

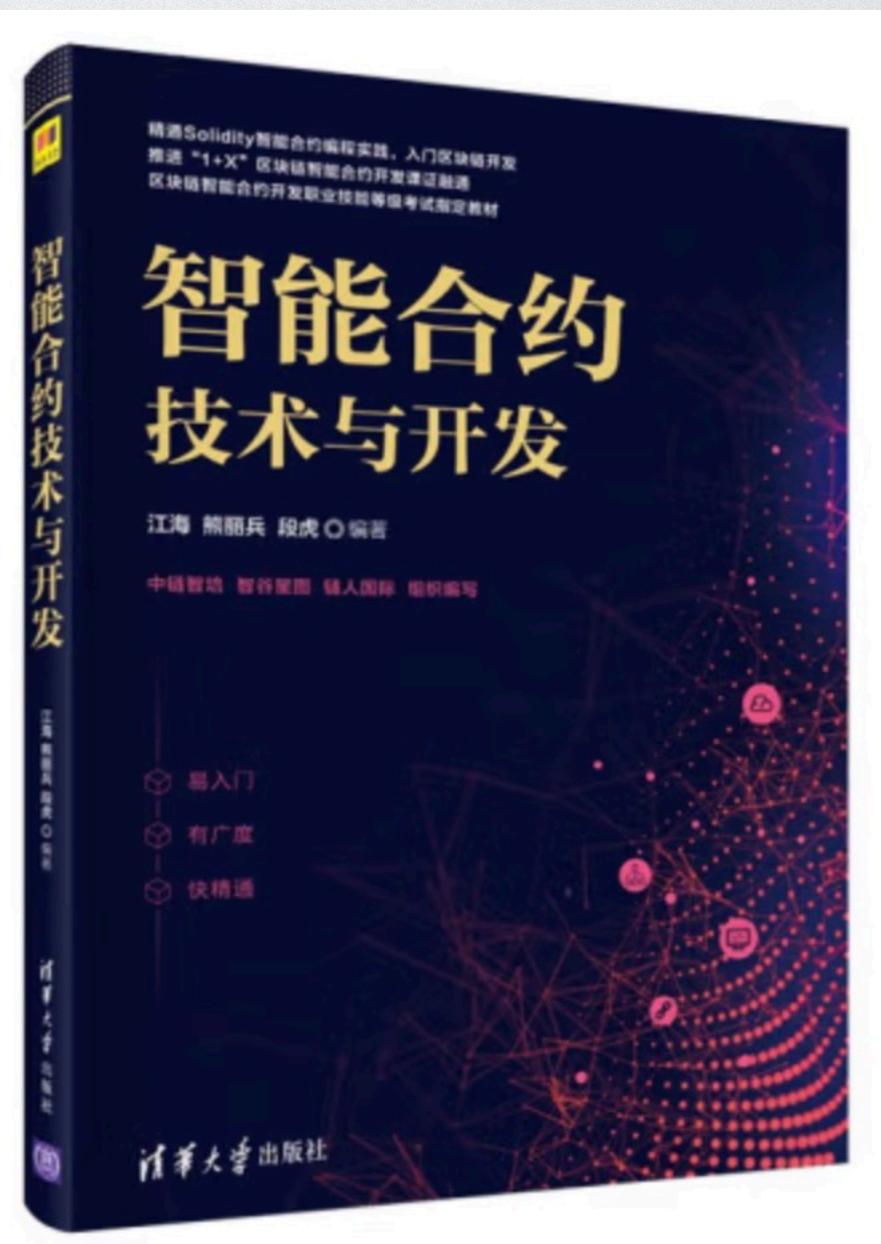
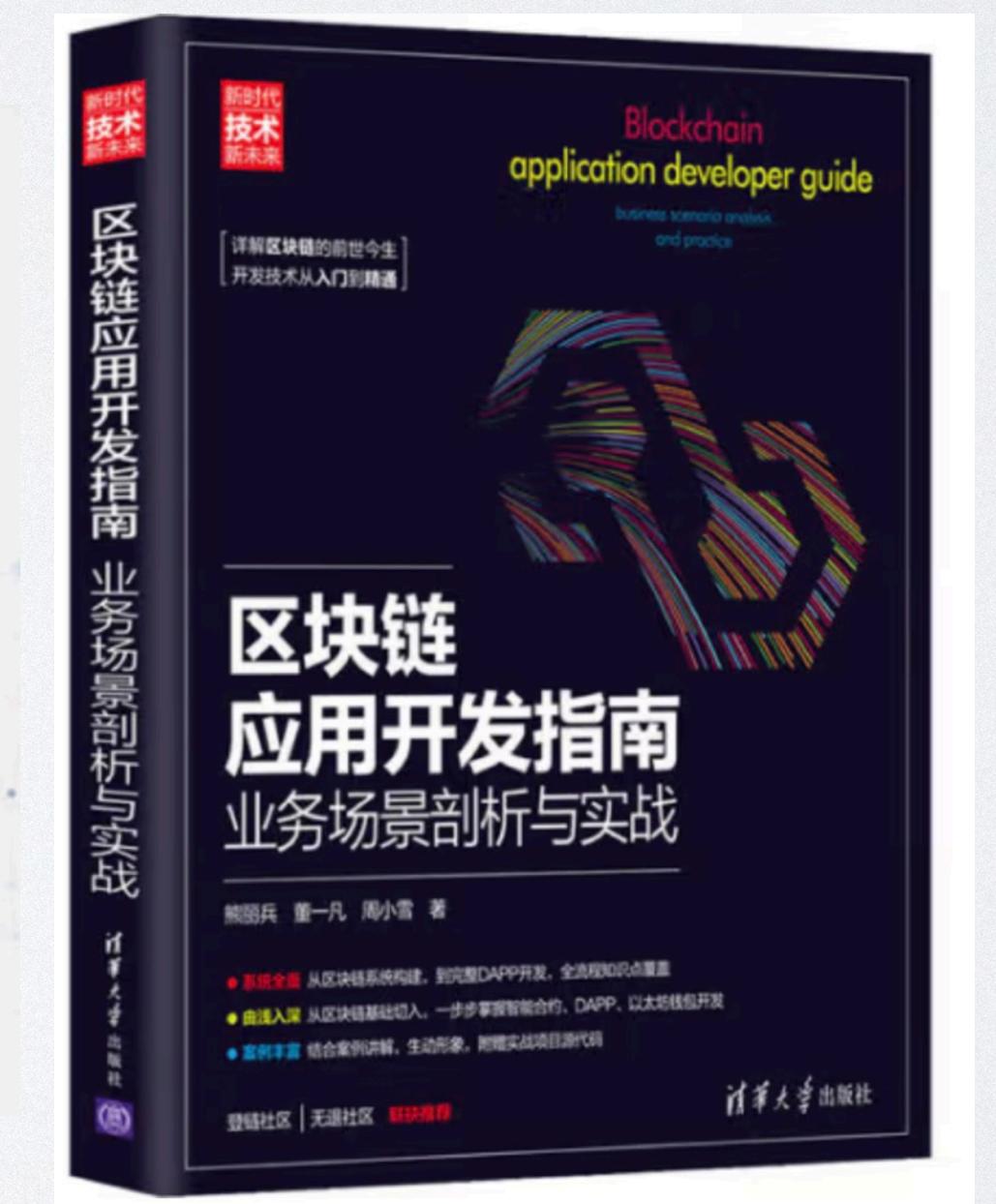
# 区块链集训营

