

# Treasury research de-quant platform hackathon docs

Team#16 - Treasury research - de-quant platform

## Introduction

*Treasury research is a decentralized quant trading platform.* Treasury research built upon highly secured and trustless data flow architecture, which allows professional quants/traders to create advanced trading strategies with **seamless on-chain / off-chain data access**, **trustworthy decentralized off-chain calculation** and **cross-chain trading execution**.

## Direction

Quant-trading, Decentralized data market, Off-chain computation, Cross-chain interaction, theGraph, Chainlink, MCDEX, Arbitrum

## What problem do we solve?

As the DeFi market grows, there is significant alpha produced on-chain for professional programmable traders (quants) to exploit. Decentralized quants is a potential market for the next couple years. However, there are some barriers before professional quants are able to monetize their trading strategies on-chain:

1. **Hard to explore and use useful on-chain signals seamlessly**, to get historical on-chain data is not hard, but it is very hard to get real-time signals, especially when signals require aggregation on the cross-blockchain.
2. **Quant finds it is hard to quickly calculate and execute their trading strategies in a trustworthy manner**. Most current complex trading strategies cannot be implemented on-chain, because not only cost and performance on blockchain is low, but also not trustworthy executable environment.
3. **Segmentation of blockchains prevents executable strategies to use data on different blockchains and access to a variety of trading platforms on different blockchains.**

## Product design highlights

Data accessibility and seamlessly - To allow quants access to universal data factors on-chain / off-chain in a seamless manner, Treasury research uses **theGraph to query on-chain data**, such that executable quants program can query fresh data directly from blockchain. Also, TR uses **Chainlink oracle to handle a variety of data queries and post aggregation**, such that the entire data flow is entirely implemented in a trustworthy manner.

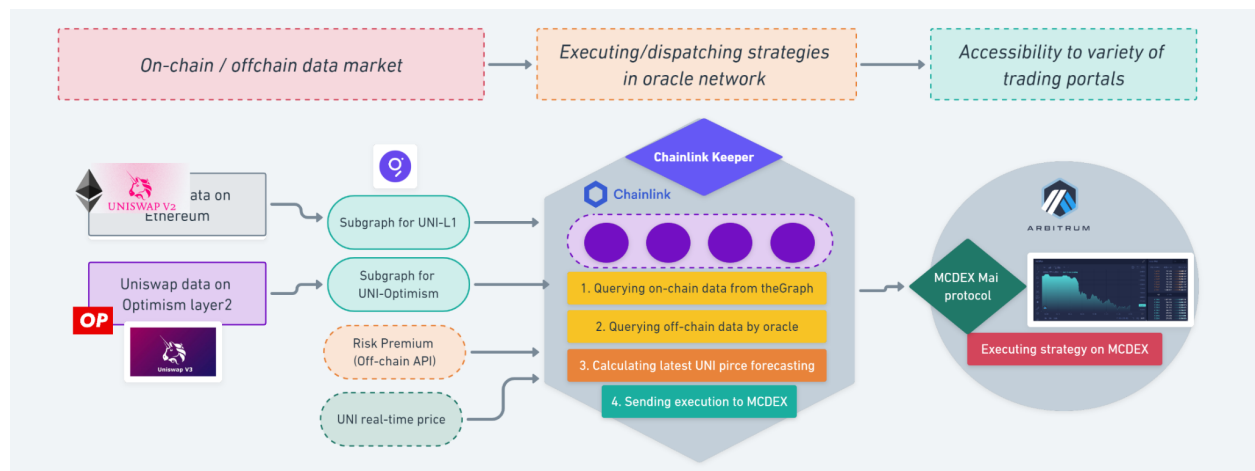
Another thing worth to highlight, Treasury research uses **Chainlink oracle as a trading strategy executable environment** to consolidate different types of row data and signals, and **make executable calculation** and, eventually, **fulfill the execution on trading platform MCDEX**.

With **MCDEX's Mai protocol**, **we can easily fulfill quants' execution strategy on-chain instantly and securely**.

### In Treasury research, we are empowered by

- **The Graph's** versatility of on-chain indexing and query;
- **Chainlink Oracle** capability of on-chain/off-chain data access and off-chain computation, additionally, we also benefited by Chainlink's accessibility on different blockchains.
- **With MCDEX**, Quant strategy can be easily executed in MCDEX derivatives platform instantly and securely, **MCDEX** is building upon Arbitrum, which is a layer 2 solution with high performance and low cost.

