从模块化到聚合：探索Polygon 2.0的Agglayer核心

作者: YBB Capital Researcher Zeke

### TLDR

Agglayer是Polygon 2.0的核心组件，通过聚合和确保原子跨链交易来统一分散的区块链。其目标为提供单链级别的无缝用户体验，解决现有区块链生态系统的流动性和状态分散问题。

Agglayer使用了一种名为悲观证明的新式验证机制，它假设所有接入链都是不安全的，最终利用零知识证明来确保跨链操作的正确性。

Agglayer更为简洁、高效，其最终形态将实现更理想的链抽象，更符合下一代Web3的定义。

### 一、Agglayer从模块化时代中衍生

#### 1.1 Agglayer简介

Agglayer是Polygon 2.0的核心组件之一，其协议名中的Agg为英文单词aggregation（聚合）的缩写，中文全称即为聚合层。该协议的作用本质上与Layerzero、Wormhole等全链互操作性协议并无二致，其目的都是为了将碎片化的区块链世界连接起来。但从构建思路上来说两者确实有一些不同之处，用通俗的话来说，传统的全链互操作性协议更像到处搭桥修建的工程公司，通过给不同的链或协议设计并搭建桥梁（其中异构链的适配较为困难），来达成互联。而Agglayer则恰如其名，形态上更类似于一个交换机构成的“局域网”，连接链只需要通过插入“网线”（ZK proof）就可以接入“局域网”，并进行数据交换。相比于到处跨桥，更快、更易用且有更好的互操作性。

#### 1.2 Shared Validity Sequencing

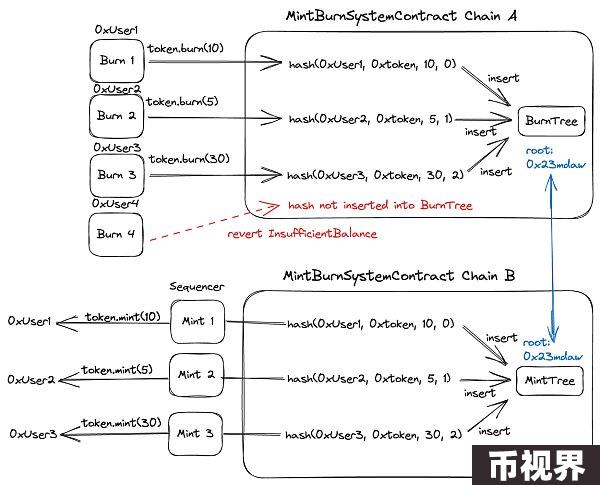
Agglayer的思想很大程度上归功于Umbra Research关于Shared Validity Sequencing（共享有效性排序）的设计，该方案旨在实现多个Optimistic Rollup之间的原子跨链互操作性。通过共享排序器，整个系统能够统一处理多个 Rollup 的交易排序和状态根发布，确保原子性和条件执行。

具体实现逻辑需要通过三个组件：

接受跨链操作的共享排序器：接收和处理跨链交易请求；

区块构建算法：共享排序器负责构建包含跨链操作的区块，确保这些操作的原子性；

共享欺诈证明：在相关 Rollup 之间共享欺诈证明机制，以强制执行跨链操作。



由于当前的Rollup基本已经具备在Layer1和Layer2之间双向传递消息的功能，以及其他特殊的预编译。所以正如上图所示，Umbra在这里仅添加了一个 MintBurnSystemContract合约（Burn与Mint）构成的简洁跨链系统来实现对三个组件的补齐。

工作流程

1.链 A 上的 burn 操作：任何合约或外部账户可以调用，成功后记录到 burnTree；

2.链 B 上的 mint 操作：排序器在执行成功后记录到 mintTree。

不变量和一致性

Merkle 根的一致性：链 A 上的 burnTree 和链 B 上的 mintTree 的 Merkle 根必须相等，如此就可以保证跨链操作的一致性和原子性。

