existing Order Token).
routingParty	Text (8)	A string used to identify the entity on the other side of an accepted or route event. *Event(s): Order Accepted, Simple Option Order Accepted, Complex Option Order Accepted* In the events above, this is the unique identifier for the firm that sent the order to the exchange. *Event(s): Order Route (EOR), Order Fill (EOF), Order Modify Route (EMR), Order Cancel Route (ECR), Option Route, Complex Option Route (OCOR), Modify Option Route (OOMR), Option Cancel Route (OOCR)* In the events above, this is the firm to which the exchange routed the order. *Event(s): Order Modified (EOM), Order Adjusted (EOJ), Option Order Modified (OOM), Complex Option Order Modified (OCOM), Option Order Adjusted (OOJ), Complex Option Order Adjusted (OCOJ)* In the events above, this value can be either the customer that sent the order to the exchange or the firm to which the exchange routed the order. When the initiator value is Firm or Market Maker, report the unique identifier for the firm that sent the order to the exchange. When the initiator value is Exchange and the event represents routed quantity returned unexecuted, report the firm to which the exchange routed the order. *Event(s): Reject Message Event (RME)* The ID string used to identify the entity that routed the rejected message to the exchange. *Event(s): Equity Best Bid and Offer (EBBO)* The ID string used to identify the entity that routed the quote to the display-only facility
saleCondition	Text (8)	*Event(s): Supplemental Trade Event (STE), Order Trade, Order Fill, Trade Correction, Simple Option Trade, Stock Leg Fill, Option Trade Correction* Indicates a special condition under which a trade was reported. The first character must be either 'E' or 'O' indicating whether the following characters are to be interpreted as OPRA sale condition codes for options or UTP/CTS sale condition codes for equities. 'E' stands for the UTP/CTS, while 'O' stands for the OPRA. The following characters will use the single-character codes as defined in the OPRA, UTP, and CTS specifications - one character code for as many conditions as apply. Note that the <space> character is a valid code. **Allowed Values: Second character if first character is O (OPRA Values)** *effective through 11/1/2019* **blank** Indicates that the transaction was a regular sale and was made without stated conditions **A** Transaction previously reported (other than as the last or opening

Field Name	Data Type	Description
saleCondition ***(continued)***		report for the particular option contract) is not to be canceled **B** Transaction is being reported late and is out of sequence, i.e. later transactions have been reported for the particular option contract. **C** Transaction is the last reported for the particular option contract and is now canceled. **D** Transaction is being reported late, but is in the correct sequence, i.e. no later transactions have been reported for the particular option contract. **E** Transaction was the first one (opening) reported for this day for the particular option contract. Although later transactions have been reported, this transaction is not to be canceled. **F** Transaction is a late report of the opening trade and is out of sequence: i.e. other transactions have been reported for the particular option contract. **G** Transaction was the only one reported this day for the particular option contract and is now to be canceled **H** Transaction is a late report of the opening trade, but is in the correct sequence, i.e., no other transactions have been reported for this particular option contract. **I** Transaction was executed electronically. This prefix appears solely for information; process as a regular transaction. **J** Transaction is a reopening of an option contract in which trading has been previously halted. This prefix appears solely for information; process as a regular transaction. **K** Transaction is an option contract for which the terms have been adjusted to reflect a stock dividend, stock split, or similar event. This prefix appears solely for information; process as a regular transaction. **L** Transaction represents a trade in two options in the same option class (a buy and sell in the same class). This prefix appears solely for information; process as a regular transaction. **M** Transaction represents a trade in two options in the same option class (a buy and sell in a put and a call). This prefix appears solely for information; process as a regular transaction **N** Transaction is the execution of a sale at a price agreed upon by the floor personnel involved, where a condition of the trade is that it be reported following a non-stopped trade of the same series at the same price. **O** Cancel stopped transaction **P** Transaction represents the option portion of an order involving a single option leg (buy or sell of a call or put) and stock. The prefix appears solely for information; process as a regular transaction. **Q** Transaction represents the buying of a call and the selling of a put for the same underlying stock or index. This prefix appears solely for information; process as a regular transaction **R** Transaction was the execution of an order that was 'stopped' at a price that did not constitute a Trade-Through on another market at the time of the stop. **S** Transaction was the execution of an order identified as an Intermarket Sweep Order **T** Transaction reflects the execution of a 'benchmark trade'. **X** Transaction is Trade Through Exempt. The transaction should be treated like a regular sale. **Allowed Values: Second character if first character is O (OPRA Values)**

Field Name	Data Type	Description
saleCondition ***(continued)***		*effective beginning 11/4/2019* **blank** Indicates that the transaction was a regular sale and was made without stated conditions. **A** Transaction previously reported (other than as the last or opening report for the particular option contract) is now to be canceled. **B** Transaction is being reported late and is out of sequence; i.e., later transactions have been reported for the particular option contract. **C** Transaction is the last reported for the particular option contract and is now canceled. **D** Transaction is being reported late, but is in the correct sequence; i.e., no later transactions have been reported for theparticular option contract. **E** Transaction was the first one (opening) reported this day for the particular option contract. Although later transactions have been reported, this transaction is now to be canceled. **F** Transaction is a late report of the opening trade and is out of sequence; i.e., other transactions have been reported for the particular option contract. **G** Transaction was the only one reported this day for the particular option contract and is now to be canceled. **H** Transaction is a late report of the opening trade, but is in the correct sequence; i.e., no other transactions have been reported for the particular option contract. **I** Transaction was executed electronically. Prefix appears solely for information; process as a regular transaction. **J** Transaction is a reopening of an option contract in which trading has been previously halted. Prefix appears solely for information; process as a regular transaction. **S** Transaction was the execution of an order identified as an Intermarket Sweep Order. Process like normal transaction. **a** Transaction was the execution of an electronic order which was “stopped” at a price and traded in a two sided auction mechanism that goes through an exposure period. Such auctions mechanisms include and not limited to Price Improvement, Facilitation or Solicitation Mechanism. **b** Transaction was the execution of an Intermarket Sweep electronic order which was “stopped” at a price and traded in a two sided auction mechanism that goes through an exposure period. Such auctions mechanisms include and not limited to Price Improvement, Facilitation or Solicitation Mechanism marked as ISO. **c** Transaction was the execution of an electronic order which was “stopped” at a price and traded in a two sided crossing mechanism that does not go through an exposure period. Such crossing mechanisms include and not limited to Customer to Customer Cross and QCC with a single option leg. **d** Transaction was the execution of an Intermarket Sweep electronic order which was “stopped” at a price and traded in a two sided crossing mechanism that does not go through an exposure period. Such crossing mechanisms include and not limited to Customer to Customer Cross. **e** Transaction represents a non-electronic trade executed on a trading floor. Execution of Paired and Non-Paired Auctions and Cross orders on an exchange floor are also included in this category. **f** Transaction represents an electronic execution of a multi leg order traded in a complex order book

Field Name	Data Type	Description	
saleCondition *(continued)*		**g**	Transaction was the execution of an electronic multi leg order which was “stopped” at a price and traded in a two sided auction mechanism that goes through an exposure period in a complex order book. Such auctions mechanisms include and not limited to Price Improvement, Facilitation or Solicitation Mechanism.
		h	Transaction was the execution of an electronic multi leg order which was “stopped” at a price and traded in a two sided crossing mechanism that does not go through an exposure period. Such crossing mechanisms include and not limited to Customer to Customer Cross and QCC with two or more options legs.
		i	Transaction represents a non-electronic multi leg order trade executed against other multi-leg order(s) on a trading floor. Execution of Paired and Non-Paired Auctions and Cross orders on an exchange floor are also included in this category.
		j	Transaction represents an electronic execution of a multi Leg order traded against single leg orders/ quotes.
		k	Transaction was the execution of an electronic multi leg stock/options order which was “stopped” at a price and traded in a two sided auction mechanism that goes through an exposure period in a complex order book. Such auctions mechanisms include and not limited to Price Improvement, Facilitation or Solicitation Mechanism.
		l	Transaction was the execution of an electronic multi leg order which was “stopped” at a price and traded in a two sided auction mechanism that goes through an exposure period and trades against single leg orders/ quotes. Such auctions mechanisms include and not limited to Price Improvement, Facilitation or Solicitation Mechanism.
		m	Transaction represents a non-electronic multi leg order trade executed on a trading floor against single leg orders/ quotes. Execution of Paired and Non-Paired Auctions on an exchange floor are also included in this category.
		n	Transaction represents an electronic execution of a multi leg stock/options order traded in a complex order book.
		o	Transaction was the execution of an electronic multi leg stock/options order which was “stopped” at a price and traded in a two sided crossing mechanism that does not go through an exposure period. Such crossing mechanisms include and not limited to Customer to Customer Cross.
		p	Transaction represents a non-electronic multi leg order stock/options trade executed on a trading floor in a Complex order book. Execution of Paired and Non-Paired Auctions and Cross orders on an exchange floor are also included in this category.
		q	Transaction represents an electronic execution of a multi Leg stock/options order traded against single leg orders/ quotes.
		r	Transaction was the execution of an electronic multi leg stock/options order which was “stopped” at a price and traded in a two sided auction mechanism that goes through an exposure period and trades against single leg orders/ quotes. Such auctions mechanisms include and not limited to Price Improvement, Facilitation or Solicitation Mechanism.
		s	Transaction represents a non-electronic multi leg stock/options order trade executed on a trading floor against single leg orders/ quotes. Execution of Paired and Non-Paired Auctions on an exchange floor are also included in this category.
		t	Transaction represents execution of a proprietary product non-electronic multi leg order with at least 3 legs. The trade price may be outside the current NBBO.
		u	Transaction represents an execution in a proprietary product done as