## 二期

登链社区 - Tiny熊

# ABOUT ME

- Tiny 熊
- 创新工场（点心）、猎豹移动、合伙创业
- 登链社区发起人 (from 2017)
- 出版过几本区块链书籍：



# 课程介绍

- 视频直播、实时讨论
- 代码开源：[https://github.com/xilibi2003/training\\_camp\\_2](https://github.com/xilibi2003/training_camp_2)
- 图文教程：<https://decert.me/tutorial/solidity/intro/>
- 课后录播巩固、习题练习
- 微信群答疑
- 优秀学员推荐就业

# 练习解答

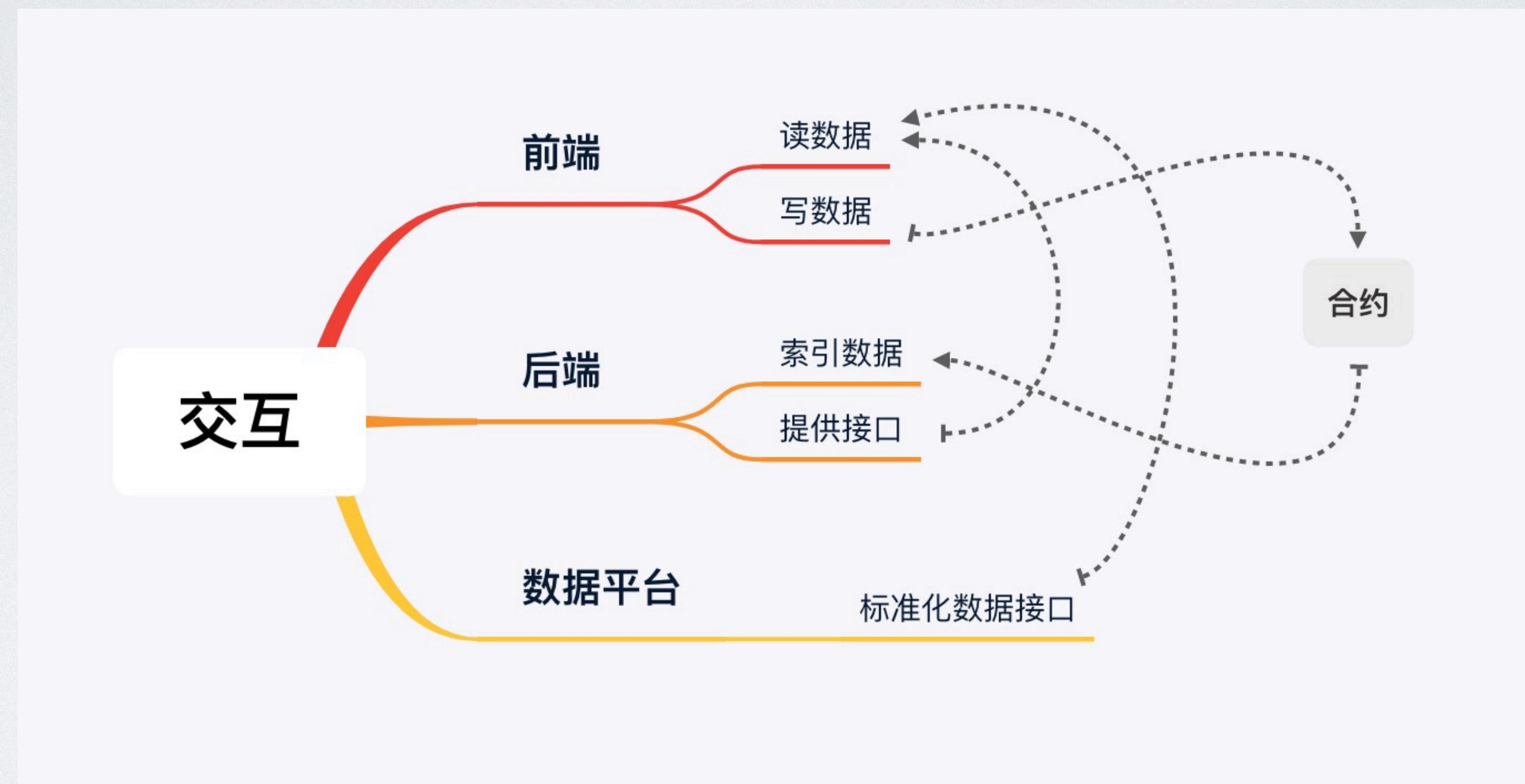
- 编写前端使用之前的Vault合约：
  - 用户可通过前端进行存款
    - 两个方式：Approve + deposit
    - 最好使用 ERC2612(Permit) 方式更好 (permitDeposit)
  - 前端显示用户存款金额
  - 用户可通过前端提取用户自己的存款 (withdraw)

[https://github.com/xilibi2003/training\\_camp\\_2](https://github.com/xilibi2003/training_camp_2) Answer 分支