在这种设计下，Rollup A 和 B 共享一个排序器。这个共享排序器负责将两个 Rollup 的交易批次和声明状态根发布到以太坊。而共享排序器可以是一个中心化的排序器，就像目前大部分Layer2的 Rollup 排序器，或者是类似Metis这样的去中心化排序器。整个系统中的关键之处在于，共享排序器必须在同一笔交易中将两个 Rollup 的交易批次和声明状态根发布到 L1。

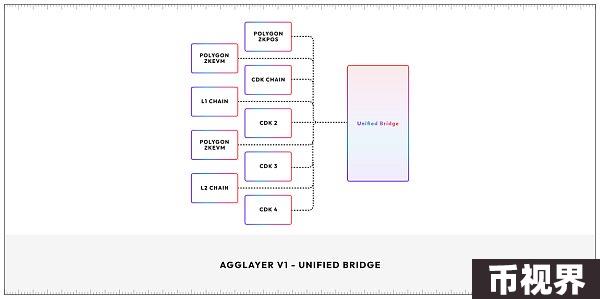
共享排序器接收交易并为 A 和 B 构建区块。对于 A 上的每笔交易，排序器都会执行该交易并检查它是否与 MintBurnSystemContract 交互。如果交易成功执行并与 burn 函数交互，则共享排序器尝试在 B 上执行相应的 mint 交易。如果 mint 交易成功，则共享排序器会在 A 上包含 burn 交易，并在 B 上包含 mint 交易；如果 mint 交易失败，则共享排序器将排除这两笔交易。

简单来说，该系统是对现有区块构建算法的简单扩展。排序器执行交易并将条件触发的交易从一个 Rollup 插入到另一个 Rollup，而在主链进行欺诈证明验证时也仅需保证链 A的燃烧与链B的铸造是正确的（也就是上文中的Merkle 根的一致性）。这种情况下多条Rollups就变得类似于一条链，与单片Rollup相比，这种设计提供了更好的分片支持、应用主权、互操作性。但相反的问题是，节点验证与排序器的负担更大，且从利益分配和Rollups自主性等多种角度来说这种方案被采用的概率还是很低。

#### 1.3 Agglayer的核心组件

Agglayer在吸纳上述方案的同时，进行了更多高效的改进，并引入了两个关键组件：统一桥与悲观证明。

统一桥：统一桥的工作流程为收集并汇总所有接入链的状态至聚合层，聚合层再生成统一的证明至以太坊，在这个过程中存在三个阶段的状态：预确认（预确认允许在临时状态假设下更快的交互）、确认（确认验证提交的证明的有效性）和最终确定，最后该证明可验证所有接入链的交易有效性。



悲观证明：Rollups接入一个多链环境将会产生两个主要问题：1.不同验证者与共识机制的引入，将导致安全性复杂；2.Optimistic Rollup的收款需要7天。为了解决这两个问题Polygon引入了一种新颖的零知识证明方式，即悲观证明。

悲观证明的思路是，假设所有连接到 AggLayer 的区块链都可能存在恶意行为，并对所有跨链操作进行最坏情况的假设。 然后，AggLayer 会利用零知识证明来验证这些操作的正确性，确保即使存在恶意行为，也无法破坏跨链操作的完整性。

#### 1.4 特性

在该方案下，可实现下述多种特性：

原生代币。通过使用统一桥，聚合层内的资产都为原生资产，没有任何包装代币，也无需第三方的信任源进行跨链，一切都是无缝的；

统一流动性。所有接入链的TVL都是共享的，也可称为共享流动性池；

主权。与上文中的Optimistic Rollup通过共享排序器获得互操作性的方式相比，Agglayer具有更好的主权，AggLayer 将与共享排序器和第三方 DA 解决方案兼容。连接链甚至可以使用其原生代币作为Gas；

