 

**课程报告**

|  |  |
| --- | --- |
| **课程名称：** | **企业软件项目实训** |
| **学生姓名：** | **石望华** |
| **学生学号：** | **201630676843** |
| **学生专业：** | **软件工程** |
| **开课学期：** | **2018-2019第二学期** |

**软件学院**

**2019年7月**

**目 录**

[1. 区块链技术原理](#_Toc390_WPSOffice_Level1)[...................................................2](#_Toc390_WPSOffice_Level1)

[2. 联盟链和公有链的异同](#_Toc2309_WPSOffice_Level1)[.............................................6](#_Toc2309_WPSOffice_Level1)

[3. 信任链是如何建立的？](#_Toc21785_WPSOffice_Level1)[.............................................7](#_Toc21785_WPSOffice_Level1)

[4. 链式存储和MPT存储](#_Toc18342_WPSOffice_Level1)[...............................................7](#_Toc18342_WPSOffice_Level1)

[5. Gas在智能合约中的作用](#_Toc29776_WPSOffice_Level1)[...........................................10](#_Toc29776_WPSOffice_Level1)

[6. EVM中的数据存储结构](#_Toc1873_WPSOffice_Level1)[.............................................10](#_Toc1873_WPSOffice_Level1)

[7. 群组架构的好处](#_Toc20869_WPSOffice_Level1)[..................................................11](#_Toc20869_WPSOffice_Level1)

8. 分布式存储有什么优势？..........................................12

9.并行计算.........................................................14

10.当前区块链实施的难度............................................15

11.参考文献........................................................15

1. **区块链技术原理**

以太坊区块链技术是当今互联网时代的一项具有创新性、革命性的综合型分布式数据库账本技术，具有去中介化、自治性、数据不可篡改、可追溯、点对点交易、可编程等特点[1]。

区块链的核心技术主要有分布式账本、非对称加密、共识机制、智能合约。下面将一一对这四种技术的原理做介绍。

（1**）分布式账本(Distributed Ledger Technology, DLT)**

分布式账本是一种数字系统，用于记录资产交易，交易记账这一核心事务由分布在不同地方的多个节点共同完成，每一个节点记录有完整的账目，它们都可以参与监督交易的合法性，同时也可以共同为其作证。

区块链的分布式存储的独特性与传统的分布式存储的不同主要体现在两个方面：区块链每个节点都按照块链式结构存储完整的数据，传统分布式存储一般是将数据按照一定的规则分成多份进行存储；区块链每个节点存储都是独立的、地位等同的，依靠共识机制保证存储的一致性，而传统分布式存储一般是通过中心节点往其他备份节点同步数据。区块链中没有任何一个节点可以单独记录账本数据，从而避免了单一记账人被控制或者被贿赂而记假账的可能性。此外，由于记账节点数足够多，理论上，除非所有的节点都被破坏，否则账目就不会丢失，从而保证了账目数据的安全性。

分布式账本的技术特性有如下三点[2]：

1. 是一种去中心化的数据库技术：信息和数据是分散记录的, 并无任何一个中心化机构负责存储或确认数据，而是借助计算机、互联网技术, 通过程序设计、算法安排设计出独有的“无中心、多节点”的数据库模式。任何一个分布式账本数据库均至少有两个以上的节点, 每个节点都有一个“账本”, 每个节点都参与任何一个新项目的处理与验证, 从而生成账本记录。每个节点的账本副本集合起来形成总账, 不需要任何人单独负责记录总账。
2. 使得信息传递及时、高效：各个节点之间的信息交换是直接传递的, 不需要任何第三方中介。每个节点都有自己的服务器, 然后通过网络技术进行通讯, 以确保维护最准确和最新的信息记录。所有参与者可以拥有自己的分类账副本, 所有副本的数据同步更新, 任何一个节点对分类账的任何更改都会在很短时间 (几分钟或者几秒钟) 内反映在其他所有分类账副本中。去中心化和去中介的数据库系统大大缩短了信息传递的时间, 同时也节约了系统维护、中心化机构或者中介的服务成本支出。
3. 重塑了信任机制

分布式账本通过程序算法和计算机网络技术, 实现了数据库信息的高度透明, 所有节点都参与并见证数据库的信息录入, 大家高度认可账本的真实性。由于每个节点的信息变动均受到其他节点的监控并依赖其他节点的认可，因此理论上账本也不可私自篡改，所有参与者 (节点) 对整个数据库信息的真实性、完整性具有高度共识。相较于传统的点对点信息收集模式, 分布式账本技术适应了当前网络社会信息扁平化、共享化的趋势, 可以将社会微观个人的经济行为纳入信息采集模式中, 极大地拓展了信用信息的覆盖范围。由此，分布式账本将传统信任机制中的“相信他人”改为“自我信任”, 这是分布式账本的最大价值, 为其在不同场域的应用提供了无限空间。