Field Name	Data Type	Description
saleCondition ***(continued)***		part of a multilateral compression. Trades are executed outside of regular trading hours at prices derived from end of day markets. Trades do not update Open, High, Low, and Closing Prices. **v** Transaction represents a trade that was executed outside of regular market hours. Trades do not update Open, High, Low, and Closing Prices. **Allowed Values: Second character if first character is E (UTP and CTS Values)** **@** Regular Sale **blank** No Sale Condition required within the category it appears (Long Trade Format Only) **A** Acquisition **B** Bunched Trade or Average Price Trade **C** Cash Sale **D** Distribution **E** Automatic Execution **F** Intermarket Sweep **G** Bunched Sold Trade **H** Price Variation Trade **I** Odd Lot Trade **K** Rule 155 Trade (AMEX) **L** Sold Last **M** Market Center Official Close **N** Next Day Trade (Next Day Clearing) **O** Opening Prints / Market Center Opening Trade **P** Prior Reference Price **Q** Market Center Official Open **R** Seller **S** Split Trade **T** Form T (Extended Hours Trade) **U** Extended Trading Hours (Sold Out of Sequence) **V** Contingent Trade **W** Average Price Trade **X** Cross Trade **Y** Yellow Flag Regular Trade **Z** Sold (out of Sequence) **1** Stopped Stock (Regular Trade) **4** Derivatively Priced **5** Re-Opening Prints (Market Center Reopening Trade) **6** Closing Prints (Market Center Closing Trade) **7** Qualified Contingent Trade (QCT) **8** Placeholder for 611 Exempt **9** Corrected Consolidated Close (per listing market)
sellDetails	Order Trade Side Details	*Event(s): Order Trade, Trade Correction, Simple Option Trade, Option Trade Correction* Information for the sell side of the trade. Format and element definitions for sellDetails are described in sideTradeEvent in section 4.5.

Field Name	Data Type	Description
sentTimestamp	Timestamp	*Event(s): Quote Event (OQ), Quote Cancel Event* The date/time when the market maker sent the quote or quote cancel to the exchange.
sequenceNumber	Unsigned	*Event(s): All Stock Exchange Events, All Options Exchange Events, Note (NOTE), Reject Message Event (RME), Equity Best Bid and Offer Event (EBBO)* The sequence number of the event, used to identify the sequence of events when multiple events have the same timestamps. The sequence number is required to be strictly increasing for a given reporter, date [8], and symbol, and can be used to sort each event in chronological order where multiple events have the same timestamp. For more detail, please refer to section 3.1: Timestamps and Sequence Numbers.
seqNumSub	Text (10)	*Event(s): All Options Events, Note (NOTE), Reject Message Event (RME), Equity Best Bid and Offer Event (EBBO)* A sequence number subsystem identifier.
session	Text (40)	*Event(s): Order Accepted, Order Route, Order Modified, Order Adjusted, Order Fill, Order Cancel Route, Order Modify Route, Simple Option Order Accepted, Complex Option Order Accepted, Option Order Modified, Complex Option Order Modified, Option Order Adjusted, Complex Option Order Adjusted, Option Route, Complex Option Order Route, Modify Option Route, Option Cancel Route, Reject Message Event (RME), Equity Best Bid and Offer Event (EBBO)* The name/ID of the session being used to send the order (from the routing firm to the exchange, or from the exchange to the routing broker). If this event represents a leg of a complex order, the Session must be the same as reported in the parent complex order. For modification and adjustment events, the value is for the firm that routed to the exchange when the initiator is Firm or Market Maker. When the initiator is Exchange and the event represents routed liquidity returned unexecuted, then the value is what the exchange used to route the order away.
sessionIdentifier	Choice	*Event(s): FINRA Halt/Resume (FHR)* Indicates the market session of the message. **Allowed Values** **A** All Market Sessions **U** US Market Sessions
settlement	Choice	*Reference Data: Option Series Dictionary Entry (OSDE)* Specifies the settlement of option in Simple Option Series Dictionary Entries. **Allowed Values** **AM** At the open **PM** At the close **Asian** European/PM settlement, but the exercise settlement value is the arithmetic average of the closing prices of the underlying index on 12 pre-determined, consecutive monthly observation dates. **Cliquet** European/PM settlement, but the exercise settlement value is the greater of zero, or [(closing price of the underlying index on the initial trade date) * (sum of the monthly capped returns)] + strike price.

[8] For purposes of 24-hour trading, a “day” is considered to be a single cycle date. See the definition of `cycleDate` for details.

Field Name	Data Type	Description
side	Choice	*Reference Data: Complex Option Dictionary Entry (CODE)* *Event(s): Supplemental Trade Event (STE), Order Accepted, Order Route, Order Modified, Order Adjusted, Order Trade, Order Fill, Order Restatement, Trade Correction, Simple Option Order Accepted, Complex Option Order Accepted, Stock Leg Order, Option Route, Complex Option Order Route, Option Order Modified, Complex Option Order Modified, Option Order Adjusted, Complex Option Order Adjusted, Option Trade, Stock Leg Fill, Post Trade Allocation* Side of the event. Note that AsDirected and Opposite are only used for complex option order accepted events. **Allowed Values** **Buy** **Sell** **Short** **Exempt** **Cross** **CrossExempt** **CrossShort** **CrossShortExempt** **AsDirected** **Opposite**
specialTradeCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies special and step-out trades. **Allowed Values** **S** Step-Out Trade The following codes are only applicable to Nasdaq-TRF and ORF trades: **A** Step-Out Trade with Section 3 Fee **B** Special and Step-Out trade with Section 3 Fee **F** Fee Transfer – Occurred on Nasdaq **I** Step-In trade **J** Special and Step-In Trade **O** Fee Transfer – Occurred on Another Market **Q** Step-Out of Nasdaq Exchange Trade **X** Special and Step-Out Trade – Instructs the NSCC not to include the trade in CNS **Y** Special Trade – Instructs the NSCC not to include the trade in CNS settlement

Field Name	Data Type	Description
sroRequiredModifier	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Further classification of the trade with regard to SRO required detail. This can either be entered by the firm or appended by the system. **Allowed Values** **1** Stop stock (regular trade) **A** Acquisition **B** Bunched Trade **D** Distribution **E** Automatic execution (system) **H** Intraday trade detail (system) **I** Odd lot **K** Rule 155 Amex/Rule 127 NYSE **M** Market Center close price (system) **O** Odd lot **P** Prior reference price **Q** Market center open price (system) **R** Away from market sale **S** Split trade **V** Contingent Trade **W** Average price trade **X** Exercise of OTC option
status	Choice	*Reference Data: Member Dictionary Entry (MDE)* The status of the member on the reporting date. **Allowed Values** **Active** An active member of the SRO (ID must be CRD) **Inactive** An inactive member of the SRO (ID must be CRD) **NonMember** An entity that is not a member of the SRO. For example, if the routing broker dealer is not a member of the exchange, it would be listed here (ID must be CRD). **Internal** Some internal part of the SRO system (a utility or facility) which will be used in reportable events. **Other** Another entity (e.g., foreign firm) without a CRD number.
statusTime	Timestamp	*Reference Data: Market Maker Dictionary Entry (MMDE)* Time of change in market maker's status. If one record for a member alias and symbol combination is provided, it is assumed to be active for the entire day. For market making initiations not at the open, provide the start time
strikePrice	Numeric(10,8)	*Reference Data: Option Series Dictionary Entry (OSDE)* In Simple Option Series Dictionary Entries, this field is the pre-arranged transaction price if the option is exercised. Note that if option kind = FLEXPCT, this will be the percentage.

Field Name	Data Type	Description
supervisoryEntryCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates if a Market Operations Supervisor entered the trade message on behalf of the reporting side of the trade transaction. **Allowed Values** **D** Supervisory Entry for Service Desk Participant **S** Supervisory Entry for Non-Service Desk Participant
symbol	Symbol	*Event(s): All Stock Exchange Events, All Options Stock Leg Events, Note (NOTE), FINRA Halt/Resume (FHR), Supplemental Trade Event (STE), Reject Message Event (RME), Equity Best Bid and Offer Event (EBBO)* *Reference Data: Market Maker Dictionary Entry (MMDE), Complex Option Dictionary Entry (CODE)* The stock symbol. Note that for all events of stock exchange, or options stock leg related events, this field may be in either the symbology of the listing exchange or a valid alias. However, in Symbol Entry, or stock leg of Complex Option Dictionary entry, this must be in the symbology of the listing exchange.
Symbol Entry Pairs	Name/Value Pairs	This is a data type. Currently, this data type must be used for the field "attributes" found in the reference data element: Symbol Entry. **Allowed Values** **TPG** Tick Pilot Group (Choice) - requires one of the defined values (e.g., TPG=TG2) from the following list: **CTRL** Control Group **TG1** Test Group 1 **TG2** Test Group 2 **TG3** Test Group 3
systemAppendedTradeReportingModifierFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies if the Trade Reporting Modifier Code was entered by the reporting firm or appended by the reporting facility. **Allowed Values** **Y** System Appended **N** Not System Appended
testSeriesFlag	Boolean	*Event(s): Option Series Dictionary Entry (OSDE)* Indicates that the entry represents a test symbol.
timeInForce	Choice	*Event(s): Order Accepted, Order Route, Order Modified, Order Modify Route, Order Restatement, Simple Option Order Accepted, Complex Option Order Accepted, Complex Option Order Modified, Stock Leg Order, Option Order Modified, Option Route, Complex Option Order Route (OCOR), Modify Option Route, Option Order Restatement* Specifies the Time-In-Force for an order. Supported TIF values are listed below. **Allowed Values** **AOK** Auction or Kill **CLO** At the Close **DAY** A day order **IOC** Immediate or Cancel **GTC** Good till Canceled

Field Name	Data Type	Description
timeInForce ***(continued)***		**GTT** Good till Time (requires XTIME in handlingInstructions) **GTD** Good till Date **GTX** Good till Crossing **FOK** Fill or Kill **OPG** At the Open **REG** Regular Hours Only **WCO** While Connected **Allowed Values: Cboe** **EXT** Extended Day **Allowed Values:CHX** **AOO** Auction-only order **GFS** Good for Seconds **Allowed Values: IEX** **SYS** System Hours **EXT** Day + Extended Hours **Allowed Values: NASDAQ Equities** **AHC** After Hours Close **CLO** On Close **EXT** Extended Days **OPG** On Open **Allowed Values: MIAX** **SAO** SettlementAuctionOnly **Allowed Values: LTSE** **SYS** System Session ("SYS"). Orders entered into the System marked SYS may trade during System Hours and expire at the end of the Post-Market Session.
tradeBreakTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the reporting party submitted their break request.
tradeBrokenTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the contra party submitted their break confirmation.
tradeCorrectionClassCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Trade Correction Classification. **Allowed Values** **A** Audit Trail Only **B** Both T & C **C** Clearing **T** Tape
tradeDate	Date	The date on which a trade occurred.

