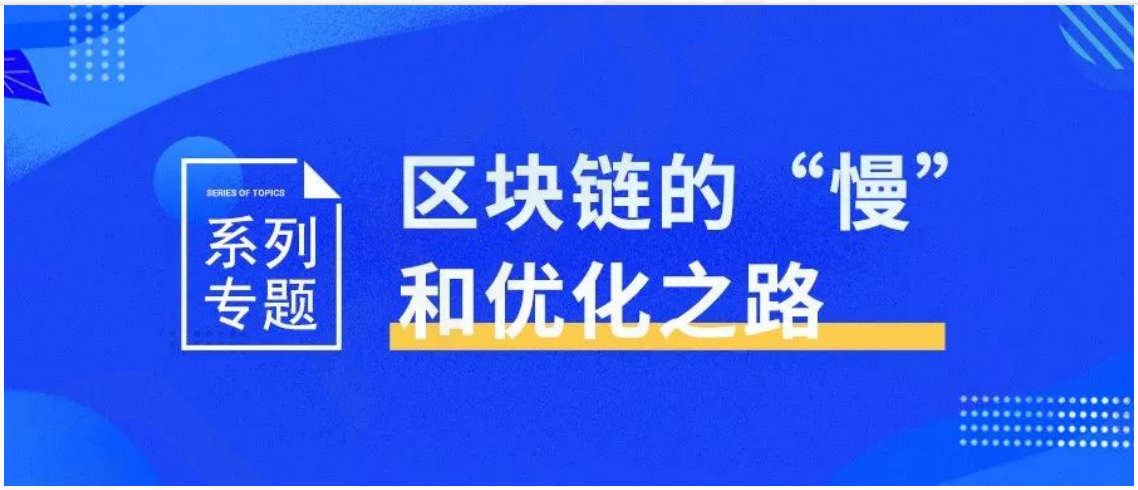
区块链的速度困境：“贵”在信任，“慢”得其所

原创 张开翔 FISCO BCOS开源社区 2019-04-16

区块链领域最受关注的一个方面是“性能”，或者说“TPS”，比起来有种“不服就跑个分”的感觉。跑分项包括TPS（每秒处理交易数）、并发能力（同时承担交易量）、交易响应时间等。然而，相比每秒能发送200万封电子邮件、支持数百万用户同时登录一个社交平台的互联网服务来说，区块链的速度简直是太！慢！了！ 甚至有人调侃说“区块链，不就是最慢的分布式数据库吗”（这句话可以展开多方面解析，本篇先讨论慢的问题）

区块链技术前景无限美好，可如果没有高性能表现作为支撑，无法运行快速的、执行复杂的智能合约逻辑，快速的完成交易事务，那些令人振奋的前景就只能是摘不到的镜中花，捞不着的水中月。

大热的区块链技术为什么这么慢？有什么方案能解开区块链性能的镣铐，让区块链轻盈飞入各行各业？我们将通过这一系列专题，与你一道尝试扩宽区块链的优化之路👇



1 为什么区块链很慢？（直面区块链的速度困境，理解区块链的“慢得其所”）
2 FISCO BCOS的性能优化（我们是如何将区块链交易处理性能提升的）
3 基于DAG的交易并行执行引擎（业界公认的并行化方案，FISCO BCOS真的具体落地了！）
4 共识与同步流程优化（抓主要矛盾，从最慢的地方下手，让共识行驶上快车道！）
5 全方位的并行处理（矛盾是相互转化的，撸完共识，还得让该并行的都并行，不让CPU有一丝丝懈怠）
6 全面的性能分析工具（工欲善其事必先利其器，工具不够自己造）
7 并行合约开发框架（是时候揭晓庐山真面目了，一起畅享高铁般的舒适与快速）

系列专题 | 区块链的 “慢”和优化之路（1）

为什么区块链很慢?

作者：张开翔

张开翔

FISCO BCOS 首席架构师

联盟链老司机

AUTHOR | 作者

#举个例子#

数钱，比如数一个亿（是不是好刺激~）

1、如果一个人数，慢，但好在专注，全力以赴，在可见的时间内可以数完。这叫单线程密集计算。

2、如果N个人一起数，每人平分，分头同时数，最后汇总总数，所用时间基本上是第一种情况的1/N，参与的人越多，所需时间就越少，TPS就越高。这叫并行计算和MapReduce。

3、如果N个人一起数，但由于这N个人互相不信任，得彼此盯着，首先抽签选一个人，这个人捡出一叠钱（比如一万块一叠）数一遍，打上封条，签名盖章，然后给另外几个人一起同时重新数一遍，数好的人都签名盖章，这叠钱才算点好了。然后再抽签换个人检出下一叠来数，如此循环。因为一个人数钱时别人只是盯着，而且一个人数完且打上封条和签名的一叠钱，其他人要重复数一遍再签名确认，那么可想而知，这种方式肯定是最慢的。**这就叫区块链。**

