2020年以来，以太坊的扩张路线图一直围绕&quot汇总&quot:使用证明(无论是零知识证明还是最优欺诈证明)继承以太坊安全的独立执行环境。。

经过多年的开发，Rollup终于完成了部署，正在被采用。仲裁法庭&#039；美国的王牌产品OptimalRollup已经上线近一年，在此期间，价值超过27亿美元的资产被存放在立交桥上。，其次是最优性。Loopring和dydX等特定应用的零知识总结也在广泛使用，未来几个月还将推出许多有竞争力的通用零知识总结。

虽然现在Rollup很快，但还是有人担心它的成本高。

事实上仲裁和优化的交易成本仍然明显高于&quot低成本&quot如索拉纳和多边形链。

那么，是什么阻碍了这些汇总的发展呢？

为了理解交易成本，我们首先需要区分区块链交易产生的各种成本：

？执行

这是网络中所有节点执行事务并验证结果是否有效的成本(例如，您实际拥有您所转移的令牌的所有权)。

？存储/状态

这是更新区块链的成本&quot数据库&quot用新值(例如，令牌转移后，发送方余额减少，接收方余额增加)。

？数据可用性

为了使区块链不被任何人信任和验证，区块链必须确保交易的所有相关数据与所有网络参与者公开共享。本质上，这是为了保证世界上的每个人都能看到你的交易。没有这种保证，各种攻击都可能发生(称为阻塞攻击)。

正如我们所看到的，数据可用性是当今区块链的主要瓶颈之一。

rollup:

rollup的主要进步是它将区块链的执行和存储移到&quot链外&quot。也就是说，

在有限的一组节点上。我们可以直接将这个任务委托给Rollup操作者，而不是让网络中的每个以太坊节点执行所有事务或存储每个更新。

然而这是否意味着我们需要信任这些运营商？不是&#039；这不是集权吗？

Rollup使用各种证明类型来继承以太坊的安全性。。最佳汇总允许单个诚实实体提交一个&quot欺诈证书&quot并为行为不端的定序器赢得奖励，而ZKRollup使用零知识证书来证明第二层链已经被正确更新。

数据可用性的权衡

从主链转移可以大大降低执行和状态存储的成本，但是Rollup仍然需要将他们的数据发布到第一层链，以保证数据的可用性。生性Rollup支付第二层的低执行和存储成本，但仍然需要支付第一层的费用来发布他们的数据。

这可以在&quot高级事务信息&quotArbiScan块浏览器中任何事务的选项卡。交易成本包括发布到L1的呼叫数据成本、在L2和L2存储上使用的计算，以及在几乎所有交易中，，L1通话数据是费用的主要来源。换句话说，Rollup要解决的最重要的问题是将数据发布到第1层的成本。

数据可用性的未来

尽管数据可用性仍然是汇总的瓶颈，但随着时间的推移，这种情况将会得到缓解。

以太坊的升级，如Proto-Danksharding和最终完成的Danksharding，将大大降低将数据发布到以太坊的成本。此外，像Celestia这样的项目旨在提供独立的链。这些链是专门为提供廉价的数据可用性而构建的。

从长远来看，Danksharding和Celestia等系统将降低数据可用性的成本，增加其丰富性，同时将问题扔回执行层面。然而这些解决方案需要时间才能完全成熟：Celestia在几个月内不会发布其主网络，并且在以太坊可以添加像Proto-Danksharding这样的数据可用性升级之前。可能要一年多。数据压缩是一个比计算机本身更古老的领域。莫尔斯电码发明于1838年，这是已知的最早的数据压缩的例子。然后，计算机的使用加速了人们&#039；数据压缩的研究所以像霍夫曼编码这样的算法在20世纪50年代被发明出来。

鉴于Rollup的实施成本低，但数据可用性成本高，这些团队一直在将数据压缩算法集成到他们的协议中。。optimization已经将Zlib压缩算法集成到他们的Rollup中，而Arbitrum即将推出的Nitro升级则使用brotli压缩算法。

注：这个实验可能是在Nitro发布之前匆忙完成的，以便在未压缩的Arbitrum调用数据上进行实验。

数据压缩算法绝对是一个有用的工具，有助于降低这些通话数据的成本。。然而，压缩区块链事务是一项艰巨的任务：数据压缩的作用是找到相同的模式并缩短它们。然而，事务充满了地址、散列值和签名。对于这些压缩算法，它们本质上是&quot随机数据&quot没有相似之处。

只有当开发者开始关心如何减少他们应用中的通话数据时，这种数据的成本才能真正降低。2020-2021年天然气的天价迫使开发人员优化他们的代码，以最小化执行和状态存储。

