Web3 数据采集

七哥

https://x.com/0xqige

思考

❖ 如何在自有DB中存储NFT交易动态?

❖ 交易所是如何感知用户充值 USDT 到帐的?

采集哪些数据

区 块　　交 易　　交易回执　　交易日志

数据采集途径

1. 通过 RPC API 从节点读取数据

轮询 Poll　or　订阅 Subscribe

2. 从第三方数据服务商拉取

通过 RPC API 从节点读取数据

https://ethereum.org/en/developers/docs/apis/json-rpc/

区 块	eth_getBlockByHash，eth_getBlockByNumber
交 易	eth_getTransactionByHash
交易回执	eth_getTransactionReceipt
交易日志	eth_getLogs

```
> curl -s  https://cloudflare-eth.com/ --data-raw '{"jsonrpc":"2.0","id":0,"method":"eth_getBlockByNumber","params":["0x1362b35",false]}' |jq
{
  "jsonrpc": "2.0",
  "result": {
    "baseFeePerGas": "0x2b8c68165",
    "blobGasUsed": "0x80000",
    "difficulty": "0x0",
    "excessBlobGas": "0x60000",
    "extraData": "0x6265617665726275696c642e6f7267",
    "gasLimit": "0x1c9c380",
    "gasUsed": "0xc93da0",
    "hash": "0xcabb9c7abc06bb685af784d6a11dab5fcabd0353a73293970a9e8c7b2c2618d1",
    "logsBloom": "0x1aaba1841d0141d570da1e12f733410910ca65529bfb006c82888803c4630b94c0e13506a1480d1897001d110d2
503ee9751dbdcda2c6fe027c4f2785c2e49511205f4984aa78c49480252ab866395f9950386e316468f548197bc20c0b4c2a3000a3d781a
654026008f62320c084dc57b0873c909820c81cb3849b4580a9059ae018e59984a6a8029b9aaeb9776a11874417693816cc24974373c787
218830f728c894e22c2a1b50280d8d84c501d8e45532720302ad10006a26c1616eb84417f1b6017825684528d6089225f6b42d433b3040a
dea0763881f0d30bc985e04c5c797315eba12d521cd43291b000cb75a40ed9283b6280c2490d1972f2335c8f",
    "miner": "0x95222290dd7278aa3ddd389cc1e1d165cc4bafe5",
    "mixHash": "0xfbada9c89e5418c57c6671c06d1274f914bd8d3317cbacbda6404d49cfd5b125",
    "nonce": "0x0000000000000000",
    "number": "0x1362b35",
    "parentBeaconBlockRoot": "0xd63be2abef9e01ad7599f09f2ce94f41d83d395e3d65de867d8e3c6497bc72d6",
    "parentHash": "0x2ff8ce17cd40e8e448dd56dd6a3f1c463526281e2aa27ab25ee56762f6fafe9f",
    "receiptsRoot": "0x9ad3ef6f746916a6e5b97ba0783dbbb1d7be21c73ec4c214ed075b59398956d4",
    "sha3Uncles": "0x1dcc4de8dec75d7aab85b567b6ccd41ad312451b948a7413f0a142fd40d49347",
    "size": "0xe88b",
    "stateRoot": "0x29ea801f9903fcfd611686889d09bf7ae3ba638e92209d334a2a6aec7a2fc7f0",
    "timestamp": "0x6697e6d7",
    "totalDifficulty": "0xc70d815d562d3cfa955",
    "transactions": [
      "0x47067b128def6fb144367f17be71974589f5f8fd84b2afea8b66fab453c5e7fd",
      "0xcf89304b1da8d845ce499fc2b254e644403b5e73cd0fdeb7585cb9094cd0599f",
      "0x9ff38d2dbc529d4310908e381969d369f25e599272030531e4f94a1c4d1bba31",
      "0x6572a68231ee5ecd1098cb494f2d8b305bd80b565cff13ece1b967388bd1e5d8",
      "0x54c0bf57043f43d2005ab1fdfcfe15ed116572df425efab49b065fa6e8c8fe7c",
      "0x27d6560215e233e043eac31fd70e76f18b1cc92ffda3e996111cd44ae4162e18",
      "0x71ab1b6050762e060147f173838b10bae8c18df0c7ef5f4d3f361bc5992fe33f",
      "0x45ed5c4054dc89f5ef2877191b8dc44a9b382e54833d600dcdb6df928806ed0b",
```

