链上账户与交易过程

一、链上账户

什么是（区块链）钱包?

区块链钱包（以下简称钱包）是让人们能够管理自己的区块链账户，并且与去中心化应用（dApp）交互的工具软件。

显而易见：钱包是区块链的应用入口。

也许你已经注意到，每个公链和每个dApp（去中心化应用），都至少拥有一种具有货币属性的专属通证（token）。我们通常把这个具有货币属性的通证叫做代币。代币有多种作用，譬如安保、奖励、估值等等。第四章第三节将会有详细的介绍。

具体到日常应用，当我们将智能合约部署到以太坊，我们需要通过钱包从自己的账户里支付gas（手续费）。当我们到一个去中心化交易网络里交易代币的时候，我们既需要钱包从自己的账户里转出要卖出的代币接收要买入的代币，还需要用钱包从自己的账户里支付完成这些操作的gas（手续费）。这就是说，在我们使用每个公链和每个dApp的时候，常常需要动用到代币。由于代币是在我们的账户里，要通过钱包才能动用账户，因此这些操作就常常需要钱包才能完成。

大多数区块链钱包都非常像银行，它管理的是你个人的，多个人的，一组或多组人的账户，并且一个人可以在一个钱包里生成、导入并使用多个账户。

但与银行不同的是，钱包和账户并非绑定关系：一个账户可以在多数钱包里使用。你也可以在一个钱包里，拥有任意多个账户。

还有：没有人能够冻结你的账号，撤销你的转账，也没有消费限额……并且无论对方在地球的哪个角落，转账的确认时间都与地域无关。

根据载体的不同，区块链钱包分为计算机钱包、手机钱包、浏览器钱包、硬件钱包、纸钱包和脑钱包等等，这样不管您使用的是何种设备，或者有什么样的需求，都有配套的钱包供您使用。

其中脑钱包因为实际使用时安全性太低，基本已经被淘汰。

硬件钱包是为了离线生成和保存私钥，是目前“公认”比较安全的代币存储方案。代表品牌有Leger、Ballet Cryptocurrency Wallet（芭蕾钱包）、imKey硬件钱包、Trezor。

浏览器钱包的代表为Metamask、Jaxx。

手机钱包的代表为Trust Wallet、MetaMask、imToken、TokenPocket。

因为使用方便、用户群体大，手机钱包已经成为主流。计算机钱包的特点则刚好相反。

以太坊账户

以太坊的账户模型则跟我们通常所理解的账户概念是一致的。

一个以太坊帐户是一个可以拥有 ETH 或其它通证余额的实体，可以在以太坊上发送交易。并且其中的通证包括但不限于代币。

帐户可以由用户控制，也可以作为智能合约部署——这样以太坊便有了两种类型的账户：

- 外部所有账户（常被简称为**外部账户**。Externally Owned Accounts，EOA）：人类常用的存储自己的通证的账户。外部账户由私钥（或者说私钥的所有者）控制。
- 合约账户（Contract Accounts，CA）：部署到网络上的智能合约。它是只受智能合约代码控制的账户。

这两种账户都能：

- 接收、持有和发送 ETH 和通证（token）。
- 与已部署的智能合约进行交互。

两种账户的主要区别

外部账户

- 创建帐户是免费的
- 可以发起交易
- 外部帐户之间只能进行 ETH 和通证交易
- 由三部分组成：地址、公钥和私钥。公钥和私钥是一对加密密钥，它们联合控制帐户的活动。

合约账户

- 创建合约存在成本，因为需要使用网络存储空间。
- 目前只能在收到交易时发送交易。
- 从外部帐户向合约帐户发起的交易能触发可执行多种操作的代码，例如转移代币甚至创建新合约。
- 合约帐户没有私钥。 它们由智能合约代码逻辑控制。因为合约代码就是合约账户的组成部分，你可以理解为合约账户目前是由外部请求驱动的，根据代码逻辑和状态（数据）自运行的账户。

比较	外部账户	合约账户
拥有私钥	是	否
codeHash内容	为空	非空
主动发起交易	是	否，只能被动发起交易
拥有余额	是	是
地址长度	20字节	20字节