## Technical spec & design architecture

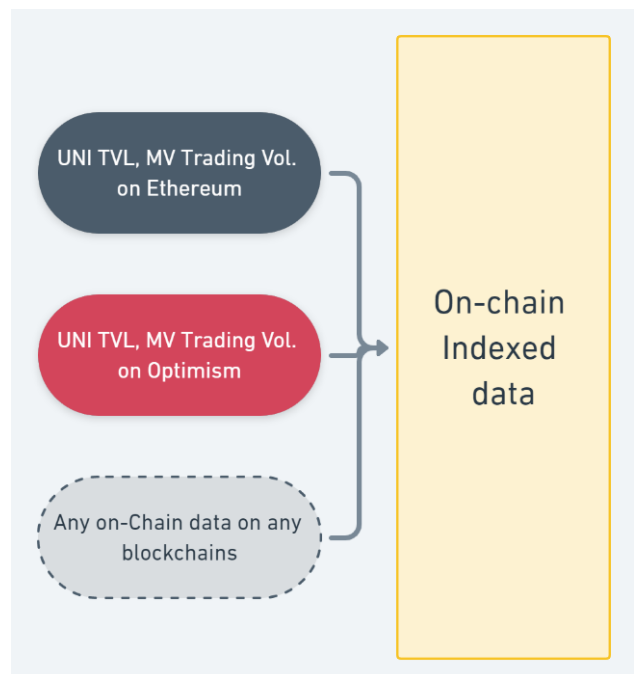


### I. Real-time signal collection & consolidation

#### A decentralized financial data market

*With empowering of theGraph and Chainlink, we can build a decentralized financial data market*

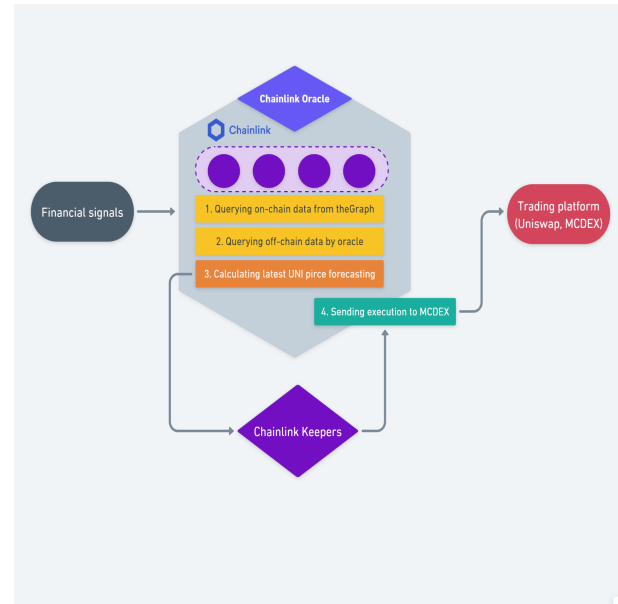
*→ de-quants can use it to cook their own on-chain strategies.*



## An off-chain computation environment

With empowering of Chainlink off-chain computation and task scheduler

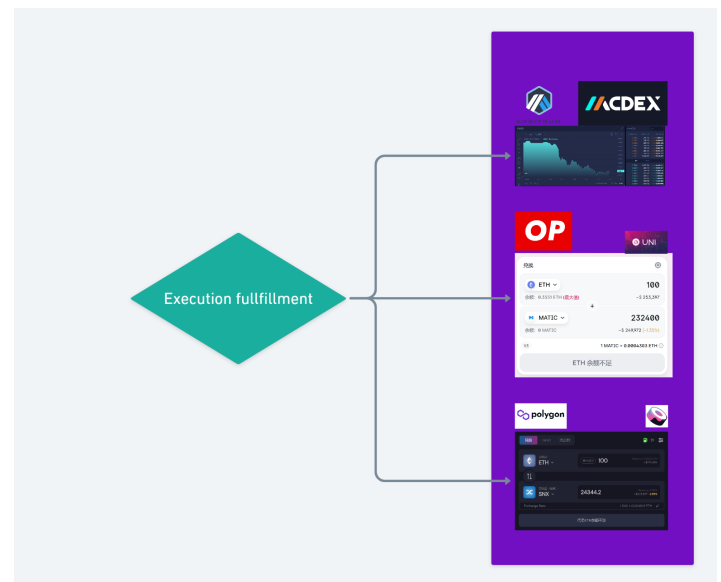
→ *de-quants can implement their strategy into oracle to master its data flow and execution.*