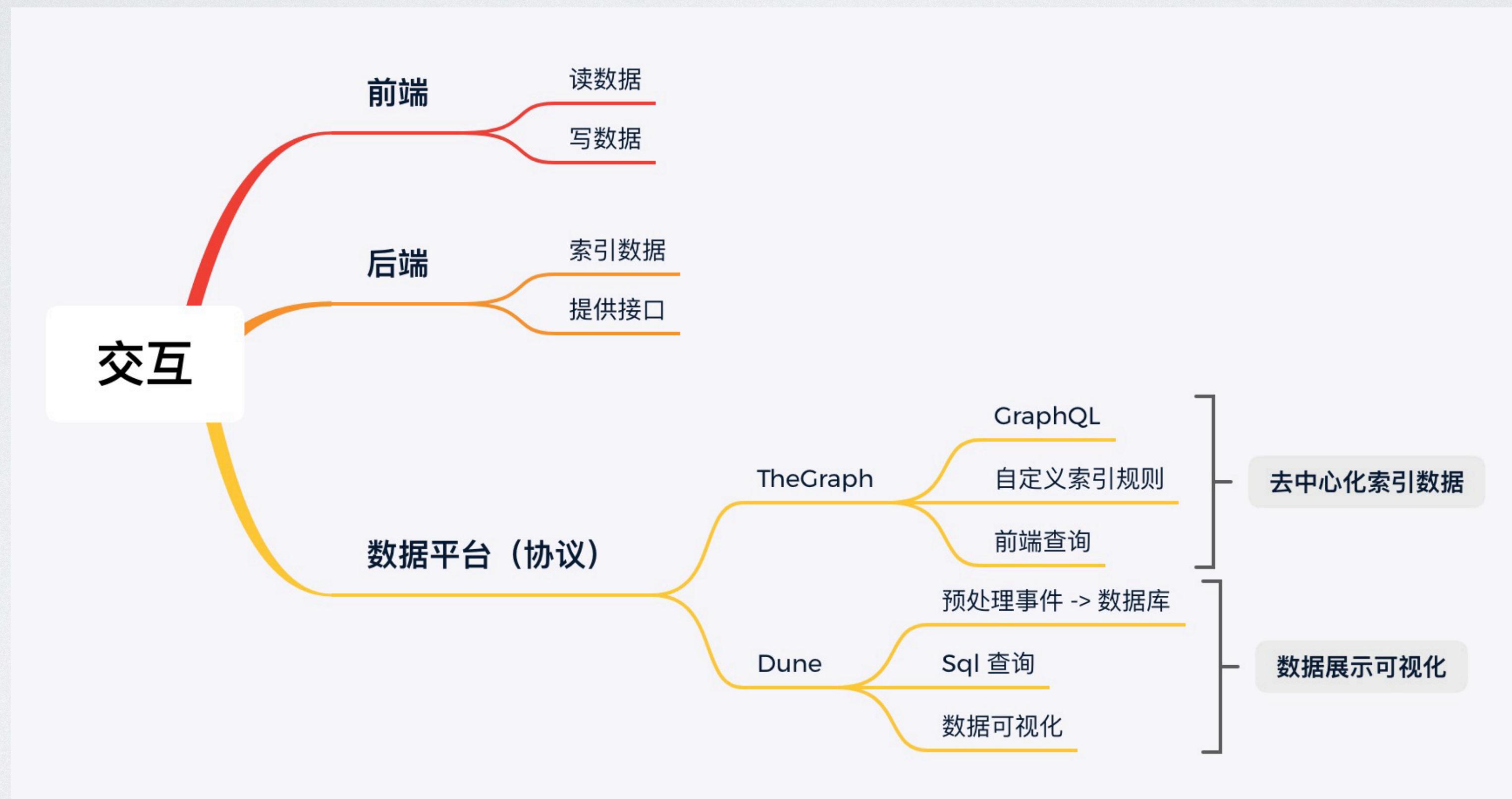
合约：w3\_2\_code/contracts/Bank.sol 及ERC2612.sol

前端：w4-vue

# 前后端交互



# 数据处理分析



# 任务

用 TheGraph 索引 Mumbai 网络上 Token  
转账记录及用户持有数据，并在前端展示

代码库：[https://github.com/xilibi2003/training\\_camp\\_2](https://github.com/xilibi2003/training_camp_2)

文件夹：w4\_thegraph

# 效果展示

Playground    Logs

Example query    Default    Save    Save as new    Cancel    ...

```
{  
  transfers(first: 5) {  
    id  
    from  
    to  
    value  
  }  
  users(first: 5) {  
    id  
    balance  
  }  
}
```

▶ {  
 "data": {  
 "transfers": [  
 {  
 "id": "0x57cf70539b7163c8a6e4f799ab20dfca4f316f419995c749a673093c58f7ae80b000000",  
 "from": "0x0000000000000000000000000000000000000000000000000000000000000000",  
 "to": "0xe74c813e3f545122e88a72fb1df94052f93b808f",  
 "value": "261200000000000000000000"  
 },  
 {  
 "id": "0x5e1d33e0a5e85fd6f9cd4d6c803a75f9b2babaa532f0a7a5734a1b317da7ba85b1d000000",  
 "from": "0xe74c813e3f545122e88a72fb1df94052f93b808f",  
 "to": "0x4923baaeb34f7081d41cc7a635a6a611d0086dd",  
 "value": "100000000000000000000000"  
 },  
 {  
 "id": "0x5e1d33e0a5e85fd6f9cd4d6c803a75f9b2babaa532f0a7a5734a1b317da7ba85b1d000000",  
 "from": "0xe74c813e3f545122e88a72fb1df94052f93b808f",  
 "to": "0x4923baaeb34f7081d41cc7a635a6a611d0086dd",  
 "value": "100000000000000000000000"  
 },  
 {  
 "id": "0x5e1d33e0a5e85fd6f9cd4d6c803a75f9b2babaa532f0a7a5734a1b317da7ba85b1d000000",  
 "from": "0xe74c813e3f545122e88a72fb1df94052f93b808f",  
 "to": "0x4923baaeb34f7081d41cc7a635a6a611d0086dd",  
 "value": "100000000000000000000000"  
 }  
 ]  
 }  
}

<https://thegraph.com/hosted-service/subgraph/xilibi2003/the-sample>

# 效果展示

交易列表:

从 : 0x00000000000000000000000000000000  
到 : 0xe74c813e3f545122e88a72fb1df94052f93b808f  
金额 : 26120000000000000000000000000000

从 : 0xe74c813e3f545122e88a72fb1df94052f93b808f  
到 : 0x4923baaeb34f7081d41ccd7a635a6a611d0086dd  
金额 : 10000000000000000000000000000000

从 : 0xe74c813e3f545122e88a72fb1df94052f93b808f  
到 : 0x72fc37a97d61229a471310d414ba36cc77886afb  
金额 : 10000000000000000000000000000000

用户列表:

地址 : 0x4923baaeb34f7081d41ccd7a635a6a611d0086dd  
持有 : 10000000000000000000000000000000

地址 : 0x72fc37a97d61229a471310d414ba36cc77886afb  
持有 : 10000000000000000000000000000000