但换个角度，方式1，一个人数有可能会数错，这个人有可能生病或休假，导致没有人干活，更坏的结果是，这个人可能调换假币或者私藏一部分钱，报一个错的总数。

方式2，N个人中会有一定比例数错，也可能其中一个人休假或者怠工，导致最终结果出不来，更可能因为人多手杂，出现部分人偷钱、换假钱、报假数……

方式3，很慢，但是很安全，因为所有人都会盯着全过程进行验算，所以肯定不会数错。如果其中有人掉线，可以换人捡出新的一叠钱继续数，工作不会中断。所有数过的钱上面都有封条和签名，不会被做手脚，万一出错了也可以找到责任人进行追责。这种情况下，资金安全是完全得到保障的，除非所有的参与者都串通一气了。该模式下，参与的人越多，资金安全性就越高。

所以，区块链方案致力追求的是，在缺乏互相信任的分布式网络环境下，实现交易的安全性、公允性，达成数据的高度一致性，防篡改、防作恶、可追溯，付出的代价之一就是性能。

最著名的比特币网络，平均每秒只能处理5~7笔交易，10分钟出1个块，达到交易的最终确定性需要6个块也就是1个小时，且出块过程相当损耗算力（POW挖矿）。

号称“全球计算机”的以太坊，每秒能处理的交易数也仅是2位数的量级，十几秒出1个块。以太坊目前也是采用损耗算力的共识机制POW挖矿，会逐步迁移到POS共识机制。

这两个网络在粉丝们爆炸性地进行交易时，可能会陷入拥堵状态，大量的交易发出后，一两天甚至更长的时间才会被打包确认。

但在资金安全就是命的场景下，有些事情是“必须”的，所以，即使慢，还是会考虑选择区块链。

#区块链为什么慢#

分布式系统里有一个著名的理论叫CAP理论：2000年，Eric Brewer教授提出一个猜想：一致性、可用性和分区容错性三者，无法在分布式系统中被同时满足，并且最多只能满足其中两个。

CAP的大致解释
Consistency(一致性)：数据一致更新，所有数据变动都是同步的
Availability(可用性)：好的响应性能
Partition tolerance(分区容错性)： 可靠性

这个理论虽然有一些争议，但从工程实践中看，和光速理论一样，可以无限逼近极致但是难以突破。

区块链系统能把一致性和可靠性做到极致，但是“好的响应性能”方面一直有点被人诟病。

我们面向的“联盟链”领域，因为在准入标准，系统架构、参与节点数、共识机制等方面都和公链不同，其性能表现远高于公有链，但是目前几个主流的区块链平台，在常规PC级服务器硬件上实测，TPS一般是在千级的样子，交易延迟一般在1秒到10秒这个级别。（听说TPS十几万级和百万级千万级区块链已经做出来了？好吧，期待）

笔者曾在大型互联网公司工作多年，在海量服务领域，面对C10K问题（concurrent 10000 connection，万级并发）已经有轻车熟路的解决方案，对一般的电商业务或内容浏览服务，普通pc级服务器单机达到几万TPS，且平均延时在500毫秒以内，飞一般的体验已经是常态，毕竟互联网产品卡一下说不定就会导致用户流失。对于快速增长的互联网项目，通过平行扩容、弹性扩容、立体扩容的方式，几乎能无底线、无上限地面对山呼海啸的海量流量。

相比而言，区块链的性能比互联网服务慢，而且难以扩容，根因还是在其“用计算换信任”的设计思路上。

具体哪里慢呢?

从“古典”区块链的系统内部来看👇

1、为了安全防篡改防泄密可追溯，引入了加密算法来处理交易数据，增加了CPU计算开销，包括HASH、对称加密、椭圆曲线或RSA等算法的非对称加密、数据签名和验签、CA证书校验，甚至是目前还慢到令人发指的同态加密、零知识证明等。在数据格式上，区块链的数据结构本身包含了各种签名、HASH等交易外的校验性数据，数据打包解包、传输、校验等处理起来较为繁琐。

对比互联网服务，也会有数据加密和协议打包解包的步骤，但是越精简越好，优化到了极致，如无必要，绝不增加累赘的计算负担。

2、为了保证交易事务性，交易是串行进行的，而且是彻底的串行，先对交易排序，然后用单线程执行智能合约，以避免乱序执行导致的事务混乱、数据冲突等。即使在一个服务器有多核的CPU，操作系统支持多线程多进程，以及网络中有多个节点、多台服务器的前提下，所有交易也是有条不紊地、严格地按单线程在每台计算机上单核地进行运算，这个时候多核CPU其他的核可能完全是空闲的。