更快。依旧是不同于上文Optimistic Rollup的方案，Agglayer进行跨链无需等待7天；

安全。悲观证明只接受正确的行为，另一方面它还确保了没有一条链可以提取超过存入的金额，从而保证聚合层共享资产池的安全；

低成本。在聚合层中接入的链越多，缴纳给以太坊的证明费用就越低，因为这是均摊的，Agglayer不收取额外的协议费用。

### 二、跨链方案

#### 2.1 为何跨链那么难？

在上文中曾提到过，Agglayer与全链协议的目的基本一致，那么两者究竟孰优孰劣？在比对之前，我们可能需要先了解两个问题：1.为何跨链难，2.常见的跨链方案都有哪些。

如同最著名的公链三角难题一样，跨链协议也存在互操作性三角选择（interoperability trilemma），由于去中心化这个大前提的限制，区块链本质上是一台无法接收外部信息的复制状态机。虽然AMM和预言机的存在弥补了DeFi缺失的拼图，但对于跨链协议而言，这个问题要复杂数十倍，从某种角度来说我们甚至永远无法从原链中取出任何真正的代币，所以也就有了各种xxBTC与xxETH等包装代币。但这种包装代币方案的逻辑非常危险和中心化，因为你需要把真的BTC和ETH锁在跨链桥合约的原链地址中，而整个跨链的设计中还可能需要面临资产互不相同、虚拟机不同导致的协议不兼容、信任问题、双花问题、延迟问题等多种问题。为了高效和降低开支，大部分跨链方案其实采用的还是多签钱包的方案。所以即便时至今日，你也经常能看见xx跨链桥暴雷的信息。现在我们再以更低层的角度来细致地了解这个问题，从Connext 创始人 Arjun Bhuptani的归纳来看跨链协议只能在以下三个关键属性中选择两个来优化：

无信任性（Trustlessness）：不需要依赖于任何中心化的信任实体，能够提供与底层区块链相同水平的安全性。用户和参与者不需要信任任何中介或第三方就能确保交易的安全和正确执行；

可扩展性（Extensibility）：协议能够轻松地适用于任何区块链平台或网络，不受特定技术架构或规则的限制。这允许互操作性解决方案能够支持广泛的区块链生态系统，而不仅仅是几个特定的网络；

泛用性（Generalizability）：协议能够处理任何类型的跨域数据或资产转移，而不仅限于特定的交易类型或资产。这意味着通过该桥梁，不同的区块链可以交换各种类型的信息和值，包括但不限于加密货币、智能合约调用，和其他任意数据。

早期的跨链桥划分一般是以Vitalik等人为准，他们把跨链技术分成了三类，哈希时间锁、见证人验证、中继验证（轻客户端验证），但后来根据 Arjun Bhuptani的划分，跨链方案又可分为原生验证（无信任性+ 可扩展性）、外部验证（可扩展性 + 泛用性）、原生验证 （无信任性+ 泛用性）。这些验证方式基于不同的信任模型和技术实现，以满足不同的安全性和互操作性需求。

本地验证（Natively Verified）：

本地验证的桥梁依靠源链和目标链本身的共识机制来直接验证交易的有效性。这种方式不需要额外的验证层或中介。例如，一些桥梁可能利用智能合约在两个区块链之间直接创建验证逻辑，允许这两个链通过它们自己的共识机制来确认交易。这种方法的优点是增加了安全性，因为它直接依赖于参与链的固有安全机制。然而，这种方法在技术实现上可能更为复杂，并且不是所有的区块链都支持直接的本地验证。

外部验证（Externally Verified）：

外部验证的桥梁使用第三方验证者或验证者集群来确认交易的有效性。这些验证者可能是独立的节点、联盟成员或其他某种形式的参与者，它们在源链和目标链之外运行。这种方式通常涉及到跨链消息传递和验证逻辑，这些逻辑由外部实体执行，而不是由参与的区块链本身直接处理。外部验证允许更广泛的互操作性和灵活性，因为它不受特定链的限制，但同时也引入了额外的信任层和潜在的安全风险。（虽然有极大的中心化风险，但外部验证是最主流的跨链方式，除了灵活高效还兼具费用低廉的特性）

