Move双子星：Sui和Aptos如何挑战区块链格局

作者：YBB Capital研究员Zeke，编译：0xjs@本站

## 前言

市场最近变得越来越停滞不前，导致该领域的许多 OG 开始质疑该行业的目的。

我想就此分享一些个人想法。我一直认为，过去的许多宏伟愿景都被“揭穿”，因为它们从一开始就没有逻辑连贯性。非金融 Dapp 经常试图通过强调去中心化的价值来掩盖其缺点。但现实是，他们要求我相信他们的多重签名钱包和单节点服务器，而不是谷歌、推特或 YouTube，声称它们足够安全。许多愿景并没有被揭穿；它们只是从未真正经过考验。我仍然相信，即使这些愿景并不像最初想象的那么宏伟，它们仍然具有重要意义——它们只是需要一个强大的基础来支持它们。至少，它们应该提供去中心化或与 Web2 相当的体验。

以 TON 和 Solana 为例，它们曾经被低估，但现在正逐渐在各个方面追赶行业领导者。支持应用的区块链需要创新，创新在每个周期都会推动行业向前发展。今天，我们将探索一种长期被忽视的区块链——基于 Move 的区块链。

## 1、Move

Move 编程语言最初是为 Meta 放弃的项目 Diem（最初称为 Libra）开发的。Diem 旨在创建一种更稳定、更受监管的稳定币，作为 Meta 元宇宙愿景的基础。然而，该项目面临着全球监管机构的强烈反对和无情压力。监管机构担心，Diem 的规模加上 Facebook 庞大的用户群，可能对金融稳定、货币政策和数据隐私构成威胁。在压力之下，尤其是来自拜登政府的压力，Meta 最终不得不放弃 Diem 项目。

好在 Diem 的核心并没有被完全抛弃，从原团队中分裂出来的各种派系继续探索和发展 Move，后来演变成我们熟知的 Move 双子星：Sui 和 Aptos。除此之外，还有一些新兴项目，比如 Linera（受 Move 启发而开发的 Rust 区块链）和最近得到大力推广的 Movement。

那么，一个被砍掉一半的项目，为何留下了如此深远的影响呢？Move 作为一门由顶级 Web2 公司为区块链开发的编程语言，其复杂程度可谓是相当高，它的设计充分考虑了现有区块链编程语言，尤其是 Solidity 的性能和安全问题，其设计目标是打造一套专门为资产管理和访问控制而量身定制的类型系统。我把它的优点简单概括为三点：

· 安全性：Move 语言的首要设计原则是安全性。它使用静态类型检查和资源管理来防止常见的安全漏洞，例如溢出错误和重入攻击。与其他语言虚拟机相比，Move 支持各种安全功能，如下面的 Nansen 比较图所示。



· 可组合性：Move支持模块化和可组合性，允许开发人员轻松创建和组合不同的智能合约，从而构建更复杂的应用程序。

· 性能：Move语言的虚拟机经过优化（支持并行性、内存管理和编译器优化），使其能够高效地执行智能合约，从而提高交易速度和吞吐量。

在充斥着模块化 EVM 区块链的市场中，Move 代表着一项大胆的实验。虽然上述观点在其他区块链项目的描述中可能看起来很熟悉，但我强烈建议你亲自体验它们，以充分掌握这些功能的实际好处。

## 2、Sui



### 2.1 架构

作为双子星之一，Sui 自推出以来就饱受批评，尤其是关于空投和代币分发方法。然而，抛开这些问题，专注于项目本身，Sui 已证明在性能和用户体验方面都非常出色，尤其是在游戏方面。这一成功很大程度上归功于其创新的架构，该架构已针对主流采用进行了改进。以下是 Sui 架构创新的简要概述：

对象存储模型：该组件是 Sui 对 Move 改进的核心。对象存储模型将数据视为独立对象，每个对象都有唯一标识符。与传统数据库系统不同，对象存储模型没有固定的数据结构，可以存储各种类型的数据，例如文本、图像、视频和音频。该模型允许并行执行和水平扩展（添加节点以扩展存储容量），Sui 的设计都围绕这一模型展开。