读取日志

eth_getLogs：返回与给定过滤器对象匹配的所有日志的数组

入参 Object

· **fromBlock**：-（可选，默认值"latest"：）整数块号，或"latest"对于最后提议的块，"safe"对于最新的安全块，"finalized"对于最新的最终确定的块，或者"pending"，"earliest"对于尚未在块中的交易。

· **toBlock**：-（可选，默认值"latest"：）整数块号

· **Address**：Address 数组，20 字节 -（可选）合约地址或日志应源自的地址列表。

· **topics**：数组,-（可选）32 字节主题数组DATA。主题是顺序相关的。每个主题也可以是带有"或"选项的数据数组。

· **blockhash**: DATA，32 字节 -（可选，**保留给未来使用**）

·

Topics: [topic0In, topic1In,topic2In,topic3In]

不同 Topics 值，可以匹配过滤不同的的Event：

· **[ ]** "任何日志"
· **[A]** topic[0]为 A，其他任意" Topics: [[A]]
· **[null，B]** topic[0]任意，topic[1] 为B，其他也任意 Topics: [[null], [B]]
· **[A, B]** topic[0]为 A 并且 topci[1]为 B ，其他任意 Topics: [[A], [B]]
· **[[A, B], [A, B]]** topic[0]为A或者B 并且 topic[1]为 A或B Topics: [[A,B], [A, B]]
·

读取日志

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x429d3a",
      "toBlock": "0x429d3b",
      "address": "0xb59f67a8bff5d8cd03f6ac17265c550ed8f33907",
      "topics": [
        "0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef",
        "0x000000000000000000000000b46c2526e227482e2ebb8f4c69e4674d262e75",
        "0x0000000000000000000000054a2d42a40f51259dedd1978f6c118a0f0eff078"
      ]
    }
  ]
}
```

本质是 logBloom Filter

类似于SQL 查询

⊡ Logs

> Transfer (index_topic_1 address from, index_topic_2 address to, uint256 value)

[topic0] 0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef ▽

[topic1] 0x000000000000000000000000b46c2526e227482e2ebb8f4c69e4674d262e75

[topic2] 0x000000000000000000000001c495e48cc5612e90dbfff05b12532a69303bf72

Hex ∨ → 0000000000000000000000000000000000000000000000000000000000002a7b00

```
Select  * from  ethereum_logs  where
block_number bwtween 0x429d3a and 0x429d3b
AND  event_address = '0xb59f67a8bff5d8cd03f6ac17265c550ed8f33907'
AND  event_topic0 =  '0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef'
AND event_topci1= '0x000000000000000000000000b46c2526e227482e2ebb8f4c69e4674d262e75'
AND event_topic2= '0x0000000000000000000000054a2d42a40f51259dedd1978f6c118a0f0eff078'
```

读取日志

1. 如何读取一个区块区间中的所有日志
2. 如何读取指定区块中的 Transfer 日志
3. 如何读取所有ERC2转账给张三的日志
4. 如何读取所有NFTMarket中的上架信息以及NFT交易信息

读取日志

读取一个区块区间中的所有日志

```json
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x429d3a",
      "toBlock": "0x429d3b"
    }
  ]
}
```

读取日志

读取指定区块0x429d3b中的 Transfer 日志

```json
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x429d3b",
      "toBlock": "0x429d3b",
      "topics": ["0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef"]
    }
  ]
}
```

读取日志

读取所有ERC2转账给 0xF977814e90dA44bFA03b6295A0616a897441aceC 的日志

```json
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x13630a1",
      "toBlock": "0x13633b5",
      "topics": [
        ["0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef"],
        null,
        ["0x000000000000000000000000f977814e90da44bfa03b6295a0616a897441acec"]
      ]
    }
  ]
}
```

读取日志

读取所有ERC2转账给 0xF977814e90dA44bFA03b6295A0616a897441aceC 的日志

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x13630a1",
      "toBlock": "0x13633b5",
      "topics": [
        "0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef",
        null,
        "0x000000000000000000000000f977814e90da44bfa03b6295a0616a897441acec"
      ]
    }
  ]
}
```