当我们过渡到 L2 世界时调用数据会从最便宜的资源变成最贵的资源，开发者必须重新学习这些新的优化方案。

现在让&#039；让我们在Arbitrum上做一个实验：我们可以将简单令牌传输所需的呼叫数据压缩到什么程度？这些优化能在多大程度上降低交易成本？实验设计和控制组事务为了执行我们的实验，我们将建立一个简单的智能契约，将令牌从交易发送方传输到任何给定的地址。

这个智能合约确实需要用户之前发送我们的实际测试交易。，首先发送一个approve()事务。由于这一限制，用户可能不想使用该系统进行令牌传输。然而，本实验中使用的节约成本的方法也可以应用于其他合同(例如，优化的Uniswap路由器)。

在实验开始时，我们将发送一个&quot控制&quot事务来获得基准成本，这将调用一个简单的Solidity函数来传递令牌地址、接收者地址和要转移的令牌数量。

我们的测试交易使用了576，051个ArbiGas，总成本为0.43美元。

数据删减

在控制组中使用的呼叫数据中有许多不必要的数据，我们可以将其剔除。首先，我们需要删除所有的零，这些零只用于数据填充。虽然它们的非零字节比较便宜，但是还是有成本的，所以需要删除。

在

的开头还有一个4字节的函数签名，是我们试图调用哪个Solidity函数的标识符。我们可以删除这些数据，让我们的代码推断我们想要采取的行动。

经过这两步优化后，我们将字节码从100个减少到了43个。这样，我们的测试交易使用了494，485个阿比加斯(减少了14%)，花费了0.37美元。

「助手」合约

目前我们大部分的数据都是由调用数据中的两个地址组成：一个是我们要转移的令牌地址，一个是转移接收地址。

然而我们可以假设大多数用户都在传输相同种类的令牌(WETH、戴、)。因此，从调用数据中删除整个令牌地址的一种方法是部署特殊的&quot助手&quot令牌合同。。如果我们可以将事务发送给这个助手，我们将完全避免发送令牌地址的需要。

这样，我们将数据字节码减少到了23个字节。，测试交易使用了457，546个阿比加斯(比对照组少21%)，成本为0.34美元。

地址查询表

在最后一个阶段，我们用&quot助理合同&quot，但我们的通话数据仍包含另一个地址。我们能找到另一种更可靠的&quot压缩&gt；地址？

还好，Arbitrum有一个名为地址表注册的内置合同，我们可以使用它来缩短我们的呼叫数据。本合同实质上是一份&quot电话簿&quot，它可以将一个20字节的以太坊地址转换成一个简单的整数。。想象一下，你的朋友有一本传统的电话簿：不用把你的整个电话号码读给他们听，只要说&quot我是电话簿第200页的第四个电话号码&quot让他们查你的号码。

因此，，我们可以签订合同并使用&quot地址索引&quot替换完整的地址，并在内部查找。

这样，我们省略了令牌地址和接收地址，从而将调用数据减少到9个字节。因此，我们的测试事务使用了428，347个ArbiGas(比控制组少26%)，成本为0.32美元。

方法合并

最后让&#039；让我们将所有方法整合在一起：

？是否删除数据填充和函数选择器

？使用辅助合同删除公共地址

？使用Arbitrum地址表缩短其他地址

总之，我们的呼叫数据大小现在只有6个字节。最终的测试交易使用了426，529阿比加斯(也比对照组减少了26%，略低于之前的测试组)，花费了0.32美元。

有损压缩

我们刚才谈到的所有压缩方法都属于&quot无损压缩&quot也就是说，压缩输出包含与原始输入相同的所有数据。

但是就像照片和视频文件一样，它们通常使用&quot有损压缩&quot删除不必要信息的算法。我们也可以在大多数情况下删除不必要的数据。

我们可以通过缩短数字来去掉不必要的精度。例如，ERC-20令牌通常具有18位小数的精度，但大多数用户通常只关心4位小数。。为此，我们可以建立一个契约，默认接受小数点后8位并乘以10的10次方，为需要更多精度的用户提供相应的辅助功能。类似地日期通常表示为&quot自1970年1月1日以来的秒数&quot(也称为Unix时间)。契约可以通过设置不同的时间单位(如分钟、小时或天)来减小该整数的大小，并且可以设置自己的&quot纪元&quot，比如，2015年1月1日。

简而言之，通话数据已经从以太坊L1上最便宜的资源变成了以太坊Rollup上最贵的资源。。Proto-Danksharding和Celestia等数据可用性技术最终会解决这个问题，但它们都还没有上线，数据可用性变得廉价和普遍可能需要几年时间。

因此，区块链开发者需要非常注意他们的交易所需的调用数据量，因为这将对最终用户的交易成本产生重大影响。

本文概述了一些可用于减少呼叫数据的技术方法。我相信随着越来越多的&quot优化军队&quot把注意力转向第二层，这样的方法会越来越丰富。

来源：街区节奏