因果排序：确保交易的执行顺序与其因果关系一致，避免数据冲突和不一致。此功能使 Sui 能够处理大量并发交易，同时保持数据一致性。

Narwhal 和 Bullshark 共识引擎：Sui 使用 Narwhal 和 Bullshark 作为其共识引擎。Narwhal 负责交易排序和验证。它的工作原理是维护本地交易池，根据因果关系对交易进行排序，并广播它们以确保所有节点都具有相同的有效交易顺序。Bullshark 在收到 Narwhal 的有序交易列表后，对该列表进行投票，并使用拜占庭容错 (BFT) 共识来确保所有节点都同意交易顺序。

Sui Move：Sui 通过添加新功能扩展了 Move 语言，例如对 NFT、资产管理和数据存储的支持。

Sui 框架：Sui 提供了一个全面的框架，帮助开发人员快速构建和部署应用程序。该框架包括各种工具和库，例如 Sui 钱包、Sui SDK 和 Sui CLI。

Sui 的架构设计使其能够处理大量并发交易，同时保持高速、低费用和安全性。此外，Sui Move 语言和 Sui 框架为开发人员提供了强大的工具来构建安全、可扩展且用户友好的应用程序。

### 2.2 共识

Sui 区块链使用一种名为 Mysticeti 的共识机制，这是一种基于拜占庭容错 (BFT) 的共识，旨在优化低延迟和高吞吐量。

Mysticeti 允许多个验证者并行提议区块，从而最大限度地提高网络带宽并抵抗审查。此外，该协议只需要三轮消息传递即可从有向无环图 (DAG) 提交区块，符合最低理论要求并与 pBFT 并行。提交规则允许并行投票和区块领导者认证，从而进一步减少中位延迟和尾部延迟。它还可以容忍不可用的领导者，而不会显着增加提交延迟。

在 Sui 主网启动之前，Mysticeti 在测试网上进行了三个月的测试，取得了显著的成果，包括延迟减少了 80%。现在，Sui 网络每秒可以处理数万笔交易，端到端延迟远低于一秒。

Sui 区块链还采用了一种特定类型的权益证明共识，即委托权益证明 (DPoS)。当发生涉及共享对象的复杂交易时，Sui 使用 Narwhal 和 Bullshark 共识引擎对这些交易进行排序。与区块链使用的其他 BFT 共识机制相比，Sui 的共识具有以下优缺点：

优点：

低延迟和高吞吐量：Mysticeti 协议通过并行提议区块和优化消息传递过程，显著降低了共识延迟并提高了网络吞吐量。这使得 Sui 区块链能够每秒处理数万笔交易，端到端延迟远低于一秒。

抗审查：Mysticeti 允许多个验证者并行提出区块，从而增强网络的抵抗审查的能力。

对不可用领导者的容忍度：提交规则允许容忍不可用的领导者（当领导节点发生故障时，系统会自动选举新的领导者），而不会显著增加提交延迟。

缺点：

复杂性：Mysticeti 协议的设计相对复杂，需要更深入的技术理解才能完全掌握其运行机制。

安全性：虽然Mysticeti协议在测试网上表现良好，但其安全性需要在实际应用中进一步验证。

可扩展性：Mysticeti 协议的可扩展性仍需要进一步观察，以确保它能够适应未来不断增长的网络规模和交易量。

### 2.3 账户抽象

Sui 的账户抽象模型是一种让用户以更简单、更安全的方式管理账户和交易的机制。它从底层区块链协议中抽象出账户和交易逻辑，从而实现更高级别的账户管理和交易处理。

在 Sui 的账户抽象模型中，账户不再是简单的公钥-私钥对，而是具有更丰富属性和行为的对象。每个账户都有一个唯一标识符，称为账户 ID，与账户的公钥和私钥对相关联。