**（2）非对称加密算法**

非对称加密算法是一种秘钥的保密方法。需要两个密钥：公开密钥和私有密钥。公钥与私钥是成对的，如果用公钥对数据进行加密，只有用对应的私钥才能解密。因为加密和解密使用的是两个不同的密钥，所以这种算法叫作非对称加密算法。非对称加密算法实现机密信息交换的基本过程是：甲方生成一对密钥并将公钥公开，需要向甲方发送信息的其他角色(乙方)使用该密钥(甲方的公钥)对机密信息进行加密后再发送给甲方；甲方再用自己私钥对加密后的信息进行解密。甲方想要回复乙方时正好相反，使用乙方的公钥对数据进行加密，同理，乙方使用自己的私钥来进行解密。另一方面，甲方也可以使用自己的私钥对机密信息进行签名后再发送给乙方；乙方再用甲方的公钥对甲方发送回来的数据进行验签。本质上说，就是甲方只能用其私钥解密由其公钥加密后的任何信息。

非对称加密算法与对称加密算法的主要区别有：

1. 用于消息解密的密钥值与用于消息加密的密钥值不同;
2. 非对称加密算法比对称加密算法慢数千倍，但安全性更高。非对称加密算法强度复杂、安全性依赖于算法与密钥，使得加密解密速度没有对称加密解密的速度快。

**（3）共识机制**

共识机制是指通过特殊节点的投票，在很短的时间内完成对交易的验证和确认；对一笔交易，如果利益不相干的若干个节点能够达成共识，我们就可以认为全网对此也能够达成共识。

共识机制作为区块链技术的重要组件，其目标是使所有的诚实节点保存一致的区块链视图，同时满足两个性质：一致性，所有诚实节点保存的区块链的前缀部分完全相同；有效性，由某诚实节点发布的信息终将被其他所有诚实节点记录在自己的区块链中[3]。区块链上采用不同的共识机制，在满足一致性和有效性的同时会对系统整体性能产生不同影响。综合考虑各个共识机制的特点，可从4个维度评价各共识机制的技术水平：安全性、扩展性、性能效率、资源消耗。

现今区块链的共识机制可分为四大类：工作量证明机制(PoW）、权益证明机制（PoS）、股份授权证明机制(DPoS)和Pool验证池。

工作量证明机制即对于工作量的证明，是生成要加入到区块链中的一笔新的交易信息(即新区块)时必须满足的要求。在基于工作量证明机制构建的区块链网络中，节点通过计算随机哈希散列的数值解来争夺记账权，而求得正确的数值解以生成区块的能力是节点算力的具体表现。工作量证明机制具有完全去中心化的优点，在以工作量证明机制为共识的区块链中，节点可以自由进出。比特币网络就是应用工作量证明机制来生产新的货币。然而，由于工作量证明机制在比特币网络中的应用已经吸引了全球计算机大部分的算力，其他想尝试使用该机制的区块链应用很难获得同样规模的算力来维持自身的安全。同时，基于工作量证明机制的挖矿行为还造成了大量的资源浪费，达成共识所需要的周期也较长，因此该机制并不适合商业应用。

与要求证明人执行一定量的计算工作不同，权益证明要求证明人提供一定数量加密货币的所有权。权益证明机制的运作方式是，当创造一个新区块时，矿工需要创建一个“币权”交易，交易会按照预先设定的比例把一些币发送给矿工本身。权益证明机制根据每个节点拥有代币的比例和时间，依据算法等比例地降低节点的挖矿难度，从而加快了寻找随机数的速度。这种共识机制可以缩短达成共识所需的时间，但本质上仍然需要网络中的节点进行挖矿运算。因此，PoS机制并没有从根本上解决PoW机制难以应用于商业领域的问题。

股份授权证明机制是一种新的保障网络安全的共识机制。它在尝试解决传统的PoW机制和PoS机制问题的同时，还能通过实施科技式的民主抵消中心化所带来的负面效应。与董事会投票类似，该机制拥有一个内置的实时股权人投票系统，就像系统随时都在召开一个永不散场的股东大会，所有股东都在这里投票决定公司决策。基于DPoS机制建立的区块链的去中心化依赖于一定数量的代表用户，而非全体用户。在这样的区块链中，全体节点投票选举出一定数量的节点代表，由他们来代理全体节点确认区块、维持系统有序运行。同时，区块链中的全体节点具有随时罢免和任命代表的权力。如果必要，全体节点可以通过投票让现任节点代表失去代表资格，重新选举新的代表，实现实时的民主。股份授权证明机制可以大大缩小参与验证和记账节点的数量，从而达到秒级的共识验证。然而，该共识机制仍然不能完美解决区块链在商业中的应用问题，因为该共识机制无法摆脱对于代币的依赖，而在很多商业应用中并不需要代币的存在。