外部账户之组成：助记词、私钥、公钥和地址

以太坊账户通常是以上四个部分的复合体。

首先，你的各种“币”，都在你的钱包地址里。你给某个地址转账时，钱包会动用你的私钥对交易进行签名，在公钥的配合下广播这笔交易。

私钥、公钥和地址这三者长得相似，且也密切相关：私钥是钱包根据密码学原理（你可以理解为某个随机法则）生成；私钥通过特定的密码学原理生成公钥（至2023年8月5日，比特币和以太坊外部账户地址，使用的都是椭圆曲线数字签名算法↗），公钥再采用另一个加密算法生成地址。这两种加密算法都保证了只能产生唯一的公钥和地址，且不能反向推算，即不能由地址破解出公钥，也不能由公钥破解出私钥。

这里最重要的显然是私钥。以太坊软件使用底层操作系统的随机数生成器生成256位熵（伪随机数种子），通过 Secp256k1 椭圆曲线算法产生一个256位数字私钥（同比特币的私钥，是一组64位的16进制字符，即32字节），基本上与“选择1到 2^256 之间的数字”相同——这也就是说，以太坊的私人密钥空间的大小为2^256，是一个难以置信的大数目。十进制大约是10^77。可见宇宙估计含有10^80个原子。

私钥固定长度为256位，使用十六进制表示就是64个字符：

6954ac6d0402d7239f1cc150da224d0ef08fd1226f245f06fe4d6d68accfce8a

你可以使用硬币随机挑选你的私钥：投掷硬币256次，得到可以在以太坊钱包中使用的随机二进制数字作为私钥。那么恭喜你，你就是个书呆子。好消息是，书呆子们正在颠覆旧世界。

私钥貌似足够安全，但实际应用中总是有人会自作聪明犯下大错：

比特币和以太坊获得私钥的技术手段都是类似的；之后使用某个椭圆曲线算法将私钥转换为公钥，公钥再通过哈希算法和Base58Check编码转换得到地址。

有了某个地址的私钥，您才能使用它收款或转账。

那助记词又是咋回事呢?

很简单，因为私钥记不住，非常不便于使用，基本上只能靠拷贝来保存和移动，也就非常危险（譬如从电脑移动到手机，很多人就利用云工具，结果导致被盗）。所以 BIP39协议 通过加密算法，将一组英文单词和一个私钥形成单向对应的关系，以方便用户备份和在不同设备间使用。该方法诞生于比特币时代，后来又发生一些算法上的变迁，因此这组英文单词，通常为12~24个英文单词。

二、区块链交易流程

交易——区块链系统的核心，负责记录区块链上发生的一切。区块链引入智能合约后，交易便超脱『价值转移』的原始定义，其更加精准的定义应该是区块链中一次事务的数字记录。无论大小事务，都需