Sui 的账户抽象模型的关键组成部分包括：

1、账户对象：Sui中账户的基本单位。每个账户对象都有唯一的Account ID，并包含账户的属性和行为。

2、账户数据 (Account Data)：账户对象的核心组成部分，包括账户ID、公钥、私钥对等账户的基本信息。

3、交易上下文：Sui中交易的基本单位。它包括交易相关信息，例如交易ID，账户ID和交易数据。

4、账户逻辑：定义账户如何处理交易和管理其状态的行为和规则的集合。

Sui 的账户抽象模型通过以下步骤处理交易：

1、交易创建：用户创建交易并将其发送到Sui网络。

2、交易验证：Sui网络验证交易的有效性和完整性。

3、Account Lookup：Sui 网络根据交易中的 Account ID 查找对应的账户对象。

4、账户逻辑执行：Sui网络执行账户逻辑来处理交易并更新账户状态。

5、交易确认：Sui网络确认交易结果并记录在区块链上。

简单来说，Sui 的账户抽象模型是一种创新机制，可以简化账户管理和交易处理，使应用程序更加用户友好。

### 2.4 游戏

一条区块链要想脱颖而出，必须打下坚实的基础。我之前说 Move 是一次大胆的尝试，原因有二：

第一，在模块化概念主导的时代，基于 Move 的原生区块链（如 Move 双子星）代表了对 Layer 1 的最后尝试之一，本质上是逆潮流而行。但最近各种异构链的兴起或许证明，模块化并不是唯一的答案。

第二，决定用一种新的编程语言重建区块链，就像试图在今天的移动市场上创建一个新的操作系统来与 iOS 和 Android 竞争——这项努力注定充满挑战。基于 Move 的区块链能否在未来几年像 Solana 一样大放异彩，很大程度上取决于它们选择的发展道路。对 Sui 来说，这一挑战的答案是游戏。

游戏是 Web3 的关键切入点之一，但大多数区块链都不能很好地支持游戏。这是因为区块链主要以金融为设计目标，其去中心化架构本质上性能低下，不适合游戏。然而，Sui 不同。它的模型非常适合 DeFi 应用和非金融应用，包括游戏。如前所述，在 Sui 中，一切都被视为对象。在具有分层复杂资产的游戏或应用程序中，Sui 允许一个对象拥有其他对象（资产可以拥有资产）。例如，在英雄角色游戏中，英雄可能拥有一个包含属于该角色的其他数字资产的库存。Sui 可以以其他区块链无法做到的方式准确地对这些数据层次结构进行建模，使开发人员无需绕过链的基本限制即可构建应用程序。

此外，Sui 还积极与传统 Web2 巨头合作，去年与韩国四大游戏巨头中的三家（Netmarble、NHN、NCSoft）达成合作，今年又与 TikTok 合作开发区块链游戏和 SocialFi 项目，将传统巨头带入 Web2

## 3、Aptos



Aptos 是另一个基于 Move 语言的 Layer 1 区块链，同样专注于构建高性能、可扩展的 Web3 基础设施。它的架构设计与 Sui 有许多相似之处，但也展示了一些独特的功能。

### 3.1 架构

模块化设计：Aptos采用模块化架构，开发者可以独立开发、升级不同模块，提高开发速度和灵活性。

并行执行引擎（Block-STM）：与其他需要预先声明数据依赖关系的区块链不同，Aptos 的并行执行引擎可以并行处理交易，而无需提前知道数据位置，从而提高吞吐量并减少延迟。

流水线交易处理：Aptos 将交易处理分为多个阶段，例如传播、元数据排序和批量存储。这些阶段使用流水线方法并行执行，从而最大限度地提高吞吐量并最大限度地减少延迟。

Move 编程语言：Aptos 使用 Move 编程语言。与 Sui 的创新不同，Aptos 更注重对其进行改进，例如标准化语言、引入更强大的功能支持和定制功能。

灵活的状态同步：这允许节点选择不同的状态同步策略，例如同步完整历史记录或仅同步最新状态，从而增加节点的灵活性。

AptosBFT 共识机制：AptosBFT 是 Aptos 使用的拜占庭容错（BFT）共识机制，通过优化验证者之间的通信和同步来提高吞吐量、降低延迟。相比于 Sui，可以看作是 DiemBFT 的改进版，在效率和崩溃恢复方面有一定的提升，因此这里只会简单提一下。

Aptos 的架构设计使其能够处理大量并发交易，同时保持高速、低费用和安全性。此外，Move 语言和 Aptos 框架为开发人员提供了强大的工具来构建安全、可扩展且用户友好的应用程序。

