

解析 Transfer 事件信息

简介

我们在解析交易时往往会关注那些我们关心的事件，比如发生代币转移的 Transfer 事件。假设我们知道某笔交易的交易哈希值，那我们就可以根据交易哈希值拿到这笔交易的交易收据信息，然后从交易收据信息中取到日志信息，然后对 data 数据进行解析，就可以得到详细的转账信息了（from、to、value）。

交易收据 receipt

每一笔上链的交易都对应了一笔交易收据信息，存储了详细的交易信息，实际看起来如下所示：

[illegible]

上图告诉我们，一笔以太坊交易可以包含多个事件，不同事件之间对应了不同的日志索引。我们只关心收据信息的 logs 数据，以及 logs 中的 data 和 topics 数据。对于 Transfer 事件，topics 数组中包含了三个部分：

- topics[0] : Transfer 事件签名
- topics[1] : from 发起者地址
- topics[2] : to 接收者地址

只要是在事件中定义为 indexed 的字段都会被加入到 topics 数组中来，对于非匿名事件最多支持三个索引参数。我们看下实际的 topics 数组中数据：

```
topics: [  
  '0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef',  
  '0x0000000000000000000000000000000000000000000000000000000000000000',  
  '0x0000000000000000000000000000000000000000000000000000000000000000'  
],
```

下面我们只需要再解析一下 data 数据，就可以获取 Transfer 事件的所有有效信息了。

🌟 Event Transfer(address **indexed** from, address **indexed** to, uint256 value)

topics 中存储了事件定义中的被索引参数值；（from 和 to）

data 中存储了事件定义中非索引参数的值。（value）

解析流程

1. 准备 Transfer 事件的 ABI
2. 调用 web3.eth.abi.encodeEventSignature('Transfer(address, address, uint256)') 计算事件的签名哈希值
3. 调用 web3.eth.getTransactionReceipt(address) 获取指定交易哈希对应交易的交易收据信息
4. 调用 web3.eth.abi.decodeLog(abi, data, topic) 解析数据，并按照 [from to value] 的形式返回解析结果

Js 脚本

根据上述解析流程，可以设计如下 Js 脚本解析指定交易哈希对应交易的所有 Transfer 事件。

```
1 // import web3.js  
2 const Web3 = require("web3");  
3 // initiate web3 instance (connected to hosted mainnet node)  
4 const web3 = new Web3("https://cloudflare-eth.com");  
5  
6 // Transfer Event Abi  
7 const eventAbi = [{  
8     type: 'address',  
9     indexed: true,  
10    name: 'from'  
11  }, {  
12    type: 'address',
```

```

13     indexed: true,
14     name: 'to'
15   }, {
16     type: 'uint256',
17     indexed: false,
18     name: 'value'
19   }];
20
21 // transfer 事件签名, 作为 topics[0]
22 const eventSign = web3.eth.abi.encodeEventSignature('Transfer(address,address,uint256)');
23
24 web3.eth.getTransactionReceipt('0xa219bcdce7d79138cf6c2d8e7bbe500bceb16f377d9a9a1c1e22340e4e755fae7e509ecd06c')
25 .then((res) => {
26   let logs = res.logs; // 交易收据的 logs
27   for (let i = 0; i < logs.length; i++) {
28     if (logs[i].topics[0] == eventSign) { // 如果是 Transfer 事件
29       // 解析 data , 参数为: 事件 abi、待解析的数据 data、事件中两个被索引的参数索引
30       let decodedata = web3.eth.abi.decodeLog(eventAbi, logs[i].data, [logs[i].topics[1], logs[i].topics[2]]);
31       console.log('id:', logs[i].id);
32       console.log('from:', decodedata[0]);
33       console.log('to:', decodedata[1]);
34       console.log('value:', decodedata[2]);
35       console.log();
36     }
37   }
38 })
39 .catch((error) => {
40   console.error(error);
41 })

```

运行结果

```

$ node transaction/transaction_receipt.js
id: log_383ef54f
from: 0x9BD48e08E3444B30113812acd52458b35C33803A
to: 0xe65cdB6479BaC1e22340E4E755fAE7E509EcD06c
value: 1000000000000000000

id: log_d30bea66
from: 0xe65cdB6479BaC1e22340E4E755fAE7E509EcD06c
to: 0x9BD48e08E3444B30113812acd52458b35C33803A
value: 48496843

```