Pool验证池基于传统的分布式一致性技术建立，并辅之以数据验证机制，是目前区块链中广泛使用的一种共识机制。它不需要依赖代币就可以工作，在成熟的分布式一致性算法(Pasox、Raft)基础之上，可以实现秒级共识验证，适合有多方参与的多中心商业模式。不过，Pool验证池也存在一些不足，比如该共识机制能够实现的分布式程度不如PoW机制[4]。

**（4）智能合约**

智能合约（Smart contract ）于1995年由Nick Szabo首次提出，其严格定义是：一个智能合约是一套以数字形式定义的承诺（commitment），包括合约参与方可以在上面执行这些承诺的协议[5]。具体来说，智能合约是一种旨在以信息化方式传播、验证或执行[合同](https://baike.baidu.com/item/%E5%90%88%E5%90%8C/20705" \t "https://baike.baidu.com/item/%E6%99%BA%E8%83%BD%E5%90%88%E7%BA%A6/_blank)的计算机协议，允许在没有第三方的情况下进行可信交易，这些交易可追踪且不可逆转，因此智能合约是区块链被称之为“去中心化的”重要原因。智能合约的目的是提供优于传统合约的安全方法，并减少与合约相关的其他交易成本。智能合约包含了有关交易的所有信息，只有在满足要求后才会执行结果操作。智能合约和传统纸质合约的区别在于智能合约是由计算机生成的。因此，代码本身解释了参与方的相关义务。事实上，智能合约的参与方通常是互联网上的陌生人，受制于有约束力的数字化协议。本质上，智能合约是一个数字合约，除非满足要求，否则不会产生结果。

1. **联盟链和公有链的异同**

**异：**

1. 公有链的去中心化程度更高。以比特币以及以太坊为代表的公有区块链，不受第三方机构控制，世界上所有的人都可读取链上的数据记录、参与交易以及竞争新区块的记账权等。程序开发者无权干涉用户，各参与者（即节点）可自由加入以及退出网络，并按照意愿进行相关操作。
2. 联盟链是介于公有链以及私有链之间的区块链，可实现“部分去中心化”。链上各个节点通常有与之相对应的实体机构或者组织，参与者通过授权加入网络并组成利益相关联盟，共同维护区块链运行。从某种程度上来说，联盟链也属于私有链的范畴，只是私有化程度有所不同而已。为此其同样具有成本较低、效率较高的特点，适用于不同实体间的交易、结算等B2B交易。
3. 公有链一般会通过代币机制鼓励参与者竞争记账，来确保数据的安全性。比特币、以太币都是典型的公有链。主要特点是用户免受开发者影响、所有数据默认公开、访问门槛低。
4. 相比于公共区块链，联盟区块链在效率和灵活性上更有优势，主要体现为以下几点：（1）交易成本更便宜。交易只需被几个受信的高算力节点验证就可以了，而无需全网确认。（2）结点可以很好地连接，故障可以迅速通过人工干预来修复，并允许使用共识算法减少区块时间，从而更快完成交易。（3）如果读取权限受到限制，可以提供更好的隐私保护。（4）更灵活。

**同：**都是一种区块链，即分布式数据库，有去中心化的特点。

1. **信任链是如何建立的？**

信任链经营是供应链经营的本质，一方面需要有全球化视野，另一方面，需要认真思考如何从全球范围内寻找服务于供应链的全球信任链资源，协同合作发展。

在区块链中，信任链是通过证书机制来建立的。证书生成流程如下：

1. **联盟链委员会在**本地生成私钥ca.key，自签生成根证书ca.crt；
2. **联盟链成员机构在**本地生成私钥agency.key，由本地私钥生成证书请求文件agency.csr并将其发送至联盟链委员会
3. 联盟链委员会使用ca.key对证书请求文件agency.csr进行签发，得到联盟链成员机构证书agency.crt并将之发送至对应成员
4. **节点/SDK在**本地生成私钥node.key，由本地私钥生成证书请求文件node.csr并将之发送至联盟链成员机构
5. 盟链成员机构使用agency.key对证书请求文件node.csr进行签发，得到节点/SDK证书node.crt并将之发送至对应实体
6. **链式存储和MPT存储**