要交易的参与。

交易的一生，贯穿下图所示的各个阶段。本文将梳理交易的整个流转过程，一窥FISCO BCOS交易完整生命周期。

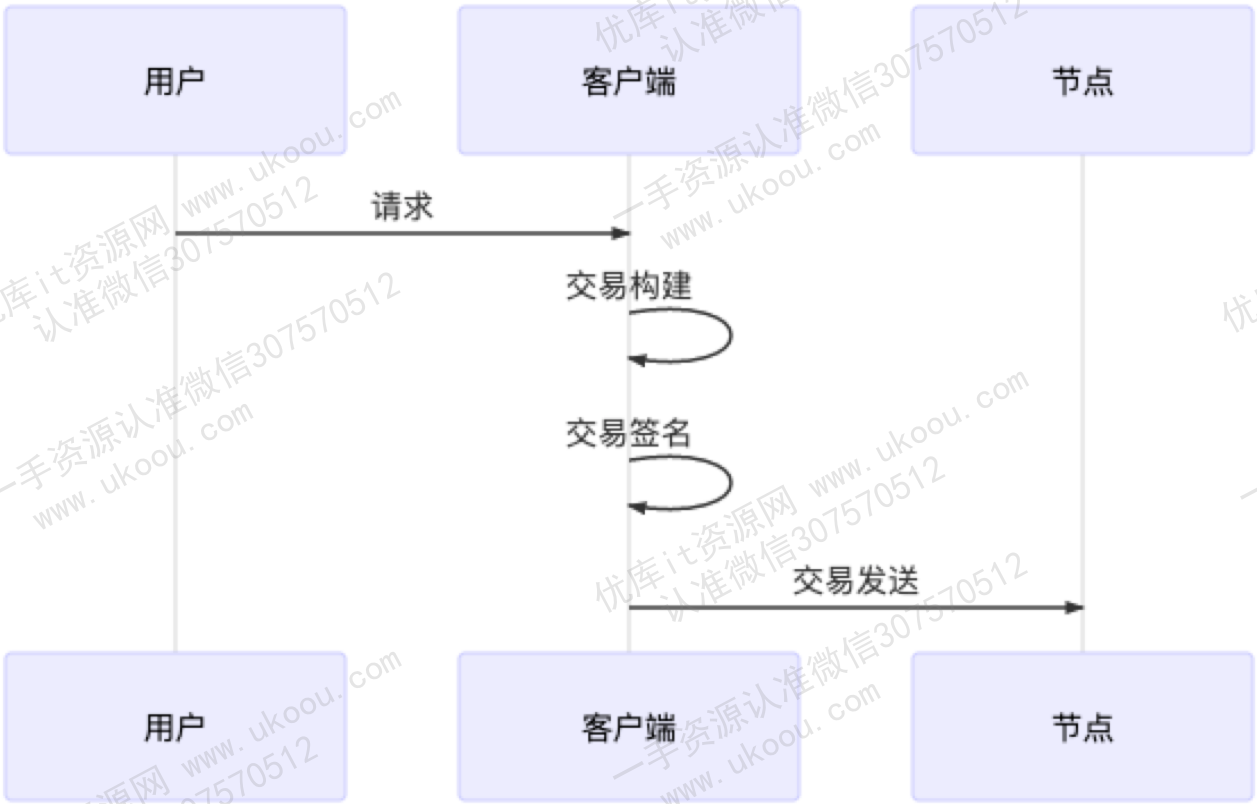
交易生成

用户的请求给到客户端后，客户端会构建出一笔有效交易，交易中包括以下关键信息：

1. 发送地址：即用户自己的账户，用于表明交易来自何处。
2. 接收地址：FISCO BCOS中的交易分为两类，一类是部署合约的交易，一类是调用合约的交易。前者，由于交易并没有特定的接收对象，因此规定这类交易的接收地址固定为0x0；后者，则需要将交易的接收地址置为链上合约的地址。
3. 交易相关的数据：一笔交易往往需要一些用户提供的输入来执行用户期望的操作，这些输入会以二进制的形式被编码到交易中。