## An-quant trading portals hub

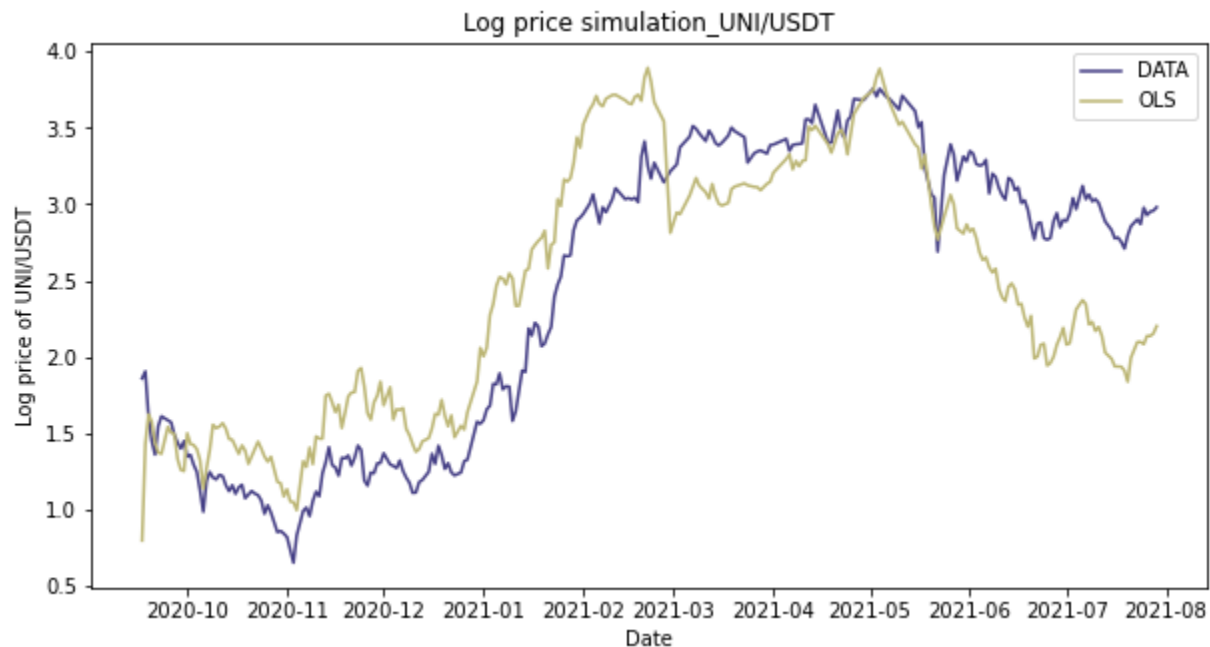
With empowering of on-chain trading derivatives / DEXs like MCDEX

→ *de-quants can access the best liquidity seamlessly with no development cost.*



## DEMO PROJECT

How to use linear regression to find correlation between a DeFi's TVL, MV trading volume to its token price? - Case study UNI.



## Modeling & backtesting

We try to build a multiple linear regression model to explore the correlation between UNI's on-chain business data on both Ethereum L1 & Optimism and UNI's price.

$$\log(\text{Price}) = \beta_0 + \beta_1 * \log(\text{TVL}) + \beta_2 * \log(24H \text{ Trading Volume})$$