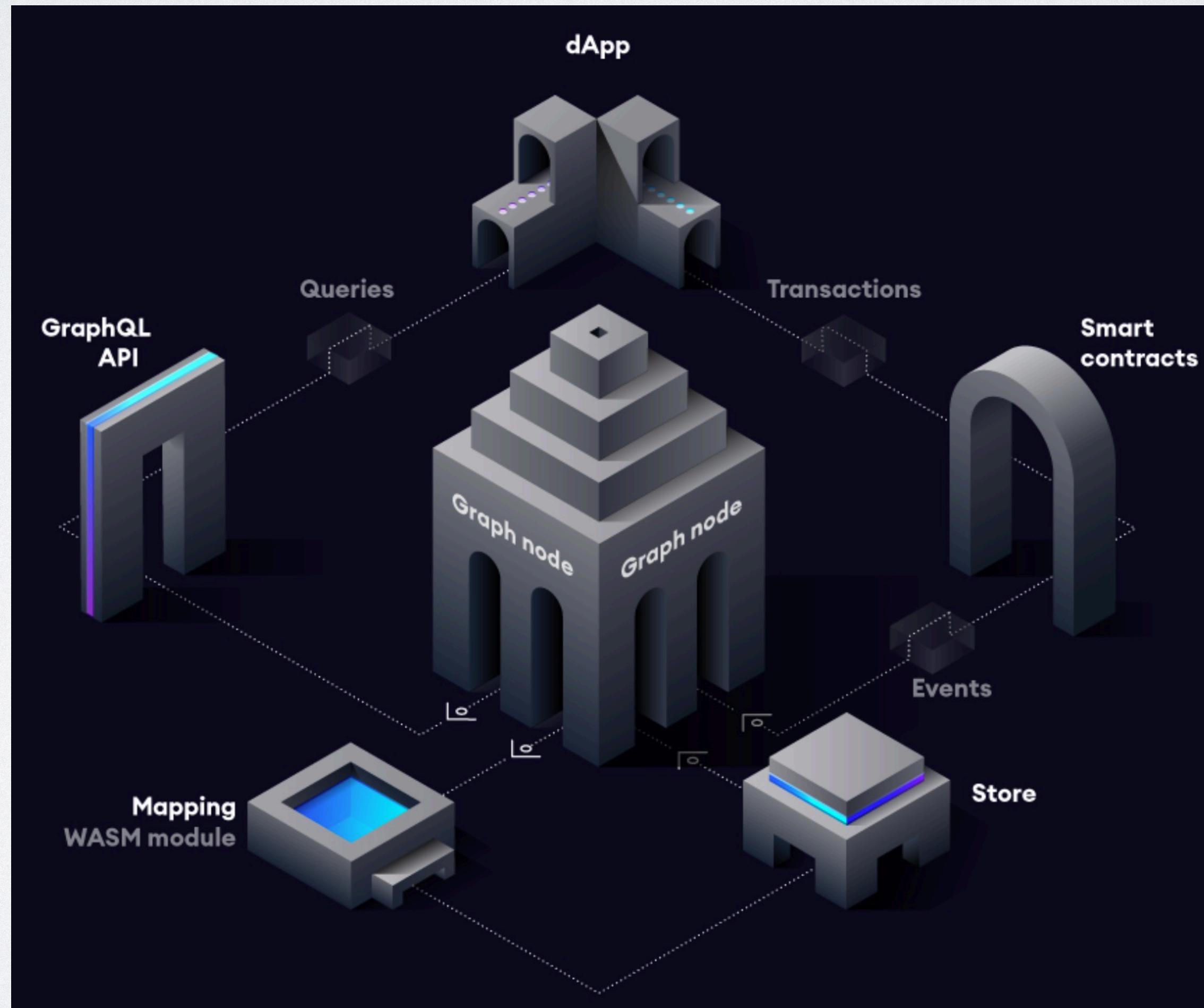
地址 : 0xe74c813e3f545122e88a72fb1df94052f93b808f  
持有 : 16020000000000000000000000000000

前端效果

# THEGRAPH

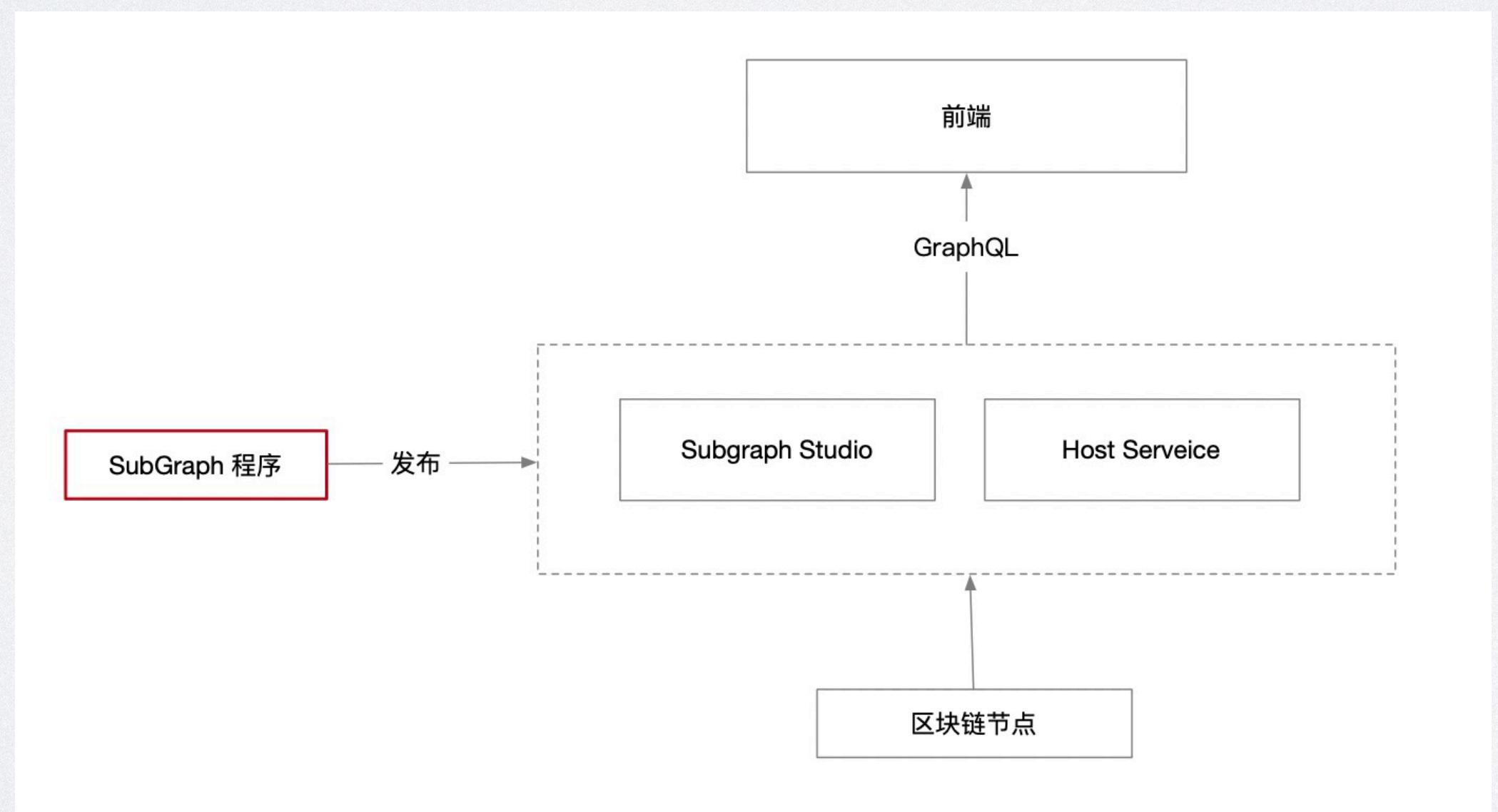
- 自己检索事件，存在的问题：
  - 消耗资源大（可能需要搭建节点、服务器）
  - 不够去中心化，用户需要信任单点服务器
  - 可能存在单点故障
- TheGraph：用于索引和查询区块链数据的去中心化协议

