为什么“链抽象”会成为“模块化”之后的热门叙事？

撰文：Haotian

为什么「链抽象」会成为「模块化」之后的热门叙事？除了 Mass Adoption 这种看似「假大空」的目标之外，真正的内核原因我认为如下：

1）链抽象是模块化到一定阶段之后的必然，因为区块链技术架构就大致包括共识层、执行层、DA 层、可交互操作性、结算层等等。传统意义上 monolithic 单一链要协调好各个分层综合强化，而模块化打破了这一局面。

DA 层、可交互操作性层甚至执行层等部分都可以被单拎出来进行性能优化并以可组合性的协作方式发挥作用。因此，「模块化」成了开发者寻求链构建差异化的主流范式，这种成本低且叙事性强关键是还不用过于忧心生态运营等长期问题的创业方法论逐渐成了主流。

然而，模块化本质上是以 infra 基建为主要目标，当过量开发力量过于聚集在模块化方向，就会造成 infra 和应用之间的失衡，导致基建多于应用的尴尬局面出现。

显然，「链抽象」的出现正是为了为此尴尬局面破局，将被模块化撕裂的区块链组件重新整合起来，提升体验级，降低用户 Onboard 的门槛，进而推动应用的增长。某种意义上，链抽象会补足模块化带来的生态和应用失衡，成为下一阶段必不可少的叙事。

2）链抽象本身也只是一种泛泛的概念，其大致可分为三个具体落地方向：

1、做工具服务促进异构链的兼容，目标可以无缝集成不同的区块链环境，增加服务本身的应用占比。比如， @chainlink

的 CCIP；

2、做中间链统一流动性成为聚合调度中心，主要汇聚不同链上的分散资产，用更安全、透明化的跨链信息传递来构建一个统一流动性层（资产跨链、消息传递）。比如， @0xPolygon 的 agglayer，@ProjectZKM 的 Entangled Rollup Network 等；

3、做适配各个异构链并推出统一的用户交互体验层，在一系列工具或服务兼容适配的基础上，构建统一的合约调度中心，去中心化 Solver 服务协调中心，以及其他统一 Gas 代付、社交登陆、社交恢复等用户体验，以此降低用户的参与门槛，引流更多用户入圈。比如, @NEARProtocol 发起的 BOS 操作系统，@dappOS\_com 面向 GMX 等热门应用推出的意图 Solver 执行网络，@ParticleNtwrk 的全链账户抽象和 intent fusion 解决方案等等；

应该讲，这些有的偏技术后端服务，有的偏用户前端体验升级，都是为了统一分散的区块链单链环境，用全链无缝交互和体验升级来给区块链应用市场扩充增量用户。

3）「链抽象」赛道更易受 VC 青睐，成为 web2 和 web3 融合的关键。最早，以太坊 ERC4337 带火了一批账户抽象钱包开发热潮，主要原因在于，一批 web2 的互联网高管人才携融资进场。这是他们初入 web3 最容易看懂且能发挥专长的领域。链抽象在账户抽象基础上进一步丰富了业务范围，但依旧是一些 web2 丰富经验和背景的人的主要发力方向。

相比模块化方向的纯概念导向，链抽象技术更偏底层，团队需要用实在的行业资源聚合和资产 TVL 沉淀以及用户数据增长累积。

相比动不动以「xxxx as a service」来勾勒商业空间的模块化方向，VC 更容易给出相对客观的估值，用户也能透过客观数据看到链本身的价值增长，从而在二级市场有更好的投资估值参考。