Tutorial: Order Trigger Bot

Introduction

Order Trigger Bots (Trigger Bots) are responsible for marking orders that satisfy the trigger condition, including:

- Trigger Market Orders -
- Stop Market and Take Profit
- Trigger Limit Orders -
- Stop Limit and Take Profit Limit

Trigger Bots receive a small compensation for each successfully marked order.

SeeKeepers & Decentralised Orderbook for a technical explanation of how the decentralised orderbook (DLOB) and matching incentives work.

Trigger Bots are similar toTutorial: Order Matching Bot in that they:

- also maintain a local copy of theKeepers & Decentralised Orderbook
- ;
- do not require the operator to manage collateral; and
- receive a small reward for performing their duties.

Getting Started

The reference implementation of a Trigger Bot is availablehere(opens in a new tab) .

Follow the instructions atKeeper Bots to set the required environment variables and make sure aClearingHouseUser is initialised.

Start the Trigger Bot:

```
yarn run

dev:trigger
```

Technical Explanation

1. Get nodes from the DLOB that are ready to be triggered

The DLOB implementation includes a method for getting orders ready to be triggered:

```
const

market

=

this . driftClient .getMarketAccounts ()[ 0 ]; // get a MarketAccount

const

oraclePriceData

=

this . driftClient .getOracleDataForMarket (marketIndex);

const

nodesToTrigger

=

this . dlob .findNodesToTrigger ( marketIndex , this . slotSubscriber .getSlot () , oraclePriceData .price );
```

2. Calltrigger_order

onDriftClient

const

user

=

this . userMap .get (nodeToTrigger . node . userAccount .toString ()); const

txSig

=

await

this . driftClient .triggerOrder (nodeToTrigger . node .userAccount , user .getUserAccount () , nodeToTrigger . node .order);

Tutorial: Order Matching Bot Tutorial: Order Liquidation Bot