# THE GRAPH 是如何工作的



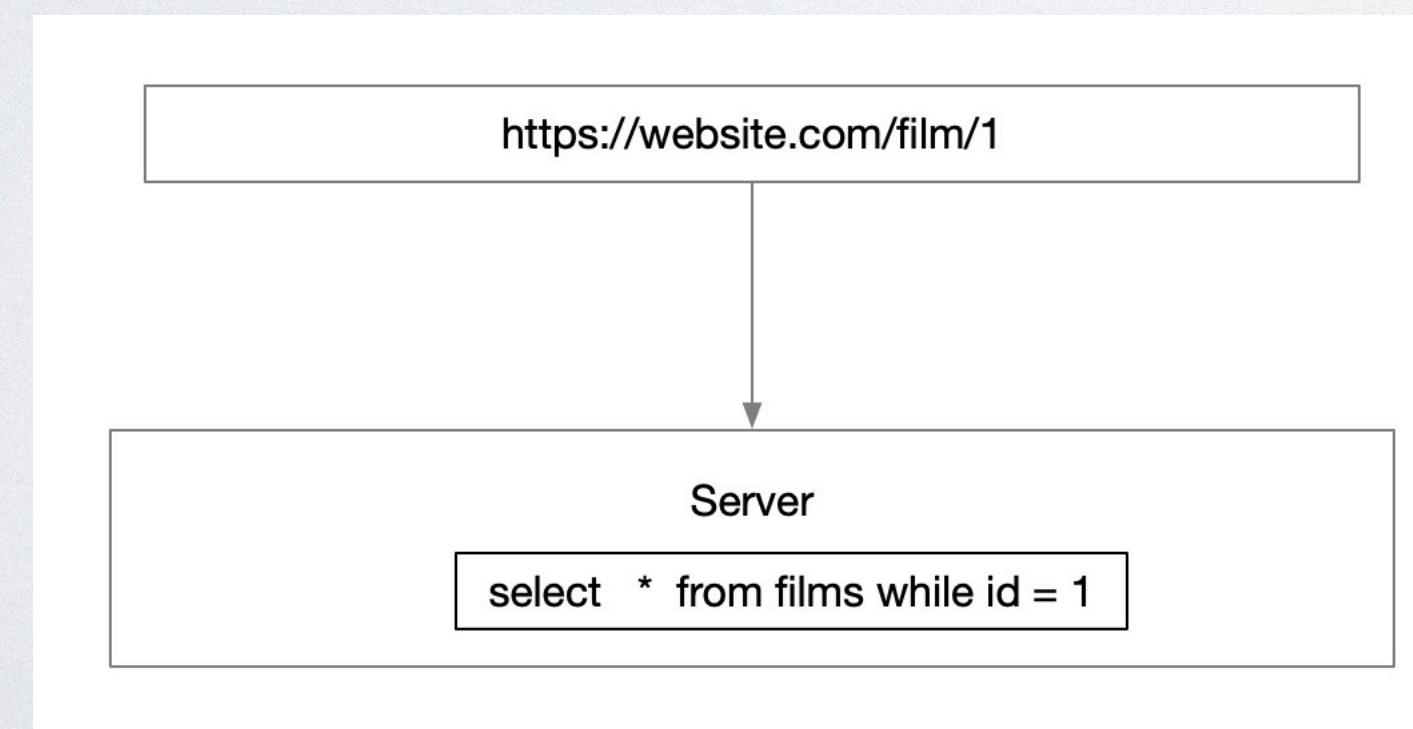
# 如何开发

- TheGraph 现在有两个服务：Subgraph Studio 和 Host Service
  - Subgraph Studio：去中心化网络（激励、付费）
  - Host Service：官方托管服务

