以太网系统上实际存在两种类型的帐户。 一个由私钥控制的外部帐户(externally-owned account，EOA
)，例如，在我们使用的钱包中的帐户，这些帐户每一个都有自己的余额。 所有者可以通过创建交易并对其签名，从自己的外部帐户发送消息。另一个是由部署在区块链中的代码控制的合同帐户(contract account
)、以及存储在智能合约帐户(有时称为智能钱包)中的以太网虚拟机代码进行控制。合同帐户收到信息后，将启用内部代码，可以读取和写入内部存储器，以及创建新合同等操作。 根据现在的以太坊合同中选择所需的族。
只有外部帐户才能启动交易，只有帐户所有者才能更改帐户状态。
什么是账户抽象？ 帐户抽象是对这两个帐户的改进，试图模糊两者之间的界限，使之成为包含复杂逻辑的通用帐户允许帐户同时具有合同帐户和外部帐户功能。 这与允许用户根据合同帐户的格式定义外部帐户相同，用户可以在智能合同钱包中包含逻辑验证。 用密钥管理的账户也能得到代码的支持。
账户抽象的各种方案一直以来，实现账户抽象是以太网开发者社区的愿景。
社区也提出了EIP-86、EIP-2938等各种方案。 EIP-86是账户抽象的技术准备定义允许用户创建基于智能合同的帐户的新帐户类型。
以太网协议本身要求将所有内容打包到来自ECDSA安全外部帐户(EOA
)的交易中，而每个用户的操作必须通过来自EOA的交易进行打包这将产生21000 gas的费用。 用户需要向单独的EOA持有ETH支付gas。 EIP-86提出的帐户抽象带来了新类型的事务，与传统事务必须具有EOA作为发件人相比这些事务没有发件人。 这种事务破坏了事务散列的唯一性。 EIP-86原本计划在Metropolis阶段升级，但由于上述问题，开发者决定将部署搁置在Metropolis上。 EIP-2938提供了一种帐户抽象解决方案，通过修改以太网协议的一部分，合同帐户可以像外部帐户一样开始交易。
但是，该方案需要在协议层进行以太网协议的变更，因此没有被广泛接受。 后来提出的新协议ERC-4337试图提供一种不需要更改一致协议的方案来达到与EIP2938相似的效果，这种更安全的实现方式现在在社区中引起了更多的关注。
ERC-4337如何实现？
ERC-4337不试图修改协议，而是将mempool功能复制到系统中。
用户发送用户交互对象，包括用户的意图、签名和其他数据。 用户操作具有单独的mempool存储池，连接到该存储池的节点可以进行特定于ERC-4337的身份验证，并过滤操作以确保只接收付费操作。 矿工或使用Flashbots服务的打包人员批量收集这些用户交互，将其打包为一个捆绑交易(bundle transaction
)，并嵌入以太网区块。 打包者为以太坊内的捆绑交易支付gas
fee，并索取和补偿为每个单独的用户操作支付的费用。
打包程序使用费用优先级逻辑来选择要包括的用户操作对象。 的用户交互UserOperation看起来像一个事务，但它是ABI编码的结构包含以下字段：
1、发件人：进行操作的钱包； 2、nonce和signature
:传递给钱包验证函数以便钱包验证操作3、initCode

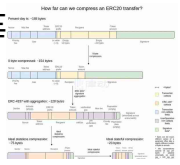
:如果钱包还不存在、用于创建钱包的初始化代码； 4、callData
:用于实际执行步骤调用钱包的数据。
每个wallet都是智能构建器，并且必须包括两个功能函数：
1、validateUserOp接受用户操作作为输入。 此函数验证用户操作的签名和nonce，如果验证成功，则需要支付费用并增加nonce如果验证失败，则抛出异常；
2.op执行函数，用于将calldata解析为钱包执行操作的一个或多个命令。
ERC-4337变化如果此方案被普遍采用，签名验证将迁移到以太网虚拟机(EVM)，validateUserOp函数添加了任何签名和随机数验证逻辑，使验证逻辑更灵活。
由此，在交易上签名时可以采用新的密码学工具，钱包也可以提供新的功能例如，
1、多重签名； 2、社会康复3、更高效、更简单的签名算法(如Schnorr、BLS)；
4、后量子安全签名算法(例如，Lamport、Winternitz)； 5、可升级钱包。
该计划还启动了各种其他交易许可证管理，包括允许通过智能协议支付gas费用。
目前，外部钱包通过以太网交换的gas
fee只能通过钱包的ETH支付如果钱包里只有ERC-20
Token，没有ETH，就不能转出这些Token。 ERC-4337被采用后，用户可以使用账户的ERC-20Token来支付费用、矿工节点以合同为中介代理ETH链支付，获取用户的ERC-20Token。
抽象实现后，外部账户所有者签署和广播交易将不再是发起交易的唯一方式。
这带来了以太坊作为原交易的中转者的可能性。 目前，许多以太网上的APP应用都需要依赖中继在区块链上发布用户事务并向中继支付费用。 如果能在钱包里编入更复杂的合同，一些转播者就不需要存在了也没有必要向他们支付额外的费用。
虽然有很多优点，但新方案也同样面临一些问题。
最突出的是更高的Gas成本，基本的ERC-4337操作大约需要42000
gas另一方面，正常交易需要21000 gas。 原因是：
1、EOA需要支付大量的单个存储读/写成本这些成本将与21000
gas的支付捆绑在一起。 (1)编辑包含pubkeynonce(5000 )的存储slot；
)2)用户交互调用数据成本(约4500通过压缩可以减少到约2500；
)3)电恢复器(~3000)； (4)首次访问钱包本身(2600 )5)首次访问收款人帐户(2600
)6)将ETH转入收款人帐户(~9000 )7)编辑存储以支付费用(~5000
)8)代理然后访问代理自身() (~2600)； 2 )除了上述存储读/写成本外，合同还必须执行“业务逻辑”(例如，用户操作解包、散列化、变量洗牌) (

)。

3、为了支付日志费用需要消费gas (eoa不发布日志)；

4、一次性合同创建成本(约32000 gas，代理中每个代码字节32000
gas除此之外，设置代理地址20000 gas )简单来说，帐户抽象地址的每一步都需要计算，需要消耗更多的资源，还需要额外的费用。 好在这并不是没有解决办法。

因为Rollup擅长数据压缩，与数据复杂的账户抽象方案有天然的契合点。
Vitalik的最新方案提出用第2层处理帐户抽象产生的数据。 改进之处是将只能逐步实现的功能打包成批量交易同时用SNARK技术保证交易的有效性。 结语在以太坊重点发展第2层的框架已经确定的今天，Vitalik开始将以太坊升级的后续计划转移到客户抽象。 最新方案显示了rollup帐户的抽象技术路径。
每个Rollup提供程序也发布了兼容帐户抽象的新版本。 今年6月，zkSync发布了V2更新信息：添加“帐户抽象”功能提高与以太坊EVM的兼容性。
10月，ERC-4337发布了新版本，添加了签名聚合功能，包括BLS签名算法。
签名聚合还允许作者和批处理发件人聚合签名。 例如，BLS、SNARKs等



大幅减少链上的数据，降低rollups的数据成本。

我们有理由相信，账户抽象带来的变化也同样蕴藏着生态爆发的可能性。 随着Rollup的发展，能够与Rollup结合的账户抽象也一定能发展出更好更精细的方案。