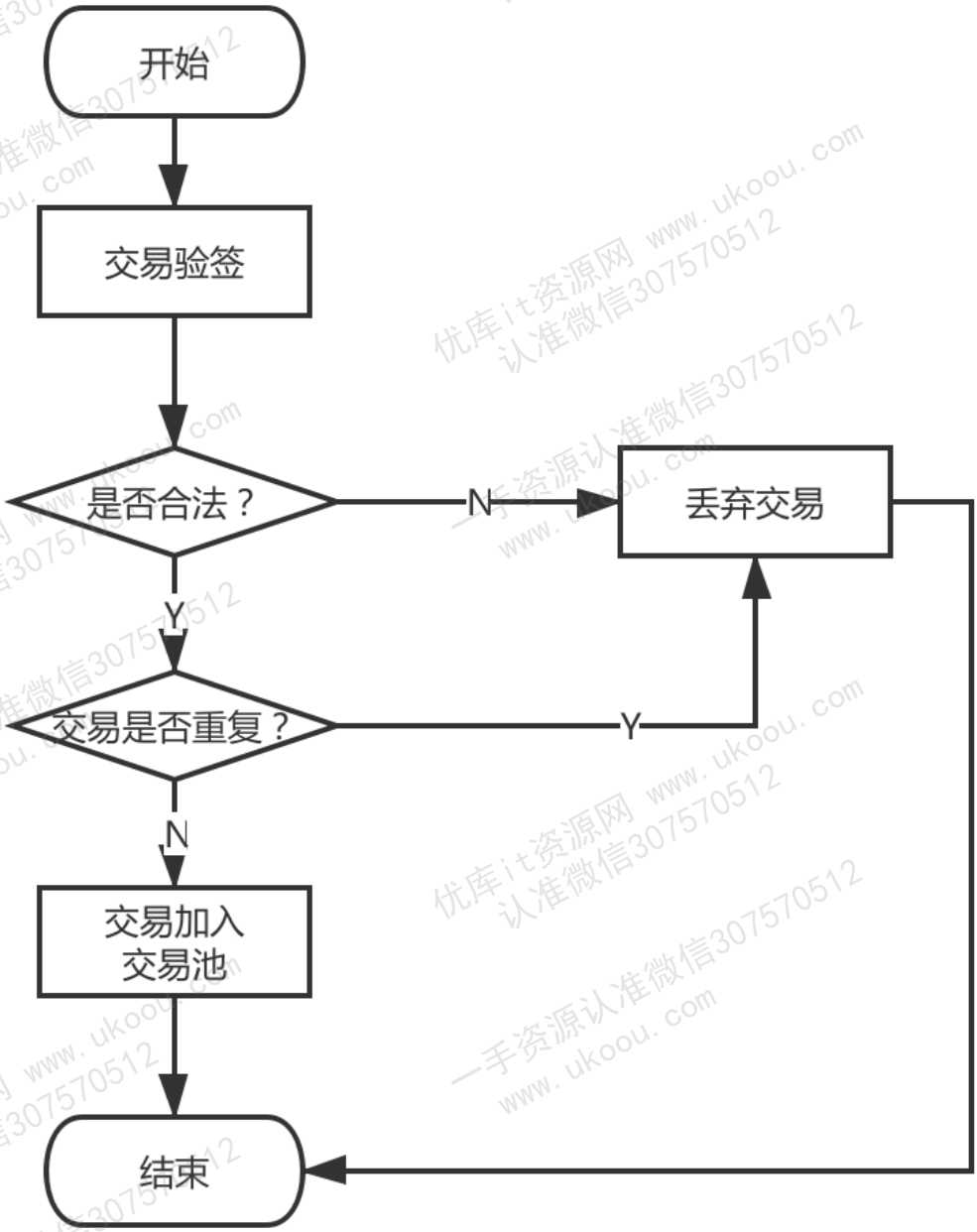
4. 交易签名：为了表明交易确实是由自己发送，用户会向SDK提供私钥来让客户端对交易进行签名，其中私钥和用户账户是一一对应的关系。

之后，区块链客户端会再向交易填充一些必要的字段，如用于防交易重放的交易ID及blockLimit。交易的具体结构和字段含义可以参考编码协议文档，交易构造完成后，客户端随后便通过Channel或RPC信道将交易发送给节点。



交易池

区块链交易被发送到节点后，节点会通过验证交易签名的方式来验证一笔交易是否合法。若一笔交易合法，则节点会进一步检查该交易是否重复出现过，若从未出现过，则将交易加入交易池缓存起来。若交易不合法或交易重复出现，则将直接丢弃交易。