链式存储，又叫链接存储，是相对顺序存储而言的，具体是指在计算机中用一组任意的[存储单元](https://baike.baidu.com/item/%E5%AD%98%E5%82%A8%E5%8D%95%E5%85%83/8727749" \t "https://baike.baidu.com/item/%E9%93%BE%E5%BC%8F%E5%AD%98%E5%82%A8%E7%BB%93%E6%9E%84/_blank)存储线性表的[数据元素](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%85%83%E7%B4%A0/715313" \t "https://baike.baidu.com/item/%E9%93%BE%E5%BC%8F%E5%AD%98%E5%82%A8%E7%BB%93%E6%9E%84/_blank)，这组存储单元可以是连续的,也可以是不连续的，它不要求逻辑上相邻的元素在物理位置上也相邻，因此它没有[顺序存储结构](https://baike.baidu.com/item/%E9%A1%BA%E5%BA%8F%E5%AD%98%E5%82%A8%E7%BB%93%E6%9E%84" \t "https://baike.baidu.com/item/%E9%93%BE%E5%BC%8F%E5%AD%98%E5%82%A8%E7%BB%93%E6%9E%84/_blank)所具有的弱点,但也同时失去了[顺序表](https://baike.baidu.com/item/%E9%A1%BA%E5%BA%8F%E8%A1%A8" \t "https://baike.baidu.com/item/%E9%93%BE%E5%BC%8F%E5%AD%98%E5%82%A8%E7%BB%93%E6%9E%84/_blank)可[随机存取](https://baike.baidu.com/item/%E9%9A%8F%E6%9C%BA%E5%AD%98%E5%8F%96" \t "https://baike.baidu.com/item/%E9%93%BE%E5%BC%8F%E5%AD%98%E5%82%A8%E7%BB%93%E6%9E%84/_blank)的优点。

链式存储的特点有：

1. [存储密度](https://baike.baidu.com/item/%E5%AD%98%E5%82%A8%E5%AF%86%E5%BA%A6/3236466" \t "https://baike.baidu.com/item/%E9%93%BE%E5%BC%8F%E5%AD%98%E5%82%A8%E7%BB%93%E6%9E%84/_blank)小(链式存储结构中每个结点都由数据域与指针域两部分组成，相比顺序存储结构增加了存储空间)；
2. 逻辑上相邻的节点物理上不必相邻；
3. 插入、删除灵活 (不必移动节点，只要改变节点中的指针)；
4. 查找节点比较慢；
5. 由于簇（簇是计算机存储数据的基本单位）是随机分配的，使得数据删除后覆盖几率降低，恢复可能提高。

# MPT指Merkle Patricia Trie，实际上是一种Trie树变种，是以太坊中一种非常重要的数据结构，用来存储用户账户的状态及其变更（状态树）、交易信息（交易树）、交易的收据信息（收据树）。MPT实际上是两种数据结构的组合，分别是Merkle树 和 Patricia 树。

先介绍Trie树。Trie树又称前缀树或字典树，是一种有序树，用于保存关联数组，其中的键通常是字符串。与二叉查找树不同，键不是直接保存在节点中，而是由节点在树中的位置决定。一个节点的所有子孙都有相同的前缀，也就是这个节点对应的字符串，而根节点对应空字符串。一般情况下，不是所有的节点都有对应的值，只有叶子节点和部分内部节点所对应的键才有相关的值。Trie树的每个节点是一个确定长度的数组，数组中每个节点的值是一个指向子节点的指针，最后有个标志域，标识这个位置为止是否是一个完整的字符串，并且有几个这样的字符串。

Trie树的优点有：

1. 相比于哈希表，使用前缀树来进行查询拥有共同前缀key的数据时十分高效；
2. 在前缀树不会存在哈希冲突的问题：也就是说一个key永远只对应一个value；
3. 在代码的实现上，对于root节点，中间节点，叶子节点只需要一种节点结构。

Trie树的缺点是直接查找效率低下，前缀树的查找效率是O(m)，m为所查找节点的key长度，一次查找会有m次IO开销，相比于直接查找，无论是速率、还是对磁盘的压力都比较大，而哈希表的查找效率为O(1)；此外，Trie树存储不平衡可能会造成空间浪费，当存在一个key值内容很长的节点，如果树中没有与他相同前缀的分支，为了存储该节点，需要创建许多非叶子节点来构建根节点到该节点间的路径，导致整棵树不平衡，相比于哈希表，Trie额外产生了过多的“无用”中间节点造成了存储空间的浪费。