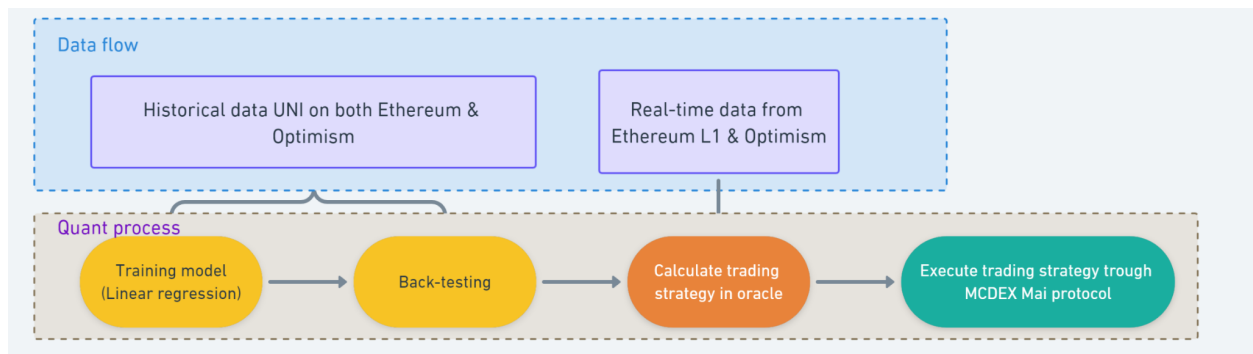
OLS Regression Results						
Dep. Variable:	price_log	R-squared:	0.801			
Model:	OLS	Adj. R-squared:	0.786			
Method:	Least Squares	F-statistic:	54.35			
Date:	Sat, 31 Jul 2021	Prob (F-statistic):	3.42e-10			
Time:	04:33:13	Log-Likelihood:	51.658			
No. Observations:	30	AIC:	-97.32			
Df Residuals:	27	BIC:	-93.11			
Df Model:	2					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
Intercept	-13.8673	1.618	-8.572	0.000	-17.187	-10.548
logLiquidity	0.9699	0.097	10.045	0.000	0.772	1.168
logusd_volume	-0.0154	0.018	-0.848	0.404	-0.053	0.022
Omnibus:	0.425	Durbin-Watson:		2.199		
Prob(Omnibus):	0.809	Jarque-Bera (JB):		0.563		
Skew:	-0.217	Prob(JB):		0.755		
Kurtosis:	2.488	Cond. No.		4.69e+03		

## Linear regression modeling result

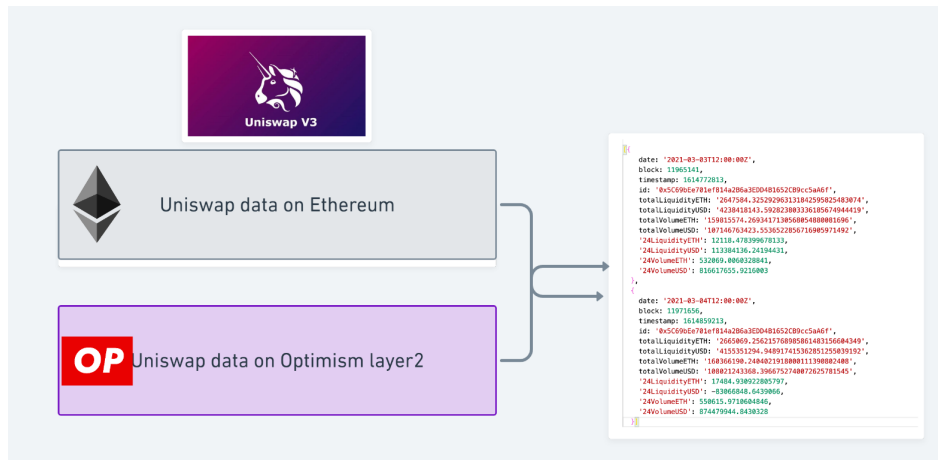
In the model, we take the log of the three variables in the regression and we get the parameters by fitting the historical data by OLS method, and the final results are as follows.

$$\log(\text{Price}) = -13.8673 + 0.9699 * \log(\text{TVL}) - 0.0154 * \log(24\text{H Trading Volume})$$

## Implementing data query and consolidation



## Quant trading on TR user flow



## Method to consolidate DeFi values from multi-blockchains

```

/**
 * Make TheGraph request
 */
function requestTheGraphData(address _oracle, string memory _jobId) public {
    Chainlink.Request memory req = buildChainlinkRequest(
        stringToBytes32(_jobId),
        address(this),
        this.fulfillEthereumData.selector
    );
    req.add("project", "TR_ETH_Aave");
    req.add("index", "FinMerics");
    req.add("copyPath", "result");
    sendChainlinkRequestTo(_oracle, req, fee);
}