读取日志

读取所有ERC2转账给 0xF977814e90dA44bFA03b6295A0616a897441aceC 和 0xCF217F59434E99fA060ECe5ff9555945b81fae40 的日志

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x13630a1",
      "toBlock": "0x13633b5",
      "topics": [
        "0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef",
        null,
        ["0x000000000000000000000000f977814e90da44bfa03b6295a0616a897441acec",
         "0x000000000000000000000000cf217f59434e99fa060ece5ff9555945b81fae40"]
      ]
    }
  ]
}
```

读取日志

读取所有NFTMarket(0x7448c7456a97769F6cD04F1E83A4a23cCdC46aBD)中的上架信息以及NFT交易信息

```json
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x13630a1",
      "toBlock": "0x13633b5",
      "address":"0x7448c7456a97769F6cD04F1E83A4a23cCdC46aBD"
    }
  ]
}
```

```json
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x13630a1",
      "toBlock": "0x13633b5",
      "address":"0x7448c7456a97769F6cD04F1E83A4a23cCdC46aBD",
      "topics":[
        ["List事件ID","成交事件ID"]
      ]
    }
  ]
}
```

读取日志

读取所有NFTMarket(0x7448c7456a97769F6cD04F1E83A4a23cCdC46aBD)中的上架信息以及NFT交易信息

```json
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x13630a1",
      "toBlock": "0x13633b5",
      "address":"0x7448c7456a97769F6cD04F1E83A4a23cCdC46aBD"
    }
  ]
}
```

```json
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "eth_getLogs",
  "params": [
    {
      "fromBlock": "0x13630a1",
      "toBlock": "0x13633b5",
      "address":"0x7448c7456a97769F6cD04F1E83A4a23cCdC46aBD",
      "topics":[
        ["List事件ID","成交事件ID"]
      ]
    }
  ]
}
```

读取日志的返回值

```
{
    "id": 0,
    "jsonrpc": "2.0",
    "result": [{
        "address": "0x1bbe973bef3a977fc51cbed703e8ffdefe001fed",
        "blockHash": "0x1683dcd79582e4f22dee9c0348a172073b899973c8d882bb8ccb1491a850b659",
        "blockNumber": "0x1363162",
        "data": "0x00000000000000000000000000000000000000000000000071a9df42f256b7440000",
        "logIndex": "0x24e",
        "removed": false,
        "topics": ["0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef", "0x0000000000000000000000028c6c06298d514db089934071355e5743bf21d60", "0x0000000000000000000000000f977814e90da44bfa03b6295a0616a897441acec"],
        "transactionHash": "0x9d3564c44c3869d9ecbe9858e6c47141ff69e167be91e023c1be850ad49799dd",
        "transactionIndex": "0x69"
    }, {
        "address": "0xfe3e6a25e6b192a42a44ecddcd13796471735acf",
        "blockHash": "0x0ac5199a9f8ce4c0fde18e4477a9cd832c248054a4eebc729fba5f51b07e0341",
        "blockNumber": "0x1363164",
        "data": "0x000000000000000000000000000000000000000000000005fd27c82cc1f3c0b0f0000",
        "logIndex": "0x15b",
        "removed": false,
        "topics": ["0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef", "0x0000000000000000000000028c6c06298d514db089934071355e5743bf21d60", "0x0000000000000000000000000f977814e90da44bfa03b6295a0616a897441acec"],
        "transactionHash": "0xae3800633f06ac9a5bc2a6beab2aac72a07fc8789400575e4cec73f6dc2ebc26",
        "transactionIndex": "0xaf"
    }, {
        "address": "0x1776e1f26f98b1a5df9cd347953a26dd3cb46671",
        "blockHash": "0xaadf982aacc3e9075f7e9bc8a14f2bb3cf77ea807f60f7b6144c7384fba12dda",
        "blockNumber": "0x1363165",
        "data": "0x000000000000000000000000000000000000000000000007dbf93d8eaeb8590000",
        "logIndex": "0x131",
        "removed": false,
        "topics": ["0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef", "0x0000000000000000000000028c6c06298d514db089934071355e5743bf21d60", "0x0000000000000000000000000f977814e90da44bfa03b6295a0616a897441acec"],
        "transactionHash": "0x2b307e230a1062d346c9b915a04c0a344170b8d17aa157h532b439c48e663f3e",
        "transactionIndex": "0x7b"
    }, {
        "address": "0xca14007eff0db1f8135f4c25b34de49ab0d42766",
        "blockHash": "0xfc06b6073baa0de058e65aa0250cf92b2e793da3f3ed64cbc81e85fc0082ac78",
        "blockNumber": "0x13633b5",
        "data": "0x000000000000000000000000000000000000000000000005451c54bc11d122ec0000",
        "logIndex": "0xc7",
        "removed": false,
        "topics": ["0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef", "0x0000000000000000000000028c6c06298d514db089934071355e5743bf21d60", "0x0000000000000000000000000f977814e90da44bfa03b6295a0616a897441acec"],
        "transactionHash": "0x53d5039e00b6fcb537df701532f8811c6fcb64971b3bd07fb43d452827fb3658",
        "transactionIndex": "0x3c"
    }]
}
```

使用 Viem 查询