Patricia树，或称Patricia trie，或crit bit tree，压缩前缀树，是一种更节省空间的Trie。对于基数树的每个节点，如果该节点是唯一的儿子的话，就和父节点合并，这样可以缩短 Trie 中不必要的深度，大大加快搜索节点速度。

Merkle Tree，通常也被称作Hash Tree，顾名思义，就是存储hash值的树。Merkle树的叶子是数据块(例如，文件或者文件的集合)的hash值，非叶节点是其对应子节点串联字符串的hash。Merkle Tree可以看做Hash List的泛化，Hash List可以看作一种特殊的Merkle Tree，即树高为2的多叉Merkle Tree。在最底层，和哈希列表一样，数据被分成小的数据块，有相应地哈希和它对应。但是往上走，并不是直接去运算根哈希，而是把相邻的两个哈希合并成一个字符串，然后运算这个字符串的哈希，这样，每两个哈希就结婚生子，得到了一个“子哈希”。如果最底层的哈希总数是单数，那到最后必然出现一个单身哈希，这种情况就直接对它进行哈希运算，所以也能得到它的子哈希。于是往上推，依然是一样的方式，可以得到数目更少的新一级哈希，最终必然形成一棵倒挂的树，到了树根的这个位置，这一代就剩下一个根哈希了，即 Merkle Root。

Merkle Tree和HashList的主要区别是，可以直接下载并立即验证Merkle Tree的一个分支。因为可以将文件切分成小的数据块，这样如果有一块数据损坏，仅仅重新下载这个数据块就行了。如果文件非常大，Merkle tree可以一次下载一个分支，然后立即验证这个分支，如果分支验证通过，就可以下载数据了，而Hash list只有下载整个hash list才能进行验证。

MPT（Merkle Patricia Trie）是在Patricia树的基础上，添加了**加密安全**功能，引用下一个节点的hash值，其次引入了很多**节点类型**来提高效率：Null Node（空节点），Extension Node（拓展节点），Leaf Node（叶子节点），Branch Node（分支节点）。空节点表示为空，在代码中就是一个空串；扩展节点包含两个字段，第一个字段是剩下的Key的可以至少被两个剩下节点共享的部分的半字节编码，第二个字段是下一个node的hash值，通过合并只有一个子节点的父节点和其子节点来缩短 trie 的深度；叶子节点也包含两个字段， 第一个字段是剩下的Key的半字节编码,而且半字节编码方法的第二个参数为true， 第二个字段是value；分支节点包含了17个字段，其前16个项目对应key中的16个可能的十六进制字符，如果有一个键值对在这个分支点终止，则最后的一个字段用于存放该键值对的值，即分支点既可以是搜索路径上的终止，也可以是路径的中间节点。

MPT树中另外一个重要的概念是一个特殊的十六进制前缀(hex-prefix, HP)编码，用来对key进行编码。因为字母表是16进制的，所以每个节点可能有16个孩子。因为有两种[key,value]节点(叶节点和扩展节点)，引进一种特殊的终止符标识来标识key所对应的值是真实的值还是其他节点的hash。如果终止符标记被打开，那么key对应的是叶节点，对应的值是真实的value；如果终止符标记被关闭，那么值就是用于在数据块中查询对应的节点的hash。无论key奇数长度还是偶数长度，HP都可以对其进行编码。

1. **Gas在智能合约中的作用**

gas是以太坊使用的特殊单位的名称，是执行交易或合同的内部定价，它衡量一个动作或一系列动作需要执行多少“工作”，gas的作用有：

1. 作为一种费用的计算方式，使以太坊网络的交易或计算成本有一个稳定的价值，有助于确保提交给网络的交易支付适当的费用，确保网络不会因为执行大量对任何人无价值的密集工作而陷入困境。
2. 保证以太坊中的复杂计算的安全性。任何失控的程序只会在请求运行的人提供的资金的情况下持续下去，当资金停止时，矿工们就停止工作，在程序中犯的错误只会影响付费使用它的人，网络的其他部分不会因为程序中的错误而遭受性能问题。
3. 确保没有任何东西可以永远运行，人们会仔细考虑它们运行的代码，保持矿工和用户的安全不受恶劣的代码影响。
4. **EVM中的数据存储结构**

可以从EVM的内存、堆栈、持久化存储来了解其数据存储结构。

内存结构提供了数据缓存的功能，提供了合约调用合约等过程中，子合约数据的临时存储。可以把内存看作一个链表，每个节点就是一个内存块，链表是可扩展的数据结构，以块为单位，支持内存的动态扩展。在该结构中有一个指针softSize,指示内存有效数据的末尾，以0为起始，以字为移动单位。内存主要的外部接口有：read、write、readWord、readByte、internalSize、extendAndWrite。