```

*Sample code: querying from theGraph*

## Strategy implementation

$$\log(P1) = -13.8673 + 0.9699 * \log(TVL) - 0.0154 * \log(MV \text{ trading volume})$$

[/contract/theGraphDataEA.sol](#)

```

/**
 * calculate
 * log(beta_0)+beta_1*log(TVL)+beta_2*log(MV trading volume)
 *
 * @params totalLiquidityUSD
 * @totalVolumeUSD totalVolumeUSD
 *
 * https://woolen-twill-715.notion.site/modeling-9a6ee46e2c40456ea944b2d6afdbe9cb
 */
const calculate = ({ totalLiquidityUSD, totalVolumeUSD }) => {
    // -13.8673+0.9699*log(TVL)-0.0154*log(MV trading volume)
    // -13.8673+0.9699*log(TVL)-0.0154*log(MV trading volume)
    const beta_0 = -13.8673;
    const beta_1 = 0.9699;
    const beta_2 = 0.0154;

    // log(beta_0)+beta_1*log(TVL)+beta_2*log(MV trading volume)

    const result = log(beta_0) + beta_1 * log(totalLiquidityUSD, 10) + beta_2 * log(totalVolumeUSD, 10)
    return result
}

```

*Code implementation of linear regression functionalities*

## Execution Fulfillment

/contract/theGraphDataEA.sol

```
function fulfillEthereumData(bytes32 _requestId, bytes32 _data)
    public
    recordChainlinkFulfillment(_requestId)
{
    data = _data;

    /**
     * Call mcdex trade contract
     * 0 1 2
     * 1 tradeBuy 2 tradeSell
     */
    if(_data == stringToBytes32("1")){
        IMcdexTrade(address(0xcdd440d33D8A1Cb2c53846A6b77586F87e9b4812)).tradeBuy();
    }
    if(_data == stringToBytes32("2")){
        IMcdexTrade(address(0xcdd440d33D8A1Cb2c53846A6b77586F87e9b4812)).tradeSell();
    }
}
```

/contract/mcdex-mai/mcdex..sol

```
/**
 * tradeBuy
 * 如果你是合约调用并且抵押物在合约中, trader应该是你的合约地址
 * 合约调用时block.timestamp不可能改变, 所以deadline==block.timestamp
 * referrer是远点地址, 可以是0, 也可以是你们团队
 * flag==0
 * flag为0时, 要先调用deposit函数
 * index 8. amount是抵押物
 *
 * 做多amount为正数, 做空为负数
 */
function tradeBuy() public {
    IPerpetual(address(0xc32a2dfe97E2bAbC90a2b5e6Aef41e789eF2E13)).trade(8,address(this),1* 10**18,30* 10**18,block.timestamp,address(0),0);
}

/**
 * tradeSell
 * 如果你是合约调用并且抵押物在合约中, trader应该是合约地址
 * 合约调用时block.timestamp不可能改变, 所以deadline==block.timestamp
 * referrer是远点地址, 可以是0, 也可以是团队地址
 * flag==0
 * flag为0时, 要先调用deposit函数
 * index 8. amount是抵押物
 *
 * 做多amount为正数, 做空为负数
 */
function tradeSell() public {
    IPerpetual(address(0xc32a2dfe97E2bAbC90a2b5e6Aef41e789eF2E13)).trade(8,address(this),1* 10**18,-30* 10**18,block.timestamp,address(0),0);
}
}
```