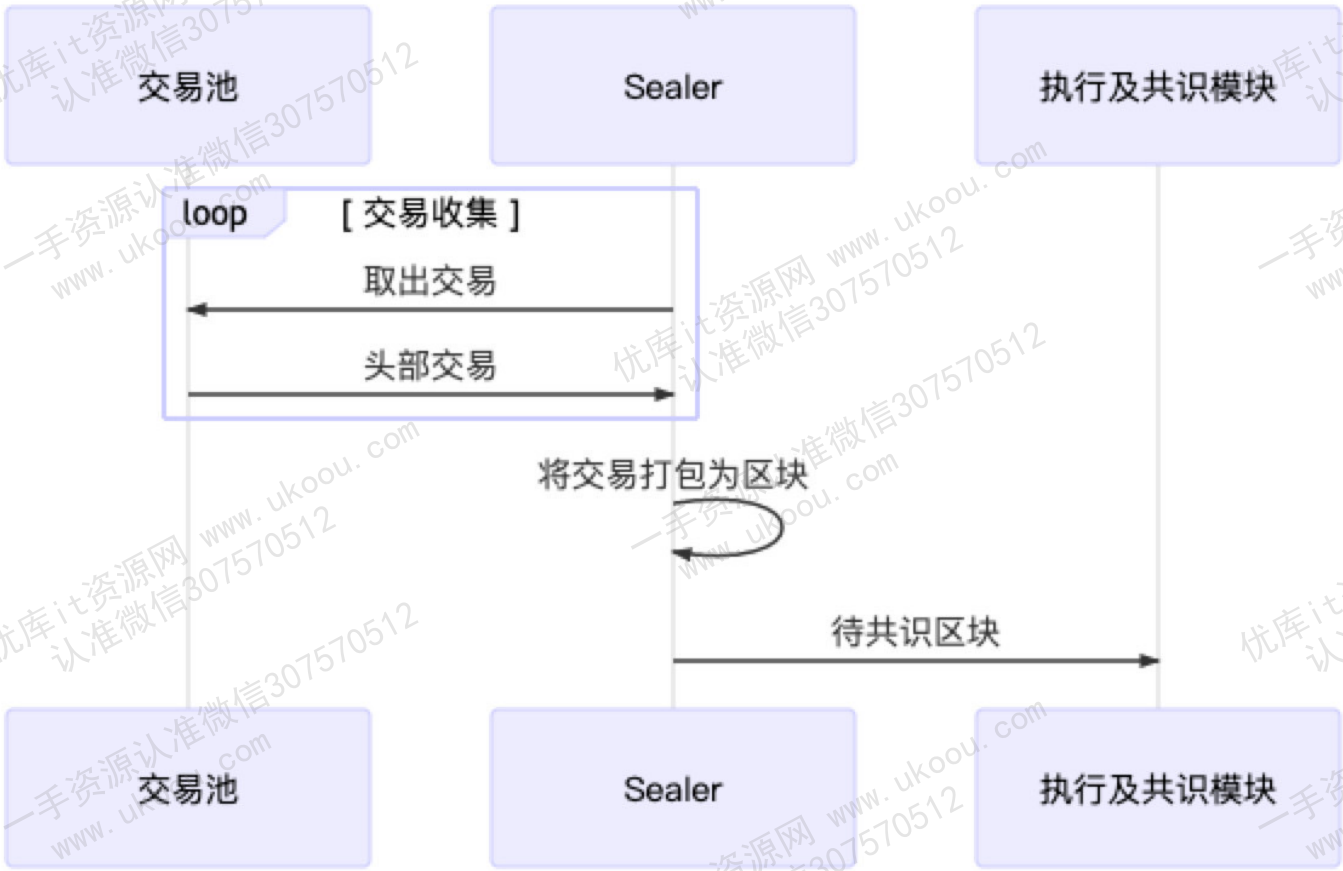
交易广播

节点在收到交易后，除了将交易缓存在交易池外，节点还会将交易广播至该节点已知的其他节点。

为了能让交易尽可能到达所有节点，其他收到广播过来的交易节点，也会根据一些精巧的策略选择一些节点，将交易再一次进行广播，比如：对于从其他节点转发过来的交易，节点只会随机选择25%的节点再次广播，因为这种情况一般意味着交易已经开始在网络中被节点接力传递，缩减广播的规模有助于避免因网络中冗余的交易太多而出现的广播风暴问题。