EVM的执行模型基于栈结构，像JVM一样，这里提供的堆栈就是用来存储字节码执行过程中的中间数据等，而且是直接继承自Java SDK的栈结构的。栈内部是一个定长数组，在需要扩展时会创建新数组，然后浅拷元素到新数组。栈的主要外部接口有：pop、push、swap。

EVM的持久化存储主要是独立于区块存储之外，存储以太坊的账户（外部账户、合约账户）、合约代码以及合约的状态。库是持久存储与以太坊联系的桥梁，包含3个缓存：账户缓存（包含地址、余额、nonce、codeHash、storageRoot）、代码缓存(包含合约地址、代码哈希和代码)、存储缓存（包含合约地址、合约状态名、合约状态值）。它封装了底层Key-Value数据库，并提供了批量提交和读、写缓存，并以此为基础提供了区块执行过程中的多级缓存、快照或者说事务提交。

1. **群组架构的好处**

群组架构自底向上主要划分为网络层、群组层，网络层主要负责区块链节点间通信，群组层主要负责处理群组内交易，分为核心层、接口层和调度层[5]。

核心层提供底层存储和交易执行接口，主要包括存储(AMDB/storage/state)和执行(EVM)两大模块，存储模块负责存储从底层数据库中存储或读取的群组账本的区块数据、区块执行结果、区块信息以及系统表等信息，执行(EVM)模块主要负责执行交易。

接口层是访问核心层的接口，接口层包括交易池(TxPool)、区块链(BlockChain)和区块执行器(BlockVerifier)三个模块。交易池是客户端与调度层交互的接口，负责从客户端或者其他节点收到的新交易，共识模块会从中取出交易打包处理，同步模块从中取出新交易进行广播。区块链模块是核心层与调度层交互的接口，是调度层访问底层存储和执行模块的唯一入口，调度层可通过该模块提交新区块和区块执行结果，查询历史区块等信息。区块链模块也是RPC模块与核心层的接口，RPC模块通过区块链模块可获取区块、块高以及交易执行结果等信息。**区块执行器**与调度层交互，负责执行从调度层传入的区块，并将区块执行结果返回给调度层。

调度层包括共识模块和同步模块。共识模块主要负责执行客户端提交的交易，并对交易执行结果达成共识，包括打包(Sealer)线程和共识(Engine)线程，Sealer线程负责从交易池获取未执行交易，并将交易打包成区块；Engine线程负责对区块执行结果进行共识，目前主要支持PBFT和Raft共识算法。同步模块来保证客户端的交易尽可能发送到每个共识节点，主要包括交易同步和区块同步。交易同步是指客户端通过RPC向指定群组交易池提交新交易时，会唤醒相应群组同步模块的交易同步线程，该线程将所有新收到的交易广播到其他群组节点，其他群组节点将最新交易插入到交易池，保证每个群组节点拥有全量的交易。区块同步是指向其他节点发送自己节点的最新块高，其他节点发现块高落后于其他节点后，会主动下载最新区块。

每个群组均运行着一个独立的账本，持有一个账本白名单，用于维护该群组的节点列表。群组架构中，所有群组共享P2P网络，群组层传递给网络层的数据包中含有群组ID信息，接收节点根据数据包中的群组ID，将收到的数据包传递给目标节点的相应群组。

**群组架构的好处有**：

1. 引入**账本白名单**机制，群组间通信数据隔离
2. 仅可通过发交易共识的方式修改账本白名单，保证了账本白名单群组内一致性
3. 通过账本白名单，有效地防止群组节点获取其他群组通信消息，保障了群组网络内部通信的隐私性
4. 同步机制避免了区块链网络中由于机器性能不一致或者新节点加入导致部分节点区块高度落后于其他节点的问题
5. 允许网络存在多个账本，使得系统吞吐能力能横向扩展[6]
6. 各群组独立执行共识流程，解决了多群组同一个共识模块导致的性能瓶颈问题。按群组的粒度划分，由群组内参与者决定如何进行共识，一个群组内的共识不受其他群组影响，群组内维护自己的交易事务和数据，使得各群组之间解除耦合，独立运作，同时也便于进行横向扩展
7. 一个组相当于一条链，实现了传统多链的扩展目的，同时一个节点可以参与到多条链，能够极大地简化运维复杂度，降低管理成本
8. **分布式存储有什么优势？**