**IF predict price > current price**

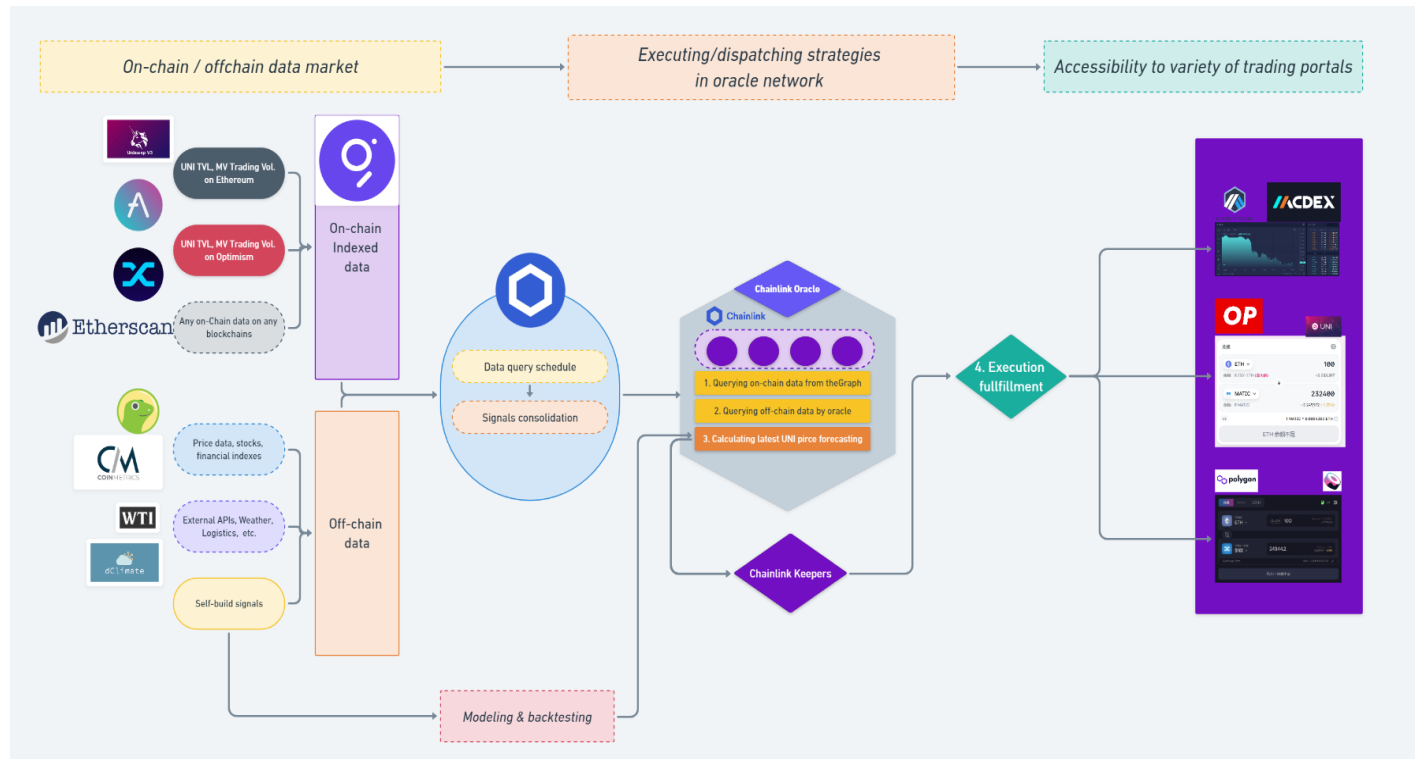
→ Buy in / keep buy position

**ELSE**

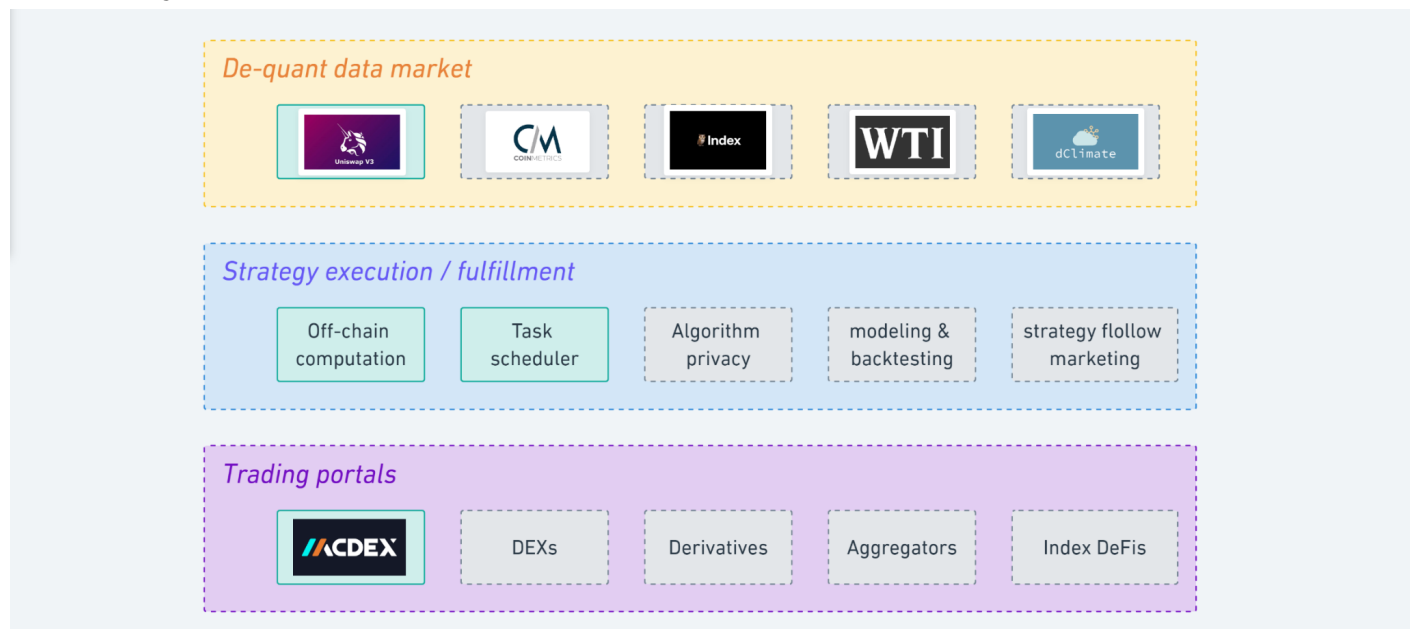
→ Sell out / keep empty position



## Overview



## Extensibility



## Team info

### ***Yu Wenqing - Team leader & product manager***

previously worked for BTCChina and Fundamental labs, He also worked for Chainlink as a Developer advocate and solution architect.

### ***Wei Yang - Smart contract developer***

Previously worked for multiple internet startups, Wei has strong development experience on Python, Node.js, and Solidity.

### ***Harry Hong - Data engineering***

Previously worked for Web3 foundation and multiple blockchain startups, Harry has plenty of blockchain technology experience, he is also a crypto enthusiast.

### ***Jamie Cheng - Solution architect***

Previously working for BTCChina, Jamie also founded a blockchain startup in 2018, he also worked for some famous blockchain projects as architect and technology advisors.

### ***Ms. X - Financial product manager***

Ms. X requires to stay anonymous, she worked for a top crypto corporate institution as a research analyst, she has strong capability in financial data analysis and solid experience on on-chain DeFi data analysis.

## Other info

Tech spec:

<https://docs.google.com/document/d/1IXZnjveEo0auYCogztoEqjKRZnxsyqEQE9CaevpH6O4/edit?usp=sharing>