Field Name	Data Type	Description
tradeID	Text (40)	*Event(s): Supplemental Trade Event (STE), Order Trade, Trade Break, Trade Correction, Option Trade, Post Trade Allocation, Option Trade Break, Option Trade Correction* An identifier for the trade, unique for the given exchange, date, and Symbol/OptionID.
tradeModifierSroTime	Time	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Time associated with Prior Reference Price or Stopped Stock trade.
tradeModifierThroughExemptTime	Time	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* User Trade Thru Exempt Modifier Time.
tradeReferenceNumber	Text (20)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Trade Reference Number
tradeReportDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date the trade report was received by the reporting facility.
tradeReportingModifier	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Further classification of the trade with regard to Extended Hours/Sequence. This can either be entered by the firm or appended by the system. **Allowed Values** **L** Sold last (late reported) **T** Pre- or Post-market Trade **U** Pre- or Post-market Trade Reported Out-of-Sequence (late) **Z** Sold Out-of-Sequence (late)
tradeReportTimestamp	Timestamp	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date and time the trade report was received by the reporting facility.
tradeSettlementDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date on which the trade will settle.
tradeSettlementModifier	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Identifies a Reg NMS Settlement Type Sale Condition Code associated with a trade transaction. **Allowed Values** **@** Regular settlement **C** Cash settlement **N** Next day settlement **R** Seller settlement

Field Name	Data Type	Description
tradeSourceCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Trade Sources. **Allowed Values** **B** Batch File **C** CTCI **F** FIX to MPP **J** FIX to ACT **K** QIX to ACT **M** Mass Cancel or Mass Correction **Q** QIX **S** FINRA Supervisor **W** Web
tradeStatusCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Final status of the trade at the time it was reported. **Allowed Values** **A** Accepted; Locked-in Trade **B** Broken **C** Canceled **D** Declined **E** Errored **F** Forced Matched; Locked-in Trade **G** One-sided Submission **H** Hanging Trade **I** Inhibited (by clearing firm) **K** Rejected Sizable Trade **L** Automatic Locked-in Trade at the end of T + 1 **M** Matched; Locked-in Trade (also used for AGU and PSA trades) **N** No Portion of No/Was Trade **R** Locked-In Trade; Received via an execution system interface for NQ TRF **T** Trade Reporting Only; Not for clearing submission **X** As-Of Open or As-Of Spilt Trade; not forwarded to NSCC, but is available for query
tradeThroughExemptFlag	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Indicates that the trade is trade through exempt. **Allowed Values** **Y** Trade Through Exemption **N** No Trade Through Exemption

Field Name	Data Type	Description
tradeThroughExemptionModifier	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Further classification of the trade with regard to Trade Through Exemption. This is entered by the firm when it reports the trade. **Allowed Values** **2** NASD Self Help Indicator **3** Intermarket Sweep - Outbound **4** Derivatively Price **5** Market Center Reopen **6** Market Center Closing **7** Error Correction **8** Print Protection **9** Correct Consolidated Close Price as per Listing Market **F** Intermarket Sweep **J** NASD Subpenny Indicator **O** Market Center Open **V** NASD Contingent Indicator
trfContraControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Number used for interaction between TRFs and Firms; populated only when trade is matched by comparison. May not be unique for a given day.
trfControlNumber	Text (30)	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Control Number used for interaction between Firms and TRFs. May not be unique for a given day.
trfProcessingDate	Date	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* Date FINRA received the record from the reporting facility.
trfTradeModifierLateCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* System Trade Modifier - Time Modifiers - Updated by TRF. **Allowed Values** **T** Executed Outside Normal Market Hours **U** Executed Outside Normal Market Hours and Reported Late **Z** Executed During Normal Market Hours and Reported Late
trfTradeModifierSroCode	Choice	*Event(s): FINRA TRF/ORF/ADF Transaction Data (TRF)* User Trade Modifier - SRO - Updated by TRF. SRO detail sale condition. **Allowed Values** **I** Odd Lot Trade **V** Contingent Trade **W** Weighted Average Price
type	Message Type	*Event(s): All* Specifies the event type. **General Events** **NOTE** Note **SHD** Self Help Declaration **STE** Supplemental Trade Event

Field Name	Data Type	Description	
type *(continued)*		**RME**	Reject Message Event
		Equities Events	
		EOA	Order Accepted
		EOR	Order Route
		EIR	Internal Order Route
		EOM	Order Modified
		EOJ	Order Adjusted
		EOC	Order Canceled
		EOT	Order Trade
		EOF	Order Fill
		ECR	Order Cancel Route
		EMR	Order Modify Route
		EORS	Order Restatement
		ETB	Trade Break
		ETC	Trade Correction
		Options Events	
		OQ	Quote
		OQC	Quote Cancel
		OOA	Simple Option Order Accepted
		OCOA	Complex Option Order Accepted
		OSL	Stock Leg Order
		OOM	Option Order Modified
		OCOM	Complex Option Order Modified
		OSLM	Stock Leg Modified
		OOJ	Option Order Adjusted
		OCOJ	Complex Option Order Adjusted
		OSLJ	Stock Leg Adjusted
		OOC	Option Order Canceled
		OOR	Option Route
		OCOR	Complex Option Route
		OIR	Internal Option Route
		OCIR	Internal Complex Option Route
		OOMR	Modify Option Route
		OOCR	Option Cancel Route
		OT	Simple Option Trade
		OSLF	Stock Leg Fill
		OPTA	Post Trade Allocation
		OORS	Option Order Restatement
		OTB	Option Trade Break
		OTC	Option Trade Correction
		FINRA Events	
		EBBO	Equity Best Bid and Offer
		FHR	FINRA Halts/Resumes
		TRF	FINRA TRF/ORF/ARF Transaction Data

Field Name	Data Type	Description
type ***(continued)***		**Reference Data** **MDE** Member Dictionary Entry **MADE** Member Alias Detail Entry **CODE** Complex Options Dictionary Entry **OSDE** Options Series Dictionary Entry **MMDE** Market Maker Dictionary Entry
undefinedNoteData	Name/Value Pairs	*Event(s): Note (NOTE)* A list of key/value pairs, providing machine parseable data for the notation in a Note Event. The attributes are not defined in the specs, and can be any values as long as they conform to the format for a list of name/value pairs.
underlyingType		*Reference Data: Option Series Dictionary Entry (OSDE)* This field specifies whether a simple option series has an equity or index as its underlying. The underlying type mapping is consistent with the same mapping used at OCC (e.g., ETF is treated as Equity and WCO is treated as Index). **Allowed Values** **Equity** **Index**
version	Version	This is a data type, not a field. Digits and decimals are the only allowed characters. The first character must be a digit group followed by any number of optional pairs of decimals and digit groups.
workingPrice	Price	*Event(s): Order Accepted, Order Restatement, Simple Option Order Accepted, Option Order Modified, Option Order Restatement* The working price of the order.

Appendix G. CATFT (fileX) Token Service Instructions and Examples

Pre-requisite requirement

The CATFT service requires established direct connectivity to CAT as the service is only accessible via direct connectivity.

Endpoints

CATFT S3 Token service should be accessed using OAUTH tokens. The OAUTH token can be obtained by making an API request to CATFIP (CAT Enterprise Authentication Service) using the endpoints defined below.

The CATFIP API POST request should be made using Basic Auth method (username and password provided by FINRA CAT) but all subsequent requests to CATFT S3 Token service APIs should be made using the obtained OAUTH token passed as a header (see CATFT API section for details). The OAUTH token is valid for up to 12 hours.

CATFIP endpoints for retrieval of OAUTH Tokens are as follows:

Environment	Endpoint
UAT (CT)	https://ews-ct.fip.catnms.com/fip/rest/ews/oauth2/access_token
Production	https://ews.fip.catnms.com/fip/rest/ews/oauth2/access_token
Disaster Recovery (CT/Prod Mirror)	https://ews-ct.fip.dr.catnms.com/fip/rest/ews/oauth2/access_token

Header	Required (Y/N)	Default	Example
Authorization	Y	N/A	Authorization: Basic base64-encoded(username: password)
Accept	N	application/json	application/json

Sample CATFIP API call:
curl --location --request POST 'https://ews-ct.fip.catnms.com/fip/rest/ews/oauth2/access_token?grant_type=client_credentials' --header 'Authorization: Basic XXXXXXXX'

Sample response from CATFIP API:
OAUTH Token is the value of “access_token” key in the JSON response.