原生验证（Locally Verified）：

原生验证指在跨链交互中目标链验证源链的状态，以确认交易并在本地执行后续交易。通常的做法是在目标链虚拟机的源链上运行轻客户端，或二者并行。原生验证需要诚实的少数派或同步假设，委员会里至少有一个诚实的中继者（即诚实的少数派），或如果委员会无法正常运行，用户必须自己传输交易（即同步假设）。原生验证是信任最小化程度最高的一种跨链通信方式，但是它成本也很高，开发灵活性较低，而且更适合状态机相似度较高的区块链，比如以太坊和L2网络之间，或者基于Cosmos SDK开发的区块链之间。

当前的跨链方案「1」

在不同方面的妥协也就导致不同类型的跨链方案层出不穷，除了验证方式外。当前的跨链方案也可分为多类，它们各自采取独特的方法来实现资产的交换、转移和合约调用。

Token交换：允许用户在一个区块链上交易某种资产，并在另一个链上接收等价的另一种资产。通过利用原子互换和跨链做市商（AMM）等技术，可以在不同链上创建流动性池，从而实现不同资产间的兑换。

资产桥：这种方法涉及到在源链上通过智能合约锁定或销毁资产，并在目标链上通过相应的智能合约解锁或创建新的资产。这种技术根据处理资产的方式可以进一步分为三种类型：

锁定/铸造模式：在这种模式下，源链上的资产被锁定，而目标链上则铸造出等价的“桥接资产”，反向操作时则销毁目标链上的桥接资产以解锁源链上的原资产；

销毁/铸造模式：此模式下源链上的资产被销毁，目标链上则铸造出等量的相同资产；

锁定/解锁模式：这种方式涉及到在源链锁定资产，然后在目标链上的流动性池中解锁等价的资产。此类资产桥往往通过提供收入分享等激励措施来吸引流动性。

原生支付：允许源链上的应用触发目标链上使用原生资产的支付操作，也可以基于一条链上的数据在另一条链上触发跨链支付。这种方式主要用于结算，可以根据区块链数据或外部事件进行。

智能合约互操作：允许源链上的智能合约根据本地数据调用目标链上的智能合约函数，实现复杂的跨链应用，包括资产交换和桥接操作。

可编程桥：这是一种高级的互操作性解决方案，结合了资产桥接和消息传输功能。当资产从源链转移到目标链时，可以立即触发目标链上的合约调用，实现多种跨链功能，例如权益质押、资产交换，或将资产存储在目标链上的智能合约中。

#### 2.2 Agglayer在未来更具优势

我们在此处将Agglayer对比当前的全链协议，以全链协议最具影响力的LayerZero为例。该协议采用了外部验证的改良版，即LayerZero将验证的信任源转化为两个彼此独立的实体——预言机和中继器，通过最极简的方式来弥补外部验证的缺陷。跨链方案上属于可以实现多种操作的可编程桥方案。从逻辑上来说似乎简洁利落的破解了，所谓的不可能三角。从宏大的叙事角度来说，LayerZero有机会成为整个Web3的跨链枢纽，并且相当契合模块化时代下链爆炸所产生的体验割裂、流动性破碎等问题，这也就是为什么头部VC要在这类协议上进行疯狂押注的主因。

但真实情况又是如何呢？我们暂且不谈近期Layerzero关于空投的各种骚操作。只从发展的角度考虑，此类协议想达到理想中贯连整个Web3的情况其实非常困难，且去中心化问题存疑。在早期的V1版本中，LayerZero采用的预言机其实存在被黑以及理论上存在预言机作恶的可能（关于这点，Wormhole采用业内机构作为守护者节点，也时常被人所诟病），直到V2版本的去中心化验证网络（DVN）诞生才平息了社交网络上的口诛笔伐，但这同样是基于大量的B端资源。