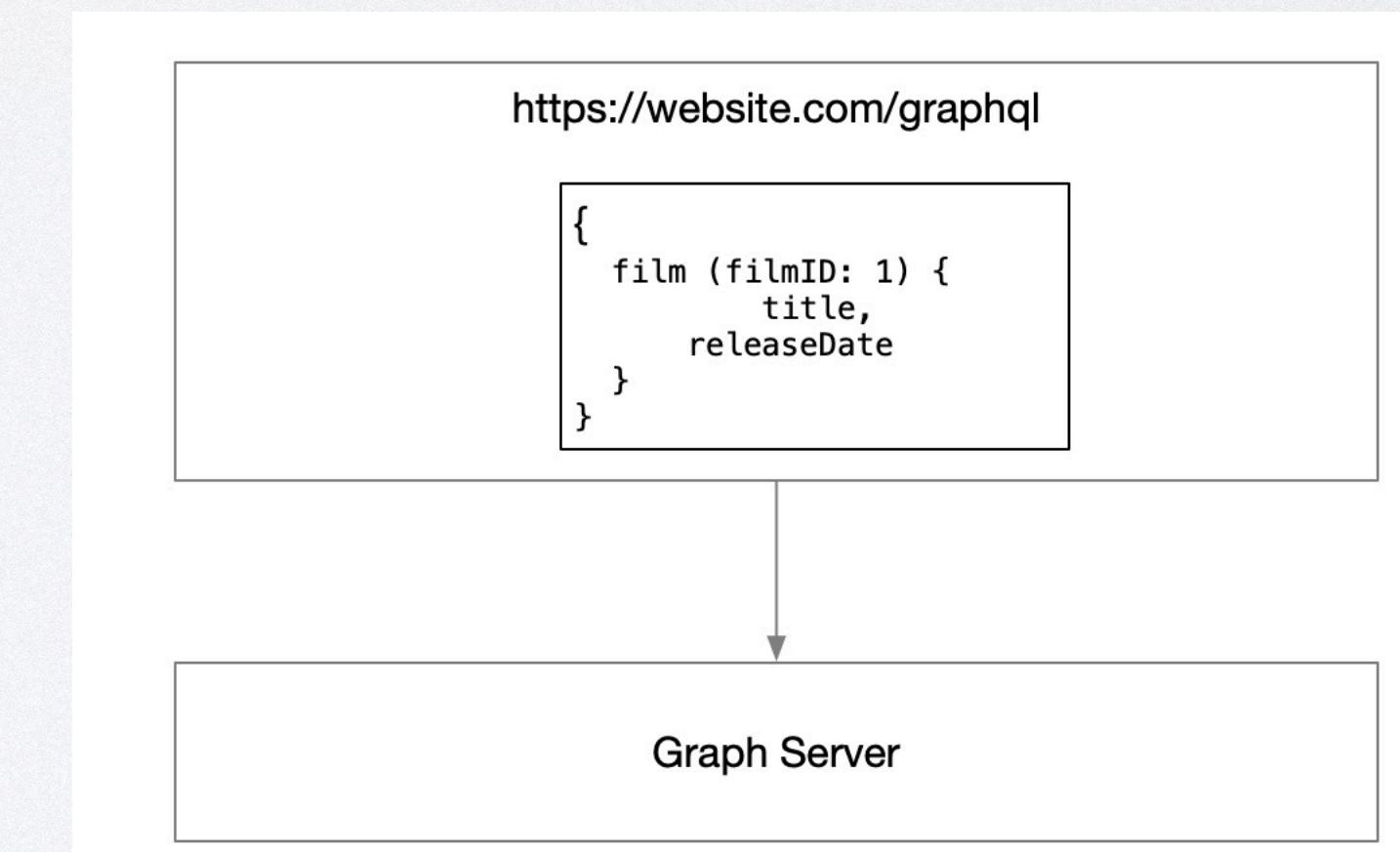


# THEGRAPH

- GraphQL 作为查询语言
  - Restful API: 查询语句在服务端编写
  - GraphQL: 查询语句 (schema) 通过请求发送到服务端



Restful API



GraphQL

# THE GRAPH 开发步骤

用 github  
注册账号

The screenshot shows the The Graph dashboard interface. At the top, there is a navigation bar with 'Hosted Service' (dropdown), 'Subgraphs', 'My Dashboard', and 'Docs'. On the right side of the header is a user profile icon. Below the header, the 'ACCOUNTS' section shows 'Tiny熊' with an access token '09781-8d69b'. A prominent purple button labeled 'Add Subgraph' is highlighted with a red border. The main content area displays four subgraphs:

- Old Ly**: Deployed 9 months ago. Associated with GitHub user `XILIBI2003`. Includes a link to "Get Old Ly balance".
- Lytoken**: Not Deployed.
- The Sample**: Not Deployed.
- TheSample**: Not Deployed.

# THE GRAPH 开发步骤

NEXT STEPS

Install	Init	Deploy
Install Graph CLI globally using NPM or Yarn so you can build and deploy Subgraphs	Scaffold a Hello World Subgraph that indexes data off Ethereum mainnet	Deploy your subgraph to the Graph Node and publish it to our community once you are ready
INSTALL GRAPH CLI USING NPM <code>npm install -g @graphprotocol/graph-cli</code>	INITIALIZE SUBGRAPH <code>graph init --product hosted-service xilibi2003</code>	AUTHENTICATED THROUGH CLI <code>graph auth --product hosted-service 09781-8c</code>
INSTALL GRAPH CLI USING YARN <code>yarn global add @graphprotocol/graph-cli</code>	CD INTO SUBGRAPH DIRECTORY <code>cd lytoken</code>	DEPLOY SUBGRAPH <code>graph deploy --product hosted-service xilibi2003</code>
RUN CODEGEN AND BUILD SUBGRAPH <code>graph codegen &amp;&amp; graph build</code>		

# THE GRAPH 开发步骤

## 1. 安装 GraphCLI

1. `npm install -g @graphprotocol/graph-cli`

## 2. 初始化TheGraph工程

• `graph init --product hosted-service xilibi2003/the-sample`

## 3. 编写索引事件的代码 (subgraph 子图)

1. 记录转帐、用户持有金额

## 4. 发布到TheGraph, 等待一会, 就可以查询数据了

代码库: [https://github.com/xilibi2003/training\\_camp\\_2](https://github.com/xilibi2003/training_camp_2)

文件夹: w4\_thegraph/the-sample

# THEGRAPH 工程结构

```
├── abis
│   └── ERC2612.json
├── build
│   ├── ERC2612
│   ├── schema.graphql
│   └── subgraph.yaml
├── generated
│   └── ERC2612
│       └── schema.ts
├── networks.json
└── schema.graphql      // 定义主体
    └── src
        └── erc-2612.ts      // 定义索引规则
    └── subgraph.yaml      // 配置
```

# 查询

Playground Logs

Example query Default Save Save as new Cancel ...

```
{  
  transfers(first: 5) {  
    id  
    from  
    to  
    value  
  }  
  users(first: 5) {  
    id  
    balance  
  }  
}
```

{"data": {"transfers": [  
 {  
 "id": "0x57cf70539b7163c8a6e4f799ab20dfca4f316f419995c749a673093c58f7ae80b000000",  
 "from": "0x0000000000000000000000000000000000000000000000000000000000000000",  
 "to": "0xe74c813e3f545122e88a72fb1df94052f93b808f",  
 "value": "26120000000000000000000000000000"  
 },  
 {  
 "id": "0x5e1d33e0a5e85fd6f9cd4d6c803a75f9b2baba532f0a7a5734a1b317da7ba85b1d000000",  
 "from": "0xe74c813e3f545122e88a72fb1df94052f93b808f",  
 "to": "0x4923baaeb34f7081d41ccd7a635a6a611d0086dd",  
 "value": "10000000000000000000000000000000"  
 },  
 {  
 "id": "0xe63457a2dbcde6affb24ff76f7b752c4e7ff0f61546518008ad4b0c07c1dd5070d000000",  
 "from": "0xe74c813e3f545122e88a72fb1df94052f93b808f",  
 "to": "0x72fc37a97d61229a471310d414ba36cc77886afb",  
 "value": "10000000000000000000000000000000"  
 }],  
 "users": [  
 {  
 "id": "0x4923baaeb34f7081d41ccd7a635a6a611d0086dd",  
 "balance": "10000000000000000000000000000000"  
 }]  
]