分布式存储系统是指将数据[分散存储](https://baike.baidu.com/item/%E5%88%86%E6%95%A3%E5%AD%98%E5%82%A8" \t "https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E5%AD%98%E5%82%A8%E7%B3%BB%E7%BB%9F/_blank)在多台独立的设备上。传统的[网络存储](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C%E5%AD%98%E5%82%A8" \t "https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E5%AD%98%E5%82%A8%E7%B3%BB%E7%BB%9F/_blank)系统采用集中的[存储服务器](https://baike.baidu.com/item/%E5%AD%98%E5%82%A8%E6%9C%8D%E5%8A%A1%E5%99%A8" \t "https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E5%AD%98%E5%82%A8%E7%B3%BB%E7%BB%9F/_blank)存放所有数据，存储服务器成为系统性能的瓶颈，也是可靠性和安全性的焦点，不能满足大规模存储应用的需要。分布式网络存储系统采用可扩展的[系统结构](https://baike.baidu.com/item/%E7%B3%BB%E7%BB%9F%E7%BB%93%E6%9E%84" \t "https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E5%AD%98%E5%82%A8%E7%B3%BB%E7%BB%9F/_blank)，利用多台存储服务器分担存储负荷，利用位置服务器定位存储信息，它不但提高了系统的可靠性、可用性和存取效率，还易于扩展。分布式存储的具体优势如下：

1. 增大了系统的容量，避免了脖子照应，有助于解决海量数据存储问题；
2. 利用冗余数据维护技术，通过分散在网络上大量存储节点之间的协作,实现长久可靠的数据存储服务、P2P网络存储和无线网络存储；
3. 加强了系统的可用性，避免了单点故障；
4. 使系统模块化，提高了模块的重用度，同时系统的可扩展性也更高了；
5. 提高了开发和发布速度，因为软件服务模块被拆分，所以开发和发布可以并行进行；
6. 采用分布式的系统结构，利用多台存储服务器分担存储负荷，利用位置服务器定位存储信息，不但提高了系统的可靠性、可用性和存取效率，还易于扩展，将通用硬件引入的不稳定因素降到了最低；
7. 高性能：通过将热点区域内数据映射到高速存储中，提高系统响应速度，一旦这些区域不再是热点，存储系统就将它们移出高速存储；
8. **支持分级存储：**通过网络进行松耦合链接，分布式存储允许高速存储和低速存储分开部署，或者任意比例混布。在不可预测的业务环境或者敏捷应用情况下，分层存储的优势可以发挥到最佳，解决了目前缓存分层存储最大的问题，即当性能池读不命中后，从冷池提取数据的粒度太大，导致延迟高，从而给造成整体的性能的抖动的问题；
9. 分区容错性[7]：分布式存储系统中的多台服务器通过网络进行连接，但是无法保证网络是一直通畅的，分布式系统具有一定的容错性来处理网络故障带来的问题，当一个网络因为故障而分解为多个部分的时候，分布式存储系统仍然能够工作；
10. **多副本的一致性：**分布式存储采用了多副本备份机制，在存储数据之前，分布式存储对数据进行了分片，分片后的数据按照一定的规则保存在集群节点上。为了保证多个数据副本之间的一致性，分布式存储通常采用的是一个副本写入、多个副本读取的强一致性技术，使用镜像、条带、分布式校验等方式满足租户对于可靠性不同的需求。在读取数据失败的时候，系统可以通过从其他副本读取数据，重新写入该副本进行恢复，从而保证副本的总数固定；当数据长时间处于不一致状态时，系统会自动恢复数据，同时租户可设定数据恢复的带宽规则，最小化对业务的影响。
11. **存储系统标准化：随**着分布式存储的发展，存储行业的标准化进程也不断推进，分布式存储优先采用行业标准接口（SMI-S或OpenStack Cinder）进行存储接入。在平台层面，通过将异构存储资源进行抽象化，将传统的存储设备级的操作封装成面向存储资源的操作，从而简化了异构存储基础架构的操作，实现了存储资源的集中管理，并能够自动执行创建、变更、回收等整个存储生命周期流程。基于异构存储整合的功能，用户可以实现跨不同品牌、介质地实现容灾，如用中低端阵列为高端阵列容灾，用不同磁盘阵列为闪存阵列容灾等等，这从侧面降低了存储采购和管理成本。
12. **并行计算**

并行计算（Parallel Computing）是指同时使用多种计算资源解决计算问题的过程，是提高计算机系统计算速度和处理能力的一种有效手段。它的基本思想是用多个处理器来协同求解同一问题，即将被求解的问题分解成若干个部分，各部分均由一个独立的处理机来并行计算。并行计算系统既可以是专门设计的、含有多个处理器的超级计算机，也可以是以某种方式互连的若干台的独立计算机构成的集群，通过并行计算集群完成数据的处理，再将处理的结果返回给用户。