```
import { parseAbiItem } from 'viem'
import { publicClient } from './client'

const logs = await publicClient.getLogs({
  address: '0xf39fd6e51aad88f6f4ce6ab8827279cfffb92266',
  event: parseAbiItem('event Transfer(address indexed, address indexed, uint256)'),
  args: {
    from: '0xd8da6bf26964af9d7eed9e03e53415d37aa96045',
    to: '0xa5cc3c03994db5b0d9a5eedd10cabab0813678ac'
  },
  fromBlock: 16330000n,
  toBlock: 16330050n
})
```

用 event 和 args 来隐藏 topcis 参数值的设定
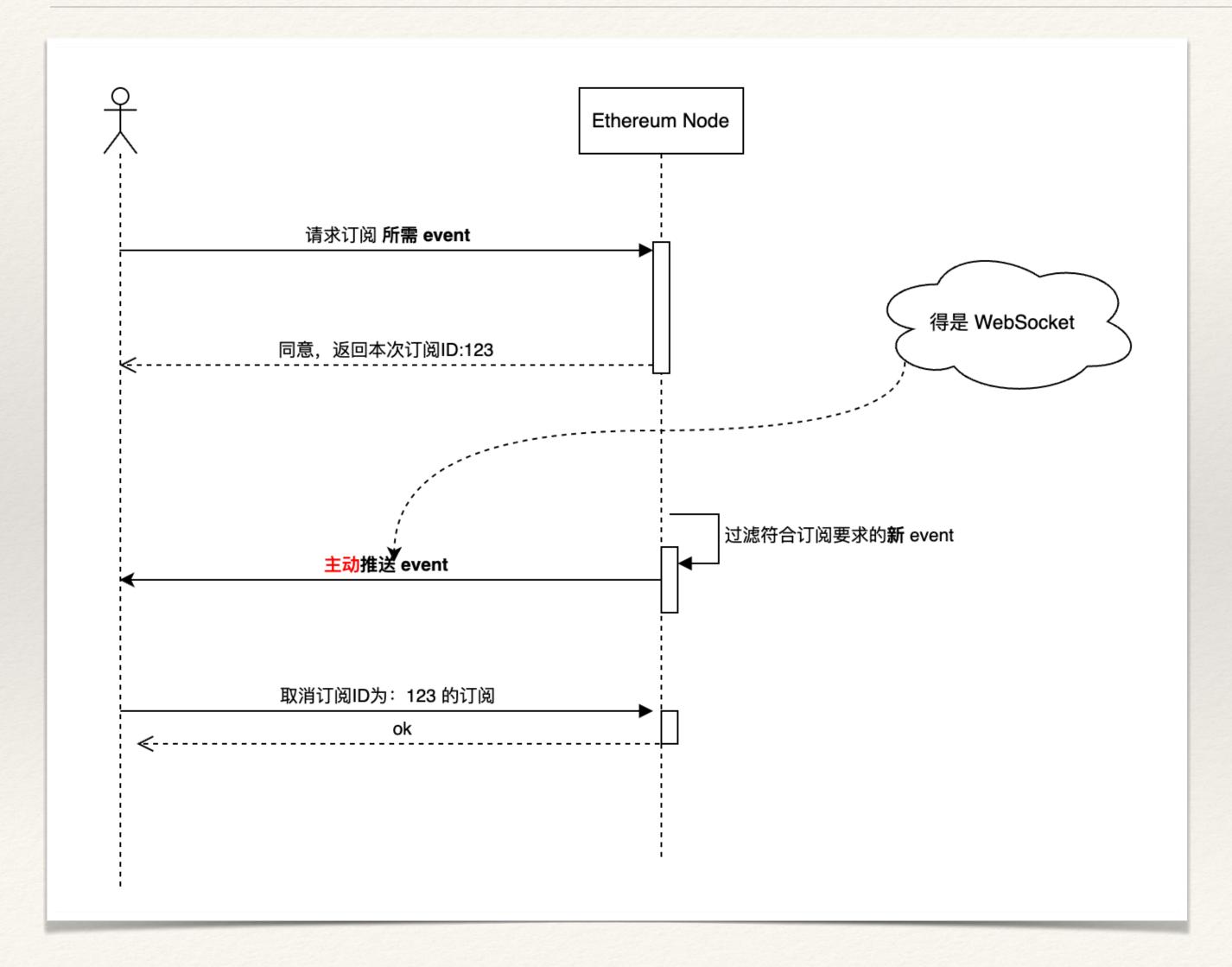
演示读取日志

Tool1 ： https://reqbin.com/post-online

Tool2: https://ethereum-json-rpc.com/?method=eth_getLogs

Logs数据采集途径

1. 通过 RPC API 从节点读取数据

1. 轮询 **Poll** ：eth_getLogs 类似于分页查询一样，偏移 fromBlock 和 toBlock 值即可

2. 订阅 **Subscribe**

2. 从第三方数据服务商拉取

订阅数据

可订阅: newHeads、logs、newPendingTransactions

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "eth_subscribe",
  "params": ["newHeads"]
}
```

```
{
  "id": 1,
  "jsonrpc": "2.0",
  "result": "0x9cef478923ff08bf67fde6c64013158d"
}
```