# 前端查询

## I. 安装 Apollo

- `npm install —save graphql graphql-tag @apollo/client`

```
gql`query {
  transfers(first: 5) {
    id
    from
    to
    value
  }
}`
```

代码库: [https://github.com/xilibi2003/training\\_camp\\_2](https://github.com/xilibi2003/training_camp_2)  
文件夹: w4\_thegraph/vue-project

# Q & A

# 练习题

- 发行一个 ERC721 Token (使用之前发布的就可以)：
  - 用 TheGraph 索引 ERC721 的转账事件。
  - 在 playground 可查询到数据 (截图)

第4周

# DUNE

@TinyXiong / camp2

0 ☆ Fork Share Edit

USDC一天交易量 usdc day values

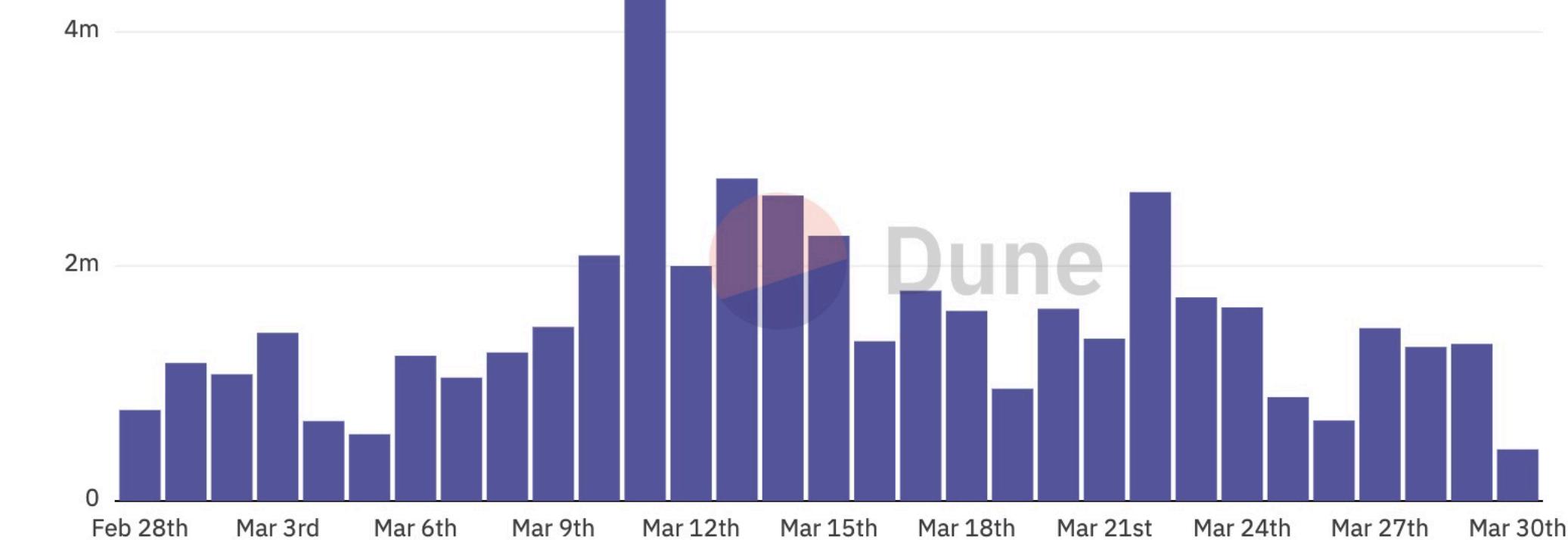
@TinyXiong

\$6,250,432,223  
USDC一天交易量

每天ETH交易量(E)