{
 "access_token" : "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
 "expires_in" : "43170",
 "scope" : "any",

"token_type" : "Bearer"
}

CATFT endpoints for retrieval of S3 tokens are as follows:

Environment / Product Type	Endpoint
UAT (CT)	
Equities	https://filetransfer.ct.catnms.com/S3TransferTokens/{XXXXX}/cat-equities
Options	https://filetransfer.ct.catnms.com/S3TransferTokens/{XXXXX}/cat-options
Production	
Equities	https://filetransfer.catnms.com/S3TransferTokens/{XXXXX}/cat-equities
Options	https://filetransfer.catnms.com/S3TransferTokens/{XXXXX}/cat-options
Disaster Recovery (CT/Prod Mirror)	
Equities	https://filetransfer.prodmirror.ct.dr.catnms.com/S3TransferTokens/{XXXXX}/cat-equities
Options	https://filetransfer.prodmirror.ct.dr.catnms.com/S3TransferTokens/{XXXXX}/cat-options

{XXXXX} is the id associated with the EWS account.

S3 Buckets

The S3 buckets and paths for submitting and retrieving data are as follows:

Environment / Product Type	Submission	Feedback
UAT (CT)		
S3 Buckets	**3275-9867-7452-eft**	**3275-9867-7452-eft-feedback**
Equities Paths	3275-9867-7452-eft/{XXXXX}/cat-equities/in	3275-9867-7452-eft-feedback/{XXXXX}/cat-equities/out
Options Paths	3275-9867-7452-eft/{XXXXX}/cat-options/in	3275-9867-7452-eft-feedback/{XXXXX}/cat-options/out
Production		
S3 Buckets	**4145-5486-2873-eft**	**4145-5486-2873-eft-feedback**
Equities Paths	4145-5486-2873-eft/{XXXXX}/cat-equities/in	4145-5486-2873-eft-feedback/{XXXXX}/cat-equities/out
Options Paths	4145-5486-2873-eft/{XXXXX}/cat-options/in	4145-5486-2873-eft-feedback/{XXXXX}/cat-options/out
Disaster Recovery (CT/Prod Mirror)		
S3 Buckets	**3275-9867-7452-eft-pm-us-east-2**	**3275-9867-7452-eft-feedback-pm-us-east-2**
Equities Paths	3275-9867-7452-eft-pm-us-east-2/{XXXXX}/cat-equities/in	3275-9867-7452-eft-feedback-pm-us-east-2/{XXXXX}/cat-equities/out

Options Paths	3275-9867-7452-eft-pm-us-east-2/{XXXXX}/cat-options/in	3275-9867-7452-eft-feedback-pm-us-east-2k/{XXXXX}/cat-options/out

{XXXX} is the id associated with the EWS account.

If the reporter name is provided in the file name, the feedback will be made available in either the equties path or options path, as appropriate for the named reporter.

If the reporter name is not provided in the file name and only one feedback path exists (either equities or options, but not both), the feedback will be available in that path.

If the reporter name is not provided in the file name and both equities and options paths exist, the feedback will be available in the options path at cat-options/out/YYYYMMDD/otherErrors.

CATFT API

The below details out the HTTP GET method to retrieve tokens against the CATFT (file transfer) service. The following header parameters should be passed when making the REST API call:

Header	Required (Y/N)	Default	Example
Authorization	Y	N/A	Authorization: Basic base64-encoded(username: password)
Accept	N	application/json	application/json, application/xml

* Org_id and the username - password to use the token service will be issued by FINRA CAT.

Sample response from the CATFT token service for an options SRO:


```
{
  "region": "us-east-1",
  "sessionName": "<session-name-detail>",
  "readPaths": [
    "4145-5486-2873-eft-feedback/<OrgID>/cat-options/out"
  ],
  "writePaths": [
    "4145-5486-2873-eft/<OrgID>/cat-options/in"
  ],
  "credentials": {
    "accessKeyId": "ABCDXXXXXXXXXXXXXXXXX",
    "secretAccessKey": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
    "sessionToken":
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX+==",
    "expiration": "2019-06-10T17:42:59.000-04:00"
  }
}
```

i. Set the received token as environment/session settings in the application (LINUX example below)

```
export AWS_ACCESS_KEY_ID=ABCDXXXXXXXXXXXXXXXXX
export AWS_SECRET_ACCESS_KEY=XXXXXXXXXXXXXXXXXXXXXXXXXXX
export AWS_SESSION_TOKEN=
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX+==
```

Sample response from the CATFT token service for an equities SRO:

```
{
  "region": "us-east-1",
  "sessionName": "<session-name-detail>",
  "readPaths": [
    "4145-5486-2873-eft-feedback/<OrgID>/cat-equities/out"
  ],
  "writePaths": [
    "4145-5486-2873-eft/<OrgID>/cat-equities/in"
  ],
  "credentials": {
    "accessKeyId": "ABCDXXXXXXXXXXXXXXXXX",
    "secretAccessKey": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
    "sessionToken":
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX+==",
    "expiration": "2019-06-10T17:42:59.000-04:00"
  }
}
```

ii. Set the received token as environment/session settings in the application (LINUX example below)

```
export AWS_ACCESS_KEY_ID=ABCDXXXXXXXXXXXXXXXXX
export AWS_SECRET_ACCESS_KEY=XXXXXXXXXXXXXXXXXXXXXXXXXXX
export AWS_SESSION_TOKEN=
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX+==
```

Note: The session token is valid for 3 hours, so perform the file transfer steps before token expires. Request and use a new token for subsequent file transfers.

Appendix H. Plan Processor Best Practices

This section contains a bulleted list of best practices for Plan Participants.

- Use a retry loop with a minimum of three attempts when submitting to or pulling from a FINRA CAT S3 location.
- Re-use valid and non-expired AWS S3 STS tokens whenever possible instead of using a new STS token per file.
- The AWS S3 STS token is temporary and will expire within an in 3 hours. For larger file uploads, refresh the token in 2 hours and 45 minutes to ensure there are no gaps

Appendix I. Historical Summary of Document Revisions

Version	Date	Author	Description
1.0	5/14/2017	Thesys CAT	Initial release.
1.1	6/2/2017	Thesys CAT	Incorporates feedback from version 1.0. • Various minor changes to correct typos, and make clarifications. • Sale Condition - Added the Supplemental Trade Event to provide a way for sale condition to be reported independently of the trade/fill event itself. In addition, the saleCondition in all the trade/fill events was marked as conditional. • Changed "style" to "exerciseStyle" for clarity • Changed timestamp format from UTC to Eastern (kept alternative timestamp format). • sequenceNumber changed from Required to Conditional • result and resultTimestamp changed from Required to Optional • Removed price from trade break event. Clarified definition of quantity in trade break event to allow for partial trade break. • Made buy/sell details on a trade correction optional - for simpler cases where only the price/qty are changed • Added executionTimestamp and reason as optional fields to trade correction events. • Fixed some Message Type typos and mismatches between tables. • Fixed inconsistent use of cancelReason and cancelReasonCode so all uses reference cancelReason. • Changed clearingFirm in stock leg from a validated MemberAlias to a free form Text(10) - as explained by SRO this field is received in the order from the BD and is passed thru to the firm executing the stock leg - there is no validation of this field. Also, changed to be optional. • exchOriginCode removed from complex option stock leg events • timeInForce, handlingInstructions, and orderAttributes added as conditional fields for complex option order modify event • liquidityCode is optional for option trades because some option exchanges do not track and report add/remove of liquidity. • Stock Leg Fill Event - renamed tradeID to fillID; removed quoteID; changed orderID to required; clearingFirm changes as mentioned above; clearingNumber is now optional • Post Trade Allocation - added optional fields as requested: openCloseIndicator, exchOriginCode, mktMkrSubAccount, reason • Upload directory will be the date for the events being reported • leavesQty in side details is not required when used in conjunction with a trade correction • cmtaFirm and mktMkrSubAccount are now conditional rather than optional • Modified Events - optional fields changed from optional to conditional since they are required if their value changes, and is more consistent with the definition of conditional than optional.

Version	Date	Author	Description
			• Substantial updates to data dictionary, including additions to orderType, executionCodes, handlingInstructions, and orderAttributes based on SRO feedback.
1.2	6/20/2017	Thesys CAT	• Minor changes to correct typos and add clarification • Data Dictionary - reformat; address typos and inconsistencies • Add ETF to issueType; add issueType to examples • Update JSON/CSV schema • Clarified orderID for option cancel and stock leg fill • Supplemental Trade Event - side is conditional on fillID • Clarifications in feedback section • Updated tables for FINRA reporting formats: sections 6.3, C.4, and D
1.3	7/6/2017	Thesys CAT	• aliases were overloaded - separated into memberAliases and symbolAliases • Clarify Inactive status for member dictionary • Add Asian and Cliquet to option settlement • Add definition of receipt time • Add symbol and optionID to the Note Event • Option trades may not have quoteID/orderID on one or both sides of a trade • Provide JSON field names for metadata file • Call out single-line restrictions on JSON/CSV files • Clarification and examples for JSON/CSV schema and conversions • Describe the Symbol Master upload file • Updated details and diagrams for connectivity changes • Clarify definition of Record Index for feedback and correction files • Add CBOE Note Event details • Clarify support for FLEX PCT trades • Defined values for ParticipantID/ExchangeID
1.5	12/07/2017	Thesys CAT	• Optionally allow space as separator in Timestamp • XTIME requires Timestamp • Add "type" field to Metadata • Update data dictionary with SRO-assigned values • Define Symbol Alias data type • Increase length of companyName field • Add symbol market move scenarios • Corrections and clarifications to text and examples • add executionCodes to option side-trade details • Update descriptions for FINRA reported OTCBB and TRF • Add FINRA halt/resume • Clarified encoding for file submissions • Placed length limit of filename group • Increase length of fileID and origFileID for metadata