交易打包

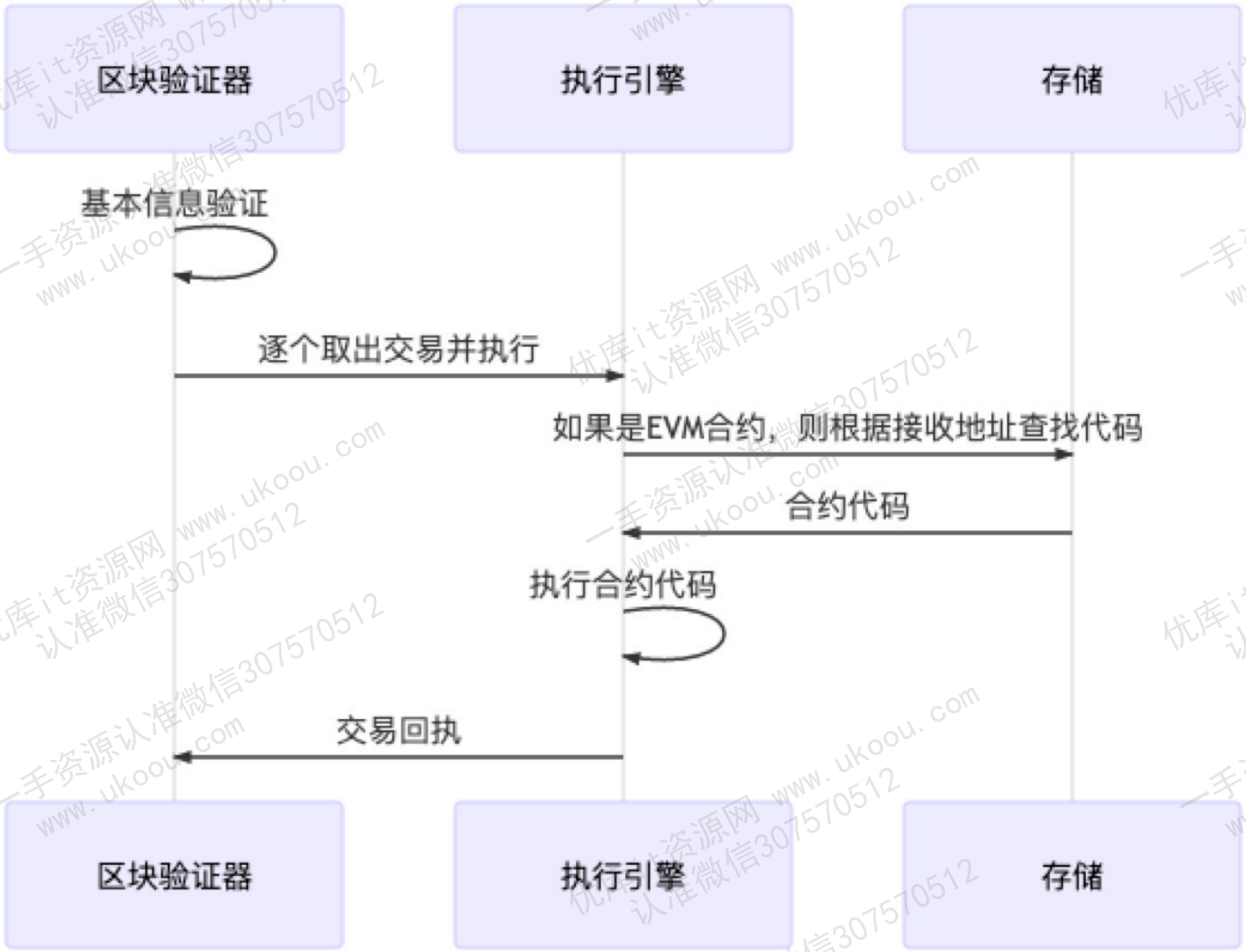
为了提高交易处理效率，同时也为了确定交易之后的执行顺序保证事务性，当交易池中有交易时，Sealer线程负责从交易池中按照先进先出的顺序取出一定数量的交易，组装成待共识区块，随后待共识区块会被发往各个节点进行处理。



交易执行

节点在收到区块后，会调用区块验证器把交易从区块中逐一拿出来执行。如果是预编译合约代码，验证器中的执行引擎会直接调用相应的C++功能，否则执行引擎就会把交易交给EVM（以太坊虚拟机）执行。

交易可能会执行成功，也可能因为逻辑错误或Gas不足等原因执行失败。交易执行的结果和状态会封装在交易回执中返回。



交易共识

区块链要求节点间就区块的执行结果达成一致才能出块。FISCO BCOS中一般采用PBFT算法保证整个系统的一致性，其大概流程是：各个节点先独立执行相同的区块，随后节点间交换各自的执行结果，如果发现超过2/3的节点都得出了相同的执行结果，那说明这个区块在大多数节点上取得了一致，节点便会开始出块。

交易落盘

在共识出块后，节点需要将区块中的交易及执行结果写入硬盘永久保存，并更新区块高度与区块哈希的映射表等内容，然后节点会从交易池中剔除已落盘的交易，以开始新一轮的出块流程。用户可以通过交易哈希等信息，在链上的历史数据中查询自己感兴趣的交易数据及回执信息。