### 3.2 Block-STM

这里，我们将详细阐述 Aptos 的核心创新，并行执行引擎 Block-STM：

Block-STM的核心原则：

预设顺序执行：Block-STM 依赖于区块内交易的预设顺序，所有交易都必须按照此顺序执行，以保证最终状态的一致性。

乐观并发控制：Block-STM 乐观地并行执行交易，假设不会发生冲突。乐观并发控制基于“冲突很少发生”的假设，允许交易无需锁定即可访问和修改数据。它假定多个交易同时发生冲突的可能性很低，因此可以继续进行修改，并且在最终提交之前检查冲突（如果有）。

多版本数据结构：为了支持乐观并发控制，Block-STM 使用多版本数据结构来存储数据。每次写入操作都会创建一个新的数据版本，而读取操作则会访问相应的数据版本。

验证和重试：执行交易后，Block-STM 会验证读取的数据版本是否仍然有效。如果验证失败，表明存在冲突，则交易被标记为无效并重新执行。

协作调度：Block-STM 使用协作调度程序来协调各个线程的执行和验证任务，以最大限度地实现并行性。

Block-STM的工作流程：

交易分组：一个区块内的交易被分组并分配给不同的线程进行并行执行。

乐观执行：每个线程乐观地执行分配给它的交易，并记录每个交易的读写集。

验证：一旦线程完成交易的执行，它就会验证读取集中的数据版本是否仍然有效。

重试：如果验证失败，表明存在冲突，则交易被标记为无效并重新执行。

提交：一旦所有交易都通过验证，结果将写入区块链状态，完成交易提交。

Block-STM的优点：

高吞吐量：通过利用乐观并发控制和协作调度，Block-STM 可以充分利用多核处理器的性能，实现高吞吐量。

低延迟：由于交易可以并行执行，Block-STM 显著减少了交易确认时间。

安全性：Block-STM预设的顺序执行和验证机制确保了最终状态的一致性和安全性。

简单来说，Block-STM 是一个高效的并行交易执行引擎，它结合了乐观并发控制、多版本数据结构和协作调度技术，以最大化区块链吞吐量，同时确保安全性和正确性。

### 3.3 账户抽象

不同于 Sui 更为直接的账户抽象方式，Aptos 支持的抽象程度较为有限，且缺乏具体的预定义标准。其账户抽象能力主要体现在以下几个方面：

模块化账户管理：使用 Move 模块定义和管理账户，开发人员可以创建自定义模块来实现不同的账户类型和功能。

灵活的密钥管理：允许用户使用不同的密钥对账户进行不同的操作，比如使用一个密钥进行交易签名，使用另一个密钥进行账户管理。

可编程交易验证：开发人员可以在 Move 模块内定义自定义交易验证逻辑，例如多重签名和消费限制，以满足不同的应用场景。

### 3.4 与微软的合作

与更专注于游戏开发的 Sui 不同，Aptos 没有特定的发展目标，而是将自己标榜为最适合生产的区块链。值得一提的是，Aptos 正在与微软合作，旨在将微软的人工智能技术融入区块链。他们的首款合作产品 Aptos Assistant 已经在官方网站上发布，这是一款基于 Aptos 网络构建的生成式人工智能助手。预计未来几个月将发布更多人工智能产品。

## 4、Move生态系统

虽然 Sui 近期表现不俗，但相较于 EVM 链和 Solana、Ton 等异构链，Move 生态的崛起仍需时间磨合。尽管 Sui 和 Aptos 明星效应明显、技术突破不断，但 Move 生态整体规模和活跃度仍落后于成熟生态，开发者数量、应用类型、用户规模均需时间磨合。从对外合作到运营，两个项目都表现出浓重的 Web2 思维，缺少一些 Web3 基因，各类合作项目在行业内也相对冷淡。

然而，考虑到 Move 生态系统的潜力，还有许多领域值得探索。一些开发人员已经注意到了 Move 的未来价值。正如介绍中提到的，已经有项目将 Move 带入 ETH Layer 2 生态系统，Move 生态系统未来很可能会在 ETH Layer 2 领域大放异彩。当前的重点应该是如何让 Move 生态系统成为焦点。