Version	Date	Author	Description
			• Add information about upcoming change in encryption process • Clarified format for hashes in metadata • Removed support for VPN access • Clarified SFTP upload procedures • Add "final" stage for file processing • Provide fileName instead of fileID for certain integrity failures • Clarification for cancelQty • Added cancelReason values for BOX, MIAX, Pearl, and CHX • Added definedNoteData values for NYSE • Added exchOriginCode values for NYSE, Bats, MIAX, and Pearl • Added executionCodes values for BOX, MIAX, CHX, and NYSE • Added general handlingInstructions, and specific ones for BOX, CHX, and NYSE, • Added liquidityCode values to support extended codes for NYSE • Added noteType values for NYSE • Added/Updated orderAttributes values for BATS, BOX, CHX, and NYSE • Added general orderType values AMPEG, LOO, MOO, MDPEG, MMPEG, RTPEG, SOL and specific values of CHX and NYSE • Changed Participant ID values for NYSE National and NYSE American • Added CrossExempt to side values • Added general timeInForce values AOK, CLO, GTX, OPG, REG, WCO and specific values for CHX • Clarified the delivery timeline for the file submission functionalities via Reporter Portal • Update FINRA OTCBB/TRF field definitions • Restrict correction records to the original fileID • Provide full equity master file to participants • Define encoding as ISO-8859-1 • Clarify underlyingType mappings • PTA event: add quoteID; clarify quoteID/orderID fields • Support complex orders in option restatement • Clarify executingBroker definition • Redefine the GROUP filename component • Indicate when finished sending a batch of files • Add complexOptionID to leg events • quoteID globally unique by reporter/date/optionID/quoteID • New upload/encryption process • Clarify initiator field definition • Modified events now require full state of order • Modify and clarify file submission process • Update Participant ID definitions
1.6	2/16/2018	Thesys CAT	• Add lifecycle keys for each event • New events: Order Adjusted, Option Order Adjusted, Complex Order Adjusted, Stock Leg Adjusted

Version	Date	Author	Description
			• Unified and clarified definitions for originalOrderID in modified, adjusted, and restatement events • Remove confusing text about a missing or empty value for the session field being used as a default value. • Updated corporate action reporting formats specified in Appendix C. • File ID no longer required in .meta file, and origFileNumber replaces origFileId for file replacement and corrections. • Reverting to the original specification, regarding the .final file. Based on SRO feedback, in version 1.7 of the input spec changes will be made to simplify the automation of file submission from the SRO perspective. • Clarify NBBO values when the NBBO may be unavailable • Ease restriction on routingFirm so it can be any text string, not just a Member Alias. • Clarify what is submitted for both JSON and CSV formats when a data field is not reported. • Correct events which were missing fields displayPrice, displayQty, and leavesQty. • Added type as first column in FINRA OTC corporate actions, TRF, OTCBB, and Halt/Resume records. • Changed type from Numeric to Unsigned in FINRA TRF and OTCBB events. • Increased max length for some text fields in daily events to make them consistent. • Time is a JSON Number
1.6.1		Thesys CAT	• Change max length of Symbol to 20. • Fix typo in NYSE Corporate Actions event. • Remove symbology and normalization feedback stages these are contained in the ingestion feedback. • Added CBOE executionCode FirmTradeTime. • Add isGloballyUnique to complex accepted event, and relax requirement on complexOptionID if the orderID is globally unique. • Add the file kinds NASDDaily, BATSDaily, NYSEDaily, and FINRADaily to the file submission process. These file kinds subsume Halt/Resume and Corporate Actions. • Add clarification of semantics of a successful file replacement.
1.7	07/24/2018	Thesys CAT	• Updates to per-SRO member dictionary values • Added member field to explicitly identify the member on orders and trades. • Clarified requirement for marking ISO orders in handlingInstruction • Added sequence number subsystem • Change routingFirm to routingParty for clarity of intent • Add Internal Route events • Add Bulk Print Event • Clarify field requirements

Version	Date	Author	Description
			• Make fields conditional regarding complex options and option legs • Remove NASD TRF • Added file submission schedule • Add refTradeID to trade correction events • Add display Qty/Price to quote events • Remove executingBroker • Add floorBroker • beginDate is optional in the expected field for symbol master updates • Update type info for amount and amountCode in NASD daily records
1.7.1	09/09/2018	Thesys CAT	• Update symbol master management • ASE is to be used only for adding a new symbol • USE is to be used for only updating fields (no longer can be used for transfer) • SMRST is for restating and/or verifying an existing symbol • SMXFR is for transferring a symbol to a new listing participant • Update appendix E (symbol master transfer topics)
1.7.2	3/6/2019	CAT NMS, LLC	• Update encryption requirements • Change connectivity requirements from SFTP to S3 Upload • Add Disaster Recovery information • Add MIAX Emerald options exchange specifications • Add appendix G with sample transmission of Participant files to S3 buckets

2.0.0	5/10/2019	FINRA CAT	• General Format Modifications
			    ◆ Table numbers added
			    ◆ Font changes
			    ◆ Data dictionary changed from list to table format
			    ◆ Typographical errors corrected
			    ◆ Revised grammar as necessary
			• File compression has been limited to BZIP2 (.bz2)
			• Equities Submissions specifications have been greyed out.
			• PP SLA Requirements have been updated with new options files submission times
			• Data flow Architecture diagram updated to show the ability of PP's to pull feedback files from the Plan Processor AWS S3 location
			• Connectivity section updated to show new architecture of mandated S3 Direct Links (required by Nov.)
			• Physical locations of the feedback subdirectories identified for each Plan Reporter ID
			• Added AWS S3 Direct Download Process
			• Disaster Recovery Information updated.
			• Feedback and corrections flow chart for visualization of the process has been added.

Version	Date	Author	Description
			• Physical locations of the feedback subdirectories identified for each CAT Reporter • FieldName and FieldValue have been added to the feedback JSON format. • Error Codes for the correction feedback loop added in Appendix B • Data Dictionary Updates • Added Plan Participant best practices appendix
2.0.0 Enhanced	6/7/2019	FINRA CAT	• Corrected data type for name/value pair to add JSON object to match verbiage of name/value description in section 1.4.1 • Marked equity event sections 4.1 (EOA event), 4.2 (EOR event) and 4.8 (EOF event) with a black font since they are applicable for the June release • Section 5.2.2.3 definition for OSLM (option stock leg modified) event was changed to swap place the seqNumber attribute ahead of the seqNumSub attribute • Enhanced OT record samples to include Side Trade Details in Section 8.3.1 • Enhanced OT record samples to include Side Trade Details in Section 8.4.1 • In section 9.1.2 the compressedHash field description was changed to be consistent with the Include Key column which mandates that this field is populated. • Section 10.1 now articulates that all feedback files will be compressed using bz2 • Modify section 10 JSON feedback examples to be in sync with document verbiage and added more descriptive language for feedback files. • Modify document in section 10 to eliminate statements indicating that the entire file will be rejected if it contains an invalid message type • Sections 6, 10.9.1 and 10.10 have been marked with a grey font since they are not applicable to the June release • Added new error codes in Appendix B • Appendix F – Data Dictionary ◆ Add new value 'd' for orderAttributes name 'REJA' for Cboe non legacy options exchanges ◆ Change data type for cancelReason to Choice from text(255) ◆ Change data type of the orderAttributes value for name NBBOProtection to Boolean from choice for Cboe (C1 Legacy) ◆ Add new values for definedNoteData field for name AuctionType for Cboe (C1 Legacy) ◆ Add new values for oderAttributes names AckSubLiquidity and RESTA for Cboe non legacy options exchanges ◆ Add new value for handlingInstructions name TifMod for Cboe non legacy options exchanges ◆ Divide ExecutionCode Attribute name/values between Cboe non legacy options exchanges and Cboe (C1 Legacy)

Version	Date	Author	Description
			◆ Added valid temporary name value pairs for executionCodes, handlingInstructions, and orderAttributes to support back processing data received from 3/29/2019 – 6/21/2019
2.1.0	9/24/2019	FINRA CAT	• Section 4.2: Remove duplicative rows from Table 20, which describes the Equity Order Route event type • Removed section 9.3 with obsolete diagram of token exchange • Section 9.5: Update connectivity section to show private line connection details • Section 10.11.1 Feedback and Correction: Enhanced the description to state a reference data error can only be corrected by resubmitting the entire file after correcting the error • Section 10.11.1 Feedback and Correction: Enhance the correction processing section to state that the record offset in the feedback file for correction processing will reference the original file and not the correction file. • Appendix F: Data Dictionary modifications • Appendix G: Update for utilizing CATFT (fileX) for token retrieval and file transfer
3.0.0	11/19/2019	FINRA CAT	• Section 4: Add routedOrderId to EOM, EOJ. Added routedOrderId to side details on EOT and ETC. These attributes were added to facilitate equity linkage discovery • Section 4.15: Added new link route keys for EOM, EOJ, EOT and ETC events • Section 5: Add routedOrderID to OOM, OCOM, OOJ, OCOJ, OT and OTC option events. All of these attributes were added to events to facilitate option linkage discovery • Section 5.6 – Added new keys for Cross order and order route • Section 7 – Added examples for stock events with routedOrderId • Section 8 – Added examples for option events with routedOrderId • Section 9.1.2: Change to mandate isKindDone is populated with “true” after transmission of a fileKind is complete for the trade date. • Section 10: Changed the directory structure for feedback files • Section 10.9.3: Added to demonstrate feedback for Intra Exchange Linkage Discovery phase • Appendix B Error Codes: Added error codes for Intra Exchange Linkage Discovery phase • Appendix F Data Dictionary: Update orderAttributes to include the pairedOrderId to facilitate linkage for cross orders. Update to data dictionary for cancelReason field and add orderType values for IEX. Updates to Cboe values due to migration to Bats technology • Fixed typo in appendix G – CATFT Token Service instructions and examples • General verbiage and grammatical corrections
3.0.1	2/25/2019	FINRA CAT	• Removed optnId from cross order key for OOA and OOM events • Section 8.4 page 174: Fixed typo in example for OT event with routedOrderId for partially executed away trade