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "eth_subscribe",
  "params": [
    "logs",
    {
      "address": "0x8320fe7702b96808f7bbc0d4a888ed1468216cfd",
      "topics": [
        "0xd78a0cb8bb633d06981248b816e7bd33c2a35a6089241d099fa519e361cab902"
      ]
    }
  ]
}
```

Diagram text:
- Ethereum Node
- 请求订阅 所需 event
- 同意，返回本次订阅ID:123
- 得是 WebSocket
- 过滤符合订阅要求的新 event
- 主动推送 event
- 取消订阅ID为：123 的订阅
- ok

演示订阅

使用工具订阅：https://github.com/vi/websocat

> websocat $ETHEREM_RPC
{"jsonrpc":"2.0", "id": 1, "method": "eth_subscribe", "params": ["newPendingTransactions"]}
{"id":1,"jsonrpc":"2.0","result":"0x17e3271919c7fdc50000000000001c30"}
{
  "jsonrpc":"2.0",
  "method":"eth_subscription","params":
  {"subscription":"0x17e3271919c7fdc50000000000001c30",
  "result":"0xf5f51f71239778446d01202e2bfda4d73e9c5a5be2c8c0b0bf8171c45580a286"}}

作业说明

- 代码在自己的 github 提交

- 在 <u>decert.me</u> 提交领取证书

- **不可抄袭作业**，一经发现将不再检查抄袭者作业！

作业

实现语言不限，尽量使用 JS 在网页中展示

➤ 监听新区块，打印区块高度和区块哈稀值　　新区块：　20329474 (0x662022f0...cc0dea26)

➤ 实时采集并展示最新 USDT Token
（0xdac17f958d2ee523a2206206994597c13d831ec7）Transfer 流水

　　在 20329474 区块 0xe52ff...7ddcf 交易中从 0x65eDc7E1...5518AeF12 转账 50 USDT 到 0xa2D30559...30348f472

https://decert.me/quests/1267b0cc-ac39-47a7-9a4b-2ef482ff2c5c

谢谢