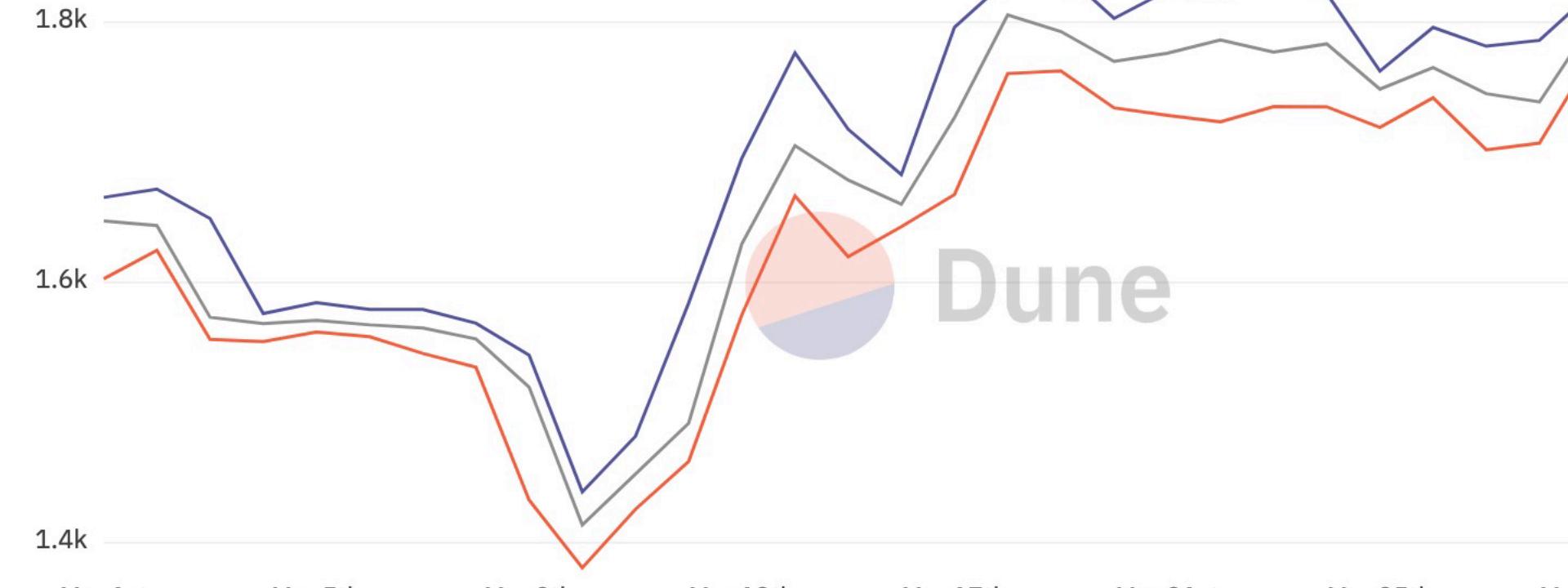
查询过去 30 天每天交易量

@TinyXiong



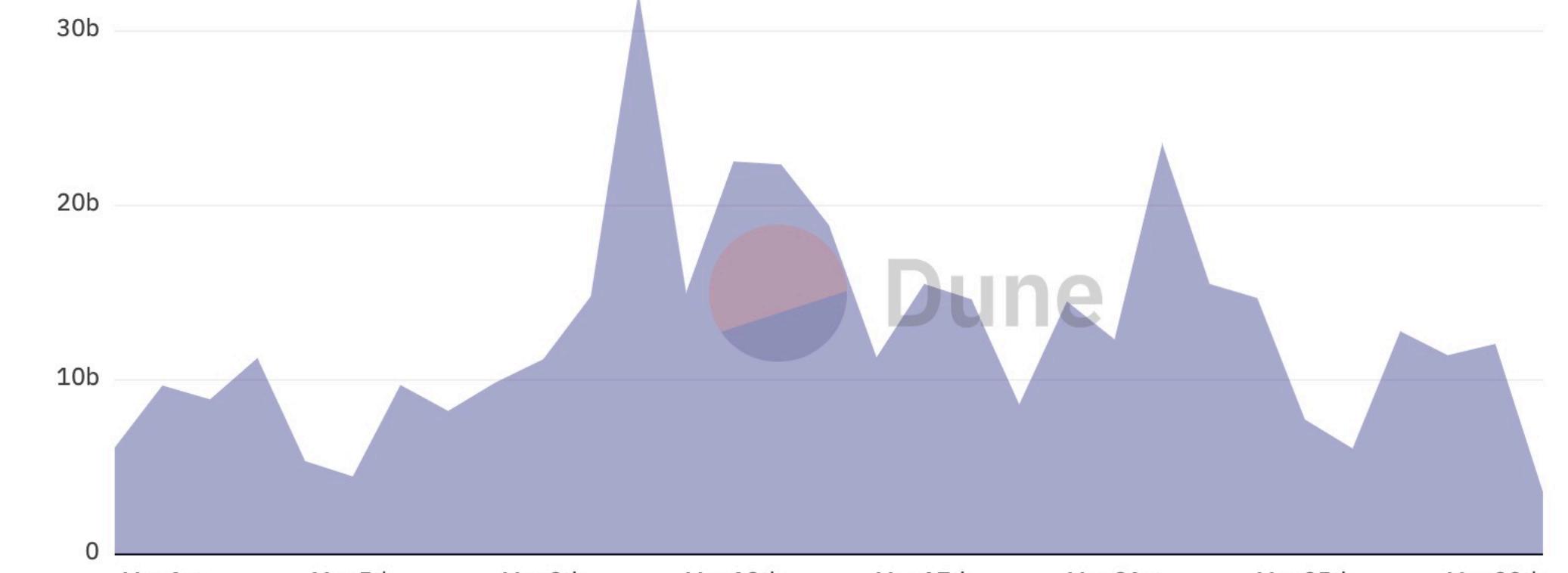
过去 30 天 ETH 价格 ETH Price High/Lows Daily (30D)

@TinyXiong



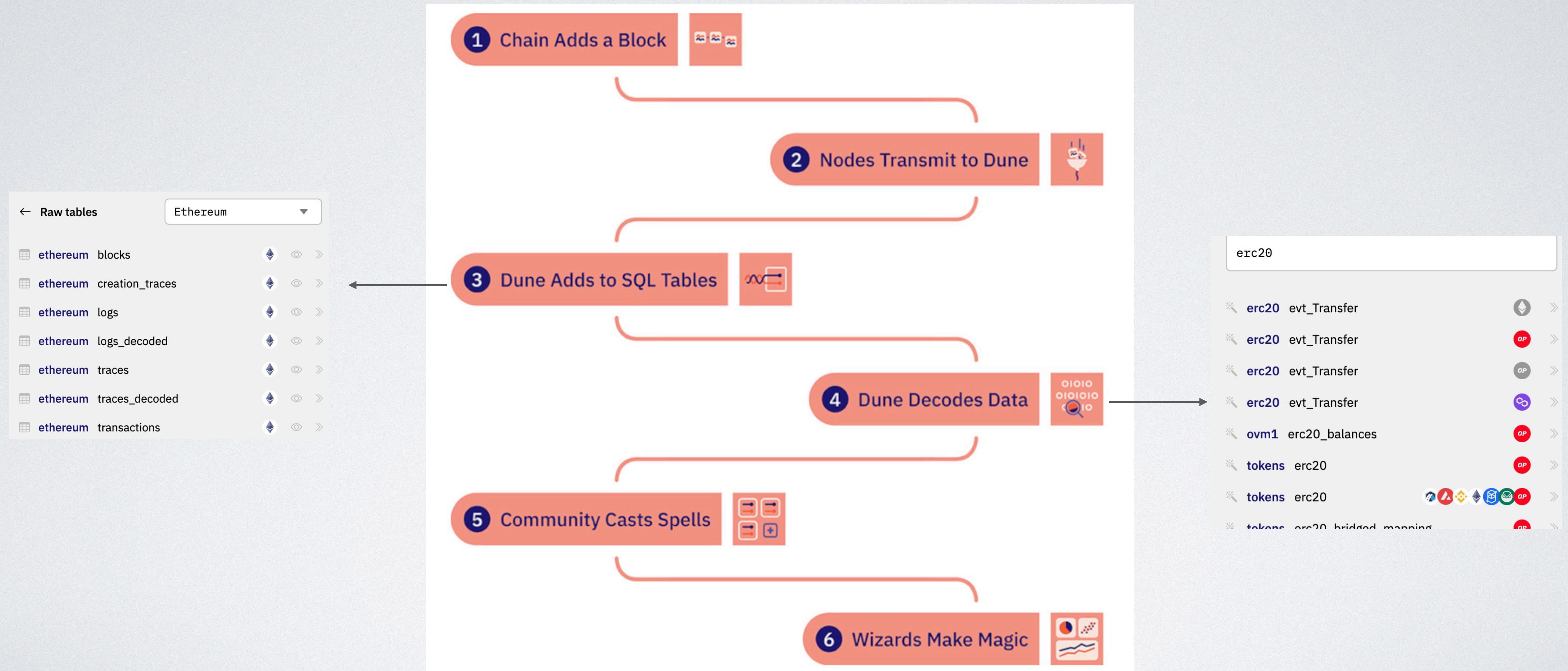
ETH 交易量\$(30D)

@TinyXiong



第4周

# DUNE 是如何工作的



# 开始探索 DUNE

<https://dune.com/>

看看 4 张表是如何完成的

# Q & A

# 练习题

- 用 Dune 查询过去 30 天 USDC 的价格变化