Version	Date	Author	Description
			• Appendix D FINRA Trade Reporting Facility (TRF) Fields: Added Related Market Center Id for MIAX PEARL Equities • Appendix F Data Dictionary: Enhance the orderAttributes definition for pairedOrderId to state the following: The Paired Order ID must uniquely identify the paired orders within the Trade Date and Exchange • Appendix F Data Dictionary: Added new order attribute name value pair for IEX for AIQ (Anti-Internalization Qualifier) • Appendix F Data Dictionary: Added new Plan Participant ID for MIAX PEARL Equities • Appendix F Data Dictionary: Removed orderAttribute Auction type 'c' for Cboe • Appendix F Data Dictionary: Removed legacy origin codes of 'P' and 'Y' from legacy Cboe possible exchangeOriginCode fields
3.1.0	3/10/2020 – 4/15/2020	FINRA CAT	*The following changes were presented to TWG on 3/13:* • Marked sections pertaining to equities with a black font from grey font in anticipation of on boarding LTSE and BSTX equity exchanges • Added Cross Order linkage key to EOA, EOM, and EOJ events • Section 10: Added clarification that if an entire file is rejected because it exceeded the ten percent threshold, then it must be corrected with a replacement file, not a correction file • Appendix F: Added BSTX – Boston Security Token Exchange as a Participant Id • Appendix G: Added new landing directories for equities exchanges *The following changes were presented to TWG on 4/16:* • Updated description of OOM, OOJ, EOM, and EOJ to clarify. • Updated definition of 'quantity' on OOM, OOJ, EOM, and EOJ to clarify. • Appendix F: Updated orderAttributes to add REJA name/value pair for LTSE for Cboe equity exchanges. Added handlingInstructions to add LTSE for NASDAQ equities.
3.1.0-r1	05/29/2020 – 06/05/2020	FINRA CAT	• Appendix F: Added executionCodes and orderAttributes values to support introduction of new Cboe Delta Adjust at Close order type. • Appendix F: Added handlingInstructions allowed value of 'e' for Cboe (Midpoint Discretionary Order with Quote Depletion Protection) • Appendix F: Added handlingInstructions allowed value of 'CUBEAUCS' for NYSE Options (Solicitation CUBE) • Appendix G: Updated format for clarity • §2.3 and §9.2 updated to include clarifications on the submission of the Options Dictionary containing products not included in OCC data. • §4 Equities Route Events (EOR, ECR, and EMR) and §5 Options Route Events (OOR, OOMR, OOCR) updated to remove session from event Route Link Keys as the session is not provided by Industry Members.
3.2.0	7/10/2020	FINRA	*Administrative updates:*

Version	Date	Author	Description
		CAT	• Moved Change Log contents prior to version 3.1.0 to Appendix I (change NOT tracked) • Renamed Change Log to "Summary of Document Revisions" to reduce confusion between document changes and Change Requests • Began reformatting of Appendix F: Data Dictionary (changes NOT tracked) • Updated XTIME in examples to reflect full timestamp format • Added OCOA and OCIR to events for exchOriginCode in Data Dictionary *Spec updates:* • §4 Events for Stock Changes; Appendix F: Add routingParty and session to EOM and EOJ events • §5 Events for Options Exchanges: Added new Floor Broker Events (Cboe) • §10.9: Intra Exchange Order Event Feedback ○ Moved to §10.10 ○ Updated to include Intervenue linkage feedback and TRF Trade linkage feedback • Appendix B: Added B.2.2 (Intervenue feedback error codes) and B.2.3 (Trade Linkage feedback error codes) • Appendix F: Data Dictionary: ○ Added orderAttributes and executionCodes values for LTSE ○ Updated definitions of routingParty and session to clarify use in modified and adjusted events • Corrected quoteID to remove Stock Leg Fill event, which does not include the quoteID field
3.2.0-r1	7/17/2020	FINRA CAT	*Administrative updates:* • Continued reformatting of Appendix F: Data Dictionary (changes NOT tracked) • Updated description of session for EOR to remove reference to matching the value reporting by the routing firm (IM do not provide session) • Added OOJ event for complexOrderID and complexOptionId in Data Dictionary *Spec updates:* • Appendix F: Data Dictionary: ○ Added cancelReason, handlingInstructions, and orderAttributes values for MEMX ○ Added 'DerivedOrderTraded' cancelReason for MIAX and MIAX Emerald ○ Added cancelReason, handlingInstructions, liquidityCode, orderAttributes, and orderType values for MIAX PEARL Equities ○ Appendix G updated to clarify placement of feedback if a reporter is not identified in the file name and the reporter has both equities and options buckets
3.2.1	8/7/2020	FINRA CAT	*Administrative updates:*

Version	Date	Author	Description

			• Updated §10.1 to provide distinct paths for Options feedback vs. Equities feedback; also corrected types in the path originally provided
			Spec updates:
			• Appdenix F: added MEMX to Participant ID list.
			• Introduced Equity Market Maker (EMM) reporting; updated the following sections:
			  ○ NEW! §2.4 Market Maker Information
			  ○ §9 File Submission Process
			  ○ NEW! §10.8 Feedback and Corrections for Market Maker Dictionary
			  ○ §10.13 Corrections
			  ○ Appendix B.1: Data Ingestion Errors
			  ○ Appendix B.3: Error Prefix Defintion
			  ○ Appendix F: Data Dictionary
			    ▪ Added: definedMMDEData, marketMakerStatus, marketMakerType, statusTime
			    ▪ Updated: marketMaker, type

3.2.1-r1	8/31/2020	FINRA CAT	*Spec updates:* • Appendix F: Added new values for NOBO in cancelReasons, executionCodes, and handlingInstructions

3.2.2	9/25/2020	FINRA CAT	*Administrative updates:*
			• Updated two references to linkageFailureFileCount in §10.11.3 to linkageErrorFileCount (as defined in §10.11.2.1)
			• Moved pre-3.2.0 changes from the main change log to the Appendix I historical change log *(changes not tracked)*
			Spec updates:
			• To support intervenue linkage (firm-to-exchange, firm-to-TRF):
			  ○ Appdenix B: Added intervenue link errors for destination did not match.
			  ○ Appendix B: Repurposed OE.TRADELNK error codes 4004 and 5005, previously identified as a mismatched eventTimestampe, to use for mistmatched marketCenterId.
			  ○ Appendix B: Added OE.TRADELNK error codes 4010 and 5011 for reporting or contra IMID cannot be found.
			• Appendix F: Added 'SYS' `timeInForce` code for LTSE

3.2.2-r1	10/21/2020	FINRA CAT	*Spec updates:* • Appendix F: Added new `cancelReason` name/value pairs for MEMX. • Appendix F: Added new name/value pairs for Cboe Position Compression Cross and Related Futures Cross in `executionCodes` > `SUBLIQ` and `orderAttributes` > `AuctionType` and `executable`.
3.2.2-r2	11/10/2020	FINRA CAT	*Administrative updates:* • Updated §10.11.3 for Intravenue Linkage example to include the Linkage Key in the description field. This is the current

Version	Date	Author	Description
			behaviour for Intravenue Linkage feedback but was not previously documented here. *Spec updates:* • Updated §10.11.2.2.1 Intravenue and Intervenue Linking element 1.n.4 to indicate that the Linkage Key is provided as part of the description field. • Updated §10.11.3 for Intervenue Linkage example to include the Linkage Key in the description field.
3.2.2-r3	11/18/2020	FINRA CAT	*Spec updates:* • Appendix F: Added new `orderAtrribute` of `R` for MEMX Retail Orders.
3.2.2-r4	12/2/2020	FINRA CAT	*Spec updates:* • Appendix F: Added new `cancelReason` of `PEARLEQ_0012` for MIAX PEARL Equities. • Appendix F: Added new `orderAttributes` of `FBT` (Floor Broker Trade) for all exchanges.
3.2.2-r5	1/11/2021	FINRA CAT	*Spec updates:* • Appendix F: Added new `cancelReason` of `MIAMI_0059` for MIAX.

4.0.0	2/22/2021	FINRA CAT	********Round 1 of Release 7 Edits********
			Spec Updates:
			Changes for Release 7 to support transition of data submission by equities exchanges from FINRA RSA feed format to CAT specified format, including:
			• Removed Bulk Print event type and all corresponding references.
			• Added `side` field to EOM, OOM, and OOJ event types; clarified definition of `side` field for EOJ.
			• New name/value pairs for the submission of single-priced auction trades such as openings, re-openings and closings
			• Side field added to modify events.
			• Changed several elements for `sideDetails` for equities from required to conditional.
			• EOF contraClearingNumber from required to optional.
			• EOT sideDetails 'side' from required to conditional.
			• Appendix F:
			  o Added `executionCodes` Allowed Values/Name Value Pairs for `NonMediaTrade`, `BulkTradeType`, `BulkTradeID`.
			  o Updated description of `side` to include newly relevant events.
			  o Remove `type` of BulkPrint

4.0.0	2/22/2021	FINRA CAT	********Round 2 of Release 7 Edits******** *Administrative Updates:* • Executive Summary, Introduction, and CAT Overview - revised to align with IM Spec and remove duplicate information. • Change Log - Moved pre-Release 7 items to Appendix I.