另一方面，全链协议的开发还涉及异构链的协议、数据格式和操作逻辑，以及不同智能合约的调用问题。想要真正实现Web3的互通不仅需要自身努力，可能还需要各类项目的协同。如果各位使用过早期的LayerZero应该不难发现它基本仅支持EVM系公链的跨链，且支持全链的生态项目并不多。这点对于Agglayer也是同理，但在互操作性上，Agglayer支持超低的延迟及异步互操作性，这要比全链协议更像我们日常使用的互联网。

综合来说，Agglayer聚合为类似单链使用的方式，整体上更简洁、高效且符合当下的模块化风向。不过两者之间在当下，并不存在绝对的高低，全链协议依旧具备最广阔的流动性、生态与更强的主动性，且发展较为成熟的优势。而Agglayer的优势在于将互为敌对的Layer1、Layer2真正意义上的聚合，打破链爆炸时代不同公链项目，分散流动性与用户的零和游戏，允许多链低延迟交互，并原生自带链抽象，共享流动性池不需要包装代币，这对于长尾链和应用链来说将是非常好的机会。所以从长远来看Agglayer是当前最具潜力的跨链方案，目前同样在开发阶段的类似项目还有Polkadot 的“Join-Accumulate Machine”，未来必定还有更多的类似方案出现，Web3的历史在如今已从单片走向模块，而下一步将走向聚合式。

### 三、Agglayer所连接的生态

由于尚在早期，Agglayer的接入链还并不多，这里主要提及三个项目：

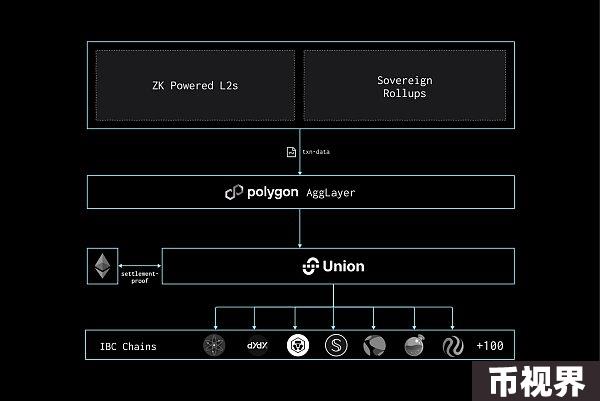
#### 3.1 X Layer

X Layer是基于Polygon CDK建立的以太坊Layer2项目，它连接了欧易和以太坊社区，让任何人都能参与到真正全球化的链上生态系统中。作为头部交易所的公链，在接入Agglayer后将为聚合层内的项目带来广泛的流动性。而OKX Web3钱包作为普通用户的访问层，也许也会对Agglayer提供更好的支持。



#### 3.2 Union

Union是一个基于Cosmos构建的零知识基础设施层，该项目用于一般消息传递、资产转移、NFT 和 DeFi。它基于共识验证，不依赖于受信任的第三方、预言机、多重签名或 MPC。作为接入链，进入聚合层后实现了EVM与Cosmos的深度连接，因为只需将Union作为IBC网关即可实现连接Union再连接IBC，从而使得两个彼此割裂的模块化生态重新组合在一起。



#### 3.3 Astar

Astar Network是日本及全球企业、娱乐和游戏项目的网络，致力于推动 “Web3“。它利用由 Polygon 和 Polkadot 支持的跨虚拟机，提供可定制的区块链解决方案。该项目作为Agglayer的首个完全集成链，将直接接入数百亿美元的流动性共享池，并实现真正的用户增长。



参考文献

1.一文读懂区块链互操作性：https://blog.chain.link/blockchain-interoperability-zh/

2.AggLayer: Why Polygon’s Scalability Solution is a Game Changer in 2024 & Beyond?：

https://www.antiersolutions.com/agglayer-why-polygons-scalability-solution-is-a-game-changer-in-2024-beyond/

3.The Aggregation Age is Coming：https://polygon.technology/agglayer

4.Shared Validity Sequencing：https://www.umbraresearch.xyz/writings/shared-validity-sequencing

5.Union：https://www.rootdata.com/zh/Projects/detail/Union?k=MTAxMjY%3D