Pitch deck:

<https://docs.google.com/presentation/d/1dK8rBgWmJkh5w2fzypOWWTOcGu6gPXXzEIGZZADZFxw/edit?usp=sharing>

GitHub: <https://github.com/Treasury-research/TR-theGraph-Chainlink-EA>

Twitter handle: <https://twitter.com/wenqingyu>

Email: [yuwenqingisu@gmail.com](mailto:yuwenqingisu@gmail.com)

## Reference

theGraph explorer

<https://thegraph.com/explorer>

theGraph Uniswap V3 Official

<https://thegraph.com/explorer/subgraph?id=0x9bde7bf4d5b13ef94373ced7c8ee0be59735a298-2&version=0x9bde7bf4d5b13ef94373ced7c8ee0be59735a298-2-0&view=Playground>

Dune Analytics - UNI borrow interest rate on AAVE

<https://duneanalytics.com/queries/93593>

Dune Analytics - Price in UNI

<https://duneanalytics.com/queries/93645>

Dune Analytics - Uniswap Ethereum vs. Optimism comparison

<https://duneanalytics.com/msilb7/Uniswap-v3-Ethereum-vs-Optimism>

Arbitrum Developer Page

[https://developer.offchainlabs.com/docs/public\\_testnet](https://developer.offchainlabs.com/docs/public_testnet)

MCDEX technical documents

<https://github.com/mcdexio>

Test trading page: UNI-USD on Arbitrum Rinkeby

<https://app.mcdex.io/trade/00009>

MCDEX Perpetual-interface document

<https://github.com/mcdexio/documents/blob/master/en/perpetual-interfaces.md>

LINK token contract list

<https://docs.chain.link/docs/link-token-contracts/>

Faucet website

<https://linkfaucet.protonfire.io/rinkebyarbitrum>