Version	Date	Author	Description

			• §1.4 Fundamental Data Types – moved Data Validation paragraph above Name Value Pairs section.
			• §9 Submission and §10 Feedback and Corrections – removed outdated references to web GUI/portal for uploading of data
			• §10 Feedback and Corrections – updated to clarify use of Replacement Files (see 10.10.2).
			• Appendix F: Data Dictionary - provided quick links via alphabet and to commonly used terms; updated format to clearing show the events lists for each field (not tracked).
			• Reconciled the following events with the Data Dictionary and made updates as applicable:
			  ○ Member Dictionary Entry (MDE)
			  ○ Option Series Dictionary Entry (OSDE)
			  ○ Complex Option Series Dictionary Entry (CODE)
			  ○ Market Maker Dictionary Entry (MMDE)
			  ○ Note (NOTE)
			  ○ Self-Help Declaration (SHD)
			  ○ Supplemental Trade Event (STE)
			  Included addition of the following to the Data Dictionary: groupID, ID, optionsSymbol, seqNumSub
			• Throughout - cleaned up references to web-based functionality that isn't provided by CAT.
			Spec Updates:
			• Revised throughout to reflect provision of Equity Symbol and Corporate Action Reference Data by FINRA:
			  ○ §2.2 Equity Symbols – ungreyed.
			  ○ §2.2.# - all 2.2 subsections remove *except* for CAT Symbol Master and Corporate Actions.
			  ○ §2.2 - updated to reflect provision of Equity Symbol and Corporate Action data by FINRA.
			  ○ §3.2 Symbology - ungreyed; edited to remove reference to Symbol Dictionary.
			  ○ §9 Submission Process - removed references to Symbol Master, Symbol Dictionary, and exchange-specific Corporate Action files where applicable.
			  ○ §10 Feedback and Corrections - greyed out Symbol Master content removed.
			  ○ §10 Feedback and Corrections - greyed out Symbol Dictionary content removed.
			  ○ Appendix C - removed exchange-specific Corporate Actions schemas.
			  ○ Appendix E - removed market move examples; these are obsolete given that equities symbol information will be provided by FINRA, which already takes market moves into consideration.
			  ○ Appendix F: Data Dictionary - removed the following terms that appeared in the removed Reference DataL listingParticipantId, issueType, beginDate, endDate, companyName, IPO, test, attributes, listedSymbol, symbolAlias.
			  ○ Throughout - updated description of 'symbol' to remove references to Symbol Dictionary.

<table>
<tr><th>Version</th><th>Date</th><th>Author</th><th>Description</th></tr>
<tr><td></td><td></td><td></td><td>
<li style="list-style:none">
Throughout - removed references to “Symbol Dictionary”, “Symbol Master”, and “Symbol Entry”.
Revised throughout to reflect FINRA Plan Participant reporting of TRF and OTC Halts data:
§6.1 TRF/ORF/ADF - ungreyed.
§6.1 FINRA TRF - added new TRF spec.
§6.3 OTC Halts - ungreyed.
§6.3 OTC Halts - updated.
§9 Submission Process - added FINRA Transactions (TRF) and OTCHalts file kinds where applicable
§10 Feedback and Corrections - ungreyed TRF/ORF/ADF content removed; updated as needed.
§10 Feedback and Corrections - added section for OTC Halts.
Appendix B: Errors Codes - error prefix list and ingestion error codes updated to reflect new TRF and OTC Halts file processing.
Appendix D - removed greyed out TRF spec.
Appendix F: Data Dictionary - added new fields for TRF and OTC Halts, including allowed values.
Per request from Plan Participants:
For EOT and OT, set routedOrderID to Optional.
For EOM and EOJ, clarified description of routedOrderID, routingParty, and session.
</td></tr>
<tr><td>4.0.0-r1</td><td>3/24/2021</td><td>FINRA CAT</td><td>
Administrative Updates:
Fix typo in Appenidix F: Data Dictionary <code>orderAttributes</code> > <code>PairedOrderID</code> to lowercase ‘<code>p</code>’.
Document updated to reconcile changes from versions 3.2.2-r4 and 3.2.2-r5. (Changes not tracked since they were previously approved.)
Spec Updates:
Updated §10.9.2.2.2. Table 96: Linkage Error Feedback <u>for Off-Exchange Trade Reports to reflect additional information to be provided for Off-Exchange events Linkage Error Feedback. (Effective June 1, 2021)</u>
Updated Appendix F: Data Dictionary to:
Add <code>childOrderID</code> to <code>orderAttributes</code> for all exchanges.
Add Allowed Values for NYSE Equities for: <code>capacity, handlingInstructions, orderAttributes, orderType</code>
Add Allowed Values for NASDAQ Equities for: <code>cancelReasons, handlingInstructions, liquidityCode, orderAttributes</code>
</td></tr>
<tr><td>4.0.0-r2</td><td>4/5/2021</td><td>FINRA CAT</td><td>
Spec Updates:
Updated Appendix F: Data Dictionary as follows:
For Cboe-BYX:
Added <code>orderAttributes</code> of <code>CrossTradeFlag</code> and <code>LockOrderForAuction</code>
Added new allowed value ‘<code>p</code>’ for <code>executionCodes</code> > <code>SUBLIQ</code>
</td></tr>
</table>

Version	Date	Author	Description
			For FINRA o Added allowed values for `finraTradeModifierSroCode`, `firmTradeModifierSroCode`, `tradeSourceCode` o Updated format of `finraTradeModifierThroughExemptTime` from Timestamp to Time For IEX: o Added new allowed values for `cancelReason`, `executionCodes`, `orderAttributes`, and `orderType` For NASDAQ: o Added allowed values for `cancelReason`, `definedMMDEData`, and `orderAttributes`
4.0.0-r3	4/20/2021	FINRA CAT	*Administrative Updates:* • Rewrote content of Section 10: Feedback and Corrections for consistency and clarity. • Update Section 9.6 Submission Feedback to remove duplicative content and refer to Section 10. • Updated NASDAQ `orderAttributes` of PegOffet to PEGOFFSET to match expected implementation. • Removed definitions of MPEG and PPEG in NYSE Equities `handlingInstructions`. *Spec Updates:* • Updated Section 10 Feedback and Corrections (specifically 10.8.1.1) to reflect submission of 'FCOR' record type. • Updated Appendix F: Data Dictionary to add 'FCOR' record type. • Updated Appendix F: Data Dictionary for NYSE Equities `orderAttributes` to add MMID.
4.0.0-r4	4/20/2021	FINRA CAT	*Administrative Updates:* • Corrected the following items that were introduced in previous versions of the spec: o Cboe `handlingInstructions`: `ExecInst` 'H' to 'h'. o NYSE Options `handlingInstructions` 'PNP' to 'PNP+'. (Note that PNP appears twice; PNP is still a valid value.) o Cboe `orderAttributes`: `ROUTESTRAT` 'N' to 'n'. o Removed duplicate NYSE Equities `orderAttributes`. *Spec Updates:* • Updated Appendix F: Data Dictionary for NYSE Equities `handlingInstructions` to add AOC and DIR and remove NALO. • Updated Appendix F: Data Dictionary to add `executionCodes` of `childOrderId` for all participants.
4.0.0-r5	5/4/2021	FINRA CAT	*Administrative Updates:* • *Spec Updates:* • Updated Appendix F: Data Dictionary as follows:

Version	Date	Author	Description
			o BOX `definedNoteData: ST` values of TraderCanceled and CanceledBySupervisor updated to include two ‘l’s to match implementation. Update approved by BOX. o Cboe `handlingInstructions: ExecInst` ‘h’, defined as Minimum Not Held, updated to ‘h’ Minimum and ‘1’ Not Held’. o Cboe `orderAttributes: AllowPriceSlide` – added line break between values ‘C’ and ‘K’. o IEX `orderType: RLPM` added. o NYSE Equities `executionCodes` – removed Auction, Close, and Open values per NYSE request.
4.0.0-r6	6/1/2021	FINRA CAT	*Spec Updates:* • FINRA Transaction event (TRF) `tradeStatusCode` changed to a Conditional field.
4.0.0-r7	6/16/2021	FINRA CAT	*Spec Updates:* • Updated Appendix F: Data Dictionary to add the following for NASDAQ Equities: o `orderAttributes` (**`OriginalChildOrderID`**) o `orderAttributes: CrossType` value (**`8`**) o `handlingInstructions` (**`ChildCancelReason`** and **`RSRV`**) o `liquidityCodes` (**`AfterHoursClose`**) o `timeInForce` (**`AHC`**) • Updated Appendix F: Data Dictionary to add the following for MIAX PEARL Equities: o `cancelReasons` (**`PEARLEQ_0013, 9002, and 9003`**)
4.1.0	6/1/2021	FINRA CAT	*Administrative Updates:* • Updated Table 1: Summary of Document Revisions to move items prior to 4.1.0 to Appendix I. *(change not tracked)* • Reconciled list of Options Events in Section 5. *Spec Updates for Plan Participant Release 7b:* • Added new Member Alias Detail Entry (MADE) event in Section 2.1 Member Information for reporting of ATSs and non-member firms. Updated Data Dictionary accordingly. • Added test flag to Option Series Dictionary Entry (OSDE). • Added `side` to OCOM and OCOJ events. • Added `routingParty` and `session` to OOM, OOJ, OCOM, OCOJ. • Added new Option Complex Order Route (OCOR) event for routing of a complex order to an external destination. *Spec Updates for Release CR-20 (Two-sided Option MM Quotes):* • Added `askQuoteID` and `originalAskQuoteID` to OQ event. Added `askQuoteID` to OQC event.
4.1.0-r1	6/15/2021	FINRA CAT	*Spec Updates for Plan Participant Release 7b:* • Updated Section 10.6 and Appendix B.1 to reflect conditional validations that will be enforced. Updated Section 10.7 and Appendix B.2 to reflect duplicate validations that will be enforced, including duplication of event data and linkage keys.

