

## Strategy Studio: LimeExecutionHandler (Execution Handler)

LimeExecutionHandler is Strategy Studio's FIX interface to Lime Brokerage's Trading Server. Orders can be submitted to many equity and options venues, including several dark pools. The available order parameters are extensive and include many route-specific settings.

This execution handler may be selected by including the following line to the server's main configuration file:

- **EXECUTION\_HANDLER\_MODE=LimeExecutionHandler**

### Session Level Settings

Many of the session-level settings are specified in the file `execution_handler-conf/fix.config`. Notable settings include:

- **StartTime=HH:MM:SS**
- **EndTime=HH:MM:SS**
  - Required. Sets the window of time that the execution handler will attempt a connection, in UTC.
- **SocketConnectHost=host**
  - Required. Specifies the IP of the TS3 server for this session.
- **SocketConnectHost[n]=host**
  - Optional. Allows additional hosts to be specified for failover purposes. After connection failure, the next host is tried in a round-robin fashion.
- **ReconnectInterval=seconds**
  - Sets the wait period before a disconnected session attempts a reconnect.
- **SenderCompID=senderCompID**
  - Required. This corresponds to a Lime account and will be provided by Lime.
- **Username=username**
  - Required. This is the username for the Lime account.
- **Password=password**
  - Required. This is the password for the Lime account.
- **CancelAllOnDisconnect=true/false**
  - Required. Determines the behavior for open orders on disconnect. The recommended value is true.
- **StrategyStudioAccount=accountName**
  - Required. Sets the Strategy Studio account that will be associated with this session. This value in conjunction with OptionsEnabled must be unique for each session.
- **StrategyStudioFirm=firm**
  - Required. Sets the firm this session will be associated with for auto-marking purposes.
- **OptionsEnabled=true**
  - Optional. If true, routes options orders for StrategyStudioAccount through this session.
- **AutoMarkSide=true/false**
  - Optional. Determines whether new sell orders will automatically have their side marked based on the firm's position. The default value is false.
- **AttributeExternalOrdersTo=strategyName**
  - Optional. If a strategy named strategyName exists, external fills will be sent to this strategy. If tag 9050 ('client order data' field in Portal) contains an instance name the fill will be routed to that instance instead.

### Route Level Settings

Route level settings are specified in `execution_handler-conf/lime_route.config` with format:

#MarketCenter,RouteString,AllowRouting,LockedCrossedAction,RoutingInstructions,WashTradePrevention

- › **MarketCenter** is a string value that represents the Strategy Studio MarketCenterID, and the rest of the settings specify settings for orders routed to this MarketCenterID. A value of DEFAULT defines a catch-all route.
- › **RouteString** is the destination route, the value that will be sent in FIX tag 100.
- › **AllowRouting** sets a default value for whether the destination exchange should be allowed to reroute orders when there's a better REG NMS protected quote at another market center. Possible values are 'true', 'false', and 'none' (used if a route does not recognize a routing tag; the Lime FIX spec has a list of such routes).
- › **LockedCrossedAction** can be one of Reject, PriceAdjust, HideNotSlide, PriceAdjustRejectCrossed, SingleReprice, Blind, NoMidpointMatch, MultipleDisplayPriceSliding. This sets a default for the market center's action in the event an order is non-routable but the incoming order would lock or cross a REG NMS protected quote.
- › **RoutingInstructions** are route specific; the Lime FIX spec provides guidance on accepted values.
- › **WashTradePrevention** can be CancelNewest, CancelOldest, or CancelBoth.

### Custom Order Parameters

Common order properties are set using OrderParams's fields. Order properties that are not included in OrderParams can be set via OrderParams's custom\_params member, which is a std::vector<boost::any>. The values are expected to be of type std::pair<int, std::string>, with the first specifying the FIX tag number and the second specifying the string value for the FIX field. A full description of supported FIX tags is available in the Lime Brokerage FIX Manual available at:

<http://resources.limebrokerage.com/client/downloads/low-latency-trading/lime-fix-manual/LimeFIXManual.pdf>

OrderParams's custom\_params may be used on both new orders and cancel replaces. Note that custom\_params are not automatically cleared after submission of a new order; therefore, if the existing Order's OrderParams is used when sending a cancel replace, custom\_params should be cleared before submission.

### Lime ACS Configuration

Strategy Studio's automatic side marking requires Lime ACS to initialize start of day positions for the accounts within a trading unit. The connections to ACS are configured with execution\_handler-conf/lime\_acs.config by entries with format:

#Hostname,Username>Password,[Account[ | Account]...]

- › **Hostname**
  - Specifies the IP for Lime's ACS server.
- › **Username**
  - The Lime username for the ACS account.
- › **Password**
  - The password for the ACS account.
- › **Account**
  - An optional pipe-delimited list of the TS3 accounts associated with this ACS connection. If omitted, this connection will be used for all TS3 accounts.

Lime's TradingServer also has automatic side marking capabilities. It is recommended clients enable this on their accounts to avoid rejects that can occur if Strategy Studio's side determination conflicts with Lime's Trade Server rule (if for example in flight data has changed the appropriate side).