而互联网服务则是能用多少服务器的多少个核，采用全异步处理、多进程、多线程、协程、缓存、优化IOWAIT等等，一定会把硬件计算能力跑满。

3、为了保证网络的整体可用性，区块链采用了P2P网络架构以及类似Gossip的传输模式，所有的区块和交易数据，都会无差别地向网络广播，接收到的节点继续接力传播，这种模式可以使数据尽可能地传达给网络中的所有人，即使这些人在不同的区域或子网里。代价是传输冗余度高，会占用较多的带宽，且传播的到达时间不确定，可能很快，也可能很慢（中转次数很多）。

对比互联网服务，除非出错重传，否则网络传输一定是最精简的，用有限的带宽来承载海量的数据，且传输路径会争取最优，点对点传输。

4、为了支持智能合约特性，类似以太坊等区块链解决方案，为了实现沙盒特性，保证运行环境的安全和屏蔽不一致性因素，其智能合约引擎要么是解释型的EVM，或者是采用docker封装的计算单元，智能合约核心引擎的启动速度，指令执行速度，都没有达到最高水平，消耗的内存资源也没有达到最优。

而用常规计算机语言如C++、JAVA、go、rust语言直接实现海量互联网服务，在这方面常常没有限制。

5、为了达到可容易校验防篡改的效果，除了第一条提到的，区块数据结构里携带数据较多之外，针对交易输入和输出，会采用类似merkle树、帕特里夏（Patricia ）树等复杂的树状结构，通过层层计算得到数据证明，供后续流程快速校验。树的细节这里不展开，可以通过网络上的资料来学习其机制。

基本上，生成和维护这种树的过程是非常非常非常非常繁琐的，既占用CPU的计算量，又占用存储量，使用了树后，整体有效数据承载量（即客户端发起的交易数据和实际存储下来的最终数据对比）急剧下降到百分之几，极端情况下，可能接受了10m的交易数据后，在区块链磁盘上可能实际需要几百兆的数据维护开销），因为存储量的几何级数增加，对IO性能要求也会更高。

互联网服务因为基本不考虑分布式互验互信的问题，很少有使用这种树的证明结构，了不起算下MD5和HASH做为协议校验位。

6、为了达到全网一致性和公信力，在区块链中所有的区块和交易数据，都会通过共识机制框架驱动，在网络上广播出去，由所有的节点运行多步复杂的验算和表决，大多数节点认可的数据，才会落地确认。

在网络上增加新的节点，并不会增加系统容量和提升处理速度，这一点彻底颠覆了“性能不足硬件补”的常规互联网系统思维，其根因是区块链中所有节点都在做重复的验算以及生成自己的数据存

储，并不复用其他节点数据，且节点计算能力参差不齐，甚至会使最终确认的速度变慢。

在区块链系统中增加节点，只会增加可容错性和网络的公信力，而不会增强性能表现，使得在同一个链中，平行扩展的可能性基本缺失了。

而互联网服务大多是无状态的，数据可缓存可复用，请求和返回之间的步骤相对简单，容易进行平行扩展，可以快速调度更多的资源参与服务，拥有无限的弹性。

7、因为区块数据结构和共识机制特性，导致交易到了区块链之后，会先排序，然后加入到区块里，以区块为单位，一小批一小批数据的进行共识确认，而不是收到一个交易立刻进行共识确认，比如：每个区块包含1000个交易，每3秒共识确认一次，这个时候交易有可能需要1~3秒的时间才能被确认。

更坏的情况是，交易一直在排队，而没有被打包进区块（因为队列拥堵），导致确认时延更长。这种交易时延一般远大于互联网服务500ms响应的标准。所以区块链其实并不适合直接用于追求快速响应的实时交易场景，行业通常说的“提高交易效率”是把最终清结算的时间都算在内的，比如把T+1长达一两天的对账或清结算时延，缩短到几十秒或几分钟，成为一个“准实时”的体验。

综上所述，区块链系统天生就背着几座大山，包括单机内部计算开销和存储较大，背着串行计算的原罪，网络结构复杂冗余度高，区块打包共识的节奏导致时延较长，而在可扩展性上又难以直接增加硬件来平行扩容，导致scale up和scale out两方面，都存在明显瓶颈。

Scale Out（等同scale horizontally）：横向扩展，向外扩展，如：向原有系统添加一组独立的新机器，用更多的机器来增加服务容量

Scale Up（等同Scale vertically）：纵向扩展，向上扩展，如向原有的机器添加CPU、内存，在机器内部增加处理能力

直面区块链的速度困境，FISCO BCOS的开发者发挥“愚公移山”的精神，努力优化。经过一段时间的努力，已经移山倒海，修出了一条又一条高速通道，使区块链找到了迈向极速时代的路子（详见下篇），这就是我们系列文章要深入解析的内容。

FISCO BCOS的代码完全开源且免费

下载地址↓↓↓

https://github.com/fisco-bcos