Version	Date	Author	Description
4.1.0-r2	7/14/2021	FINRA CAT	*Spec Updates for Release CR-21 (Plan Participant 24-Hour Trading):* • Updated Section 9.7 CAT Reporting Hours • Added Appendix D: CAT Date Definitions and Reporting Guidelines *Spec Updates for Release CR-## (Linkage of IM MOOT and Plan Participant OT):* • Appendix F: Added new `executionCodes` of `MOOTLINK`. • Appendix B: Added new codes for MOOTLINK linkage errors. *Spec Updates for Release CR-## (Trade Reversals):* • Appendix F: Added new `executionCodes` of `CORR, PRVRSL, REFTRADEID, REFTRDDATE,` and `RVRSL.` *Spec Updates for Cboe SUBLIQ* Appendix F: Added new `executionCode` > `SUBLIQ` value of `x` (Effective no later than 7/9/2021 in CT and 7/28/2021 in PROD.)
4.1.0-r3	7/28/2021	FINRA CAT	*Administrative Updates:* • Appendix F: Data Dictionary updated to clarify definition of `marketMaker`. *Spec Updates for Release CR-21 (Plan Participant 24-Hour Trading):* • Section 3.1 Timestamps and Sequence Numbers updated to reflect use of `cycleDate` for 24-hour trading. • Section 3.7 Common Events updated to add `cycleDate` for all common events. • Section 5 Events for Options Exchanges updated to add `cycleDate` to all Options events. • Section 5.7 Lifecycle Keys updated to add note about use of `cycleDate`. • Appedix F: Data Dictionary updated to clarify definition of `sequenceNumber`. *Spec Updates for PP Release 7b:* • Section 2.5 market Maker Information and 9.2 File Submission Schedule updated to change submission deadline of Market Maker Dictionary to T+1 @ 4:00 a.m. ET. • Section 10.8 Corrections, Deletions, and Replacements updated to indicate that full replacement files cannot be submitted after T+4 @ 8:00 a.m. ET for Order Events and FINRA transactions (TRF). *Spec Updates for NYSE Options on Pillar Trading Platform:* • Appendix F: Data Dictionary updated with new `definedNoteData`, `executionCodes`, `handlingInstructions`, and `orderAttributes`. *Spec Updates for Release CR-## (Linkage of IM MOOT and Plan Participant OT):* • Appendix B: Added/updated codes for MOOTLINK *side* intervenue linkage errors. **ADDITIONAL UPDATE ON 8/9/2021** Updates made in revision 4.1.0-r3 for the NYSE Pillar Migration inadvertently resulted in the use of `orderAttributess:Reserve` as both a Boolean and a Name/Value Pair for the NYSE Options Markets.

Version	Date	Author	Description
			Because this cannot be supported, FINRA CAT and NYSE agreed to the following approach: • Leave `Reserve` as a Boolean value for all NYSE Options markets (AMEROP and ARCAOP). • Remove `Reserve` as a Name/Value Pair for the NYSE Options markets. Add `PublishQuanity` as a Name/Value Pair for NYSE ARCAOP only, using the same definition provided for the Reserve Name/Value Pair. This allows ARCAOP to use *either* `Reserve` *or* `PublishQuanity` during the transition phase. `PublishQuantity` will be open to AMEROP at a later time when that market is really to migrate to Pillar.
4.1.0-r4	8/25/2021	FINRA CAT	*Administrative Updates:* • Section 3.3 updated to clarify that NBBO fields are optional on order leg events. • Updated Sections 5.2 and 5.2.1.2 to clarify the requirement for leg level order events. • Updated Section 5.2.1.2 to include special handling for QCC orders. *Spec Updates for Plan Participant Release 7b:* Appendix B updated to move the Duplicate Exchange/Firm Trade Key from OE.INTRAEXCHLNK.5010 to OE.INTERVENUELNK.6020.
4.1.0-r5	10/6/2021	FINRA CAT	*Spec Updates:* • `routingParty` updated throughout from Text (20) to Text (8) to closer align with the format of IM-related data used for linkage against the `routingParty` field. • Order of new fields on the following events have been modified to match implementation: OOM, OCOM, OOJ, OCOJ. • Added clarification to cycleDate throughout that the value should be between Event Date and T+1, inclusive. Appendix F: Data Dictionary updated to include `handlingInstruction` of XCTBL for BX, PSX and NSDQ, inadvertently omitted from prior specification. Removed duplicative `handlingInstruction` section for BX, PSX and NSDQ.
4.1.0-r6	10/19/2021	FINRA CAT	*Administrative Updates:* • Sections 1.4 and 9.1.3.1 updated to add clarification on submission of non-required fields at the end of a CSV record. *Spec Updates for NYSE Options on Pillar Trading Platform:* • Appendix F: Data Dictionary updated with new `orderAttributes` of `ClearTheBook`. *Spec Updates for Plan Participant Release 7b:* • Appendix B updated to change Intravenue Error code 5006 to 5011 and 5008 to 5012 (to eliminate conflict with IM error codes). *Spec Updates for NASDAQ:* • Added new values for `handlingInstructions:Display` *Spec Updates for IEX (added 10/8):* Added new values for `handlingInstructions` ('Reserve' and 'DisplayRange')
4.1.0-r7	12/1/2021	FINRA	*Administrative Updates:*

Version	Date	Author	Description

		CAT	*The following updates have been made to the Data Dictionary to correct errors and omissions from previous versions of the document:*
			• For `definedNoteData > ST:`
			  ○ BOX values 'TradeCanceled' and 'CanceledBySupervisor' update to 'TradeCancelled' and 'CancelledBySupervisor'
			• For `cancelReason:`
			  ○ BOX value 'CanceledBySupervisor' updated to 'CancelledBySupervisor'
			• For `handlingInstructions:`
			  ○ NADSAQ (BX, PSX, NSDQ) `ChildCancelReason` updated values 76, 77, 79, 82, 89, and 100 to 23-28, respectively; added values 1, 29-33
			  ○ NASDAQ (BX, PSX, NSDQ) `display` value 'Other' updated to 'OTHER'
			  ○ ARCAOP/AMEROP value 'FlexPCT' updated to 'FLEXPCT'
			  ○ Cboe `cancelReason` value 'FloorError' added
			  ○ Cboe `execInst` value '1' added
			• For `orderAttributes`:
			  ○ Cboe `ST` value 'Eliminated' added
			Spec Updates:
			• Updated `quoteId` on OQ event from Required to Conditional
			• Added `saleCondition` values 'u' and 'v'
			• Added information about OAUTH in Appendix G
			• Appendix B updated to change error code FILE.NAME.240 to INT.META.240
			• Added PEARLEQ `cancelReasons` values 'PEARLEQ_0014', 'PEARLEQ_0015', 'PEARLEQ_0109', 'PEARLEQ_0110', 'PEARLEQ_0111', and 'PEARLEQ_0112'
			• Added MEMX `handlingInstructions` values 'RML' and 'RMO'
			Added `orderAttributes` values 'replacedOrderDate' and 'replacedOrderID'

4.1.0-r8	12/15/2021	FINRA CAT	*Administrative Updates:* • Added clarification to Appendix G regarding OAUTH POST request The following updates have been made to the Data Dictionary to correct errors and omissions from previous versions of the document: • Update `handlingInstructions > tifMod` for Cboe Equities and Options to remove trading session times and point to documentation on the Cboe website. *(Approved with version 4.0.0-r1)* • Add Allowed value for FINRA for `contraReportingObligationFlag` *(Approved with version 4.0.0-r1)* *Spec Updates:* Added IEX `definedMMDEData` Name/Value pair of `MMRegistrationEvent`
4.1.0-r9	2/15/2022	FINRA CAT	*Spec Updates:*

Version	Date	Author	Description
			Added NYSE Equities `handlingInstructions` Name/Value pair of `DirectedTo_ATS` Added MIAX PEARLEQ `handlingInstructions` Name/Value pair of `RoutingStrategy`
4.1.0-r10	3/7/2022	FINRA CAT	*Spec Updates:* Added CBOE BYX `orderAttributes` > `MODR` value of 'p' for Periodic Auction
4.1.0-r11	3/18/2022	FINRA CAT	*Spec Updates for Plan Participant Reference Data Validations:* • Updated Section 10 throughout as necessary to reflect new Reference Data Validation processing stage and feedback generation • Added Section 10.7 documenting Reference Data Validation feedback format • Added Appendix Section B.2 documenting Reference Data Validation errors *Spec Updates for Plan Participant Conditional Data Validations:* • Added Error Code 2290 for *cycleDate* validation
4.1.0-r12	4/4/2022	FINRA CAT	*Administrative Updates:* • Removed all references to the submission of the OTCBB file kind and submission of OTCBB event data to CAT *Spec Updates for Plan Participant Reference Data Validations:* • Grayed out changes made in version 4.1.0-r11eference Data Validations will be implemented at a later date • Clarified Data Ingestion Conditional Validations in Appendix B1 *Spec Updates:* Added NASDAQ NOBO Request for Prism (RFP) cancelReason value of '1187' and handlingInstructions values of 'RfaId' and 'RfaInstruction'
4.1.0-r13	4/29/2022	FINRA CAT	*Administrative Updates:* • Updated format of event definition tables for consistency and usability (changes not tracked) • Moved Change Log content for Releases 4.1.0 through 4.1.0-r8 to Appendix I (changes not tracked) *Spec Updates:* • Updated Errors 2010, 2020, 2170, and 2180 to allow submission of 'zero' • Updated Warning 5005 to indicate that it will be retired effective June 15, 2022 • Added MIAX PEARL `cancelReason` value of 'PEARL_0038' • Added MIAX PEARLEQ `cancelReason` values of 'PEARLEQ_0113' through 'PEARLEQ_0116' Added CBOE `executionCodes > SUBLIQ` and `orderAttributes > AckSubLiquidity` value of 'G'
4.1.0-r14	7/8/2022	FINRA CAT	*Spec Updates:* • Updated definition of Error 2180 • Added `executionCode` value of 'FLOOR'