并行计算可分为时间上的并行和空间上的并行，并行计算科学中主要研究的是空间上的并行问题。时间上的并行是指流水线技术，空间上的并行是指多个处理机并发的执行计算，即通过网络将两个以上的处理机连接起来，达到同时计算同一个任务的不同部分。

并行计算有5种访存模型：均匀访存模型（UMA）、[非均匀访存模型](https://baike.baidu.com/item/%E9%9D%9E%E5%9D%87%E5%8C%80%E8%AE%BF%E5%AD%98%E6%A8%A1%E5%9E%8B" \t "https://baike.baidu.com/item/%E5%B9%B6%E8%A1%8C%E8%AE%A1%E7%AE%97/_blank)（NUMA）、全高速缓存访存模型（COMA）、一致性高速缓存非均匀存储访问模型（CC-NUMA）、[非远程存储访问模型](https://baike.baidu.com/item/%E9%9D%9E%E8%BF%9C%E7%A8%8B%E5%AD%98%E5%82%A8%E8%AE%BF%E9%97%AE%E6%A8%A1%E5%9E%8B" \t "https://baike.baidu.com/item/%E5%B9%B6%E8%A1%8C%E8%AE%A1%E7%AE%97/_blank)（NORMA）。

计算模型方面，并行计算机没有一个统一的计算模型，主要有[PRAM模型](https://baike.baidu.com/item/PRAM%E6%A8%A1%E5%9E%8B" \t "https://baike.baidu.com/item/%E5%B9%B6%E8%A1%8C%E8%AE%A1%E7%AE%97/_blank)，[BSP模型](https://baike.baidu.com/item/BSP%E6%A8%A1%E5%9E%8B" \t "https://baike.baidu.com/item/%E5%B9%B6%E8%A1%8C%E8%AE%A1%E7%AE%97/_blank)，[LogP](https://baike.baidu.com/item/LogP" \t "https://baike.baidu.com/item/%E5%B9%B6%E8%A1%8C%E8%AE%A1%E7%AE%97/_blank)模型，C^3模型等。

并行算法是[并行计算](https://baike.baidu.com/item/%E5%B9%B6%E8%A1%8C%E8%AE%A1%E7%AE%97" \t "https://baike.baidu.com/item/%E5%B9%B6%E8%A1%8C%E7%AE%97%E6%B3%95/_blank)中非常重要的问题，主要分为多机并行（如MPI技术）和多线程并行（如[OpenMP](https://baike.baidu.com/item/OpenMP" \t "https://baike.baidu.com/item/%E5%B9%B6%E8%A1%8C%E7%AE%97%E6%B3%95/_blank)技术）。

对并行计算的性能度量的加速比评测方法有：Amdahl定理、Gastofson定理、Sun-Ni定理。可扩放性标准有等效率标准、等速度标准、[平均延迟标准](https://baike.baidu.com/item/%E5%B9%B3%E5%9D%87%E5%BB%B6%E8%BF%9F%E6%A0%87%E5%87%86" \t "https://baike.baidu.com/item/%E5%B9%B6%E8%A1%8C%E8%AE%A1%E7%AE%97/_blank)等。

1. **当前区块链实施的难度**

当前实施区块链的难点有：

1. **激励制度设计、**一致性问题、安全问题。
2. 维护成本高：传统的中心化数据库只需要写入一次就可以，但是区块链需要写入上千；传统的中心化数据可以一次性检测数据，区块链需要检查上千次数据；传统中心化数据库需要只需要一次性将数据转移到存储器，区块链则需要将数据转移上千次。维护区块链的成本是数量级增长，而且这部分成本还需要根据使用性能进行更改。
3. 升级、扩容非常困难：同样的数据需要存在成百上千个地方，而不是像中心化那样，在同个地方。传输，验证和存储的开销都是很大的，因为区块链上任何数据库的复制都需要花钱，而不像中心化的数据库那样，只需要付费一次就可以。此外，升级也难以和之前的兼容。
4. 开发人力资源储备不足。

**参考文献**

[1] 张亮,张翰林,孔凡玉,于佳.基于Ethereum的房地产供应链系统设计与实现[J/OL].计算机工程与应用:1-13[2019-07-08].http://kns.cnki.net/kcms/detail/11.2127.TP.20190706.0852.006.html.

[2] 赵磊.证券交易中的信用机制——从中央存管(CSD)到分布式账本(DLT)[J].财经法学,2019(03):51-65.

[3] 韩璇. 区块链技术中的共识机制研究[A]. 中国计算机学会.第32次全国计算机安全学术交流会论文集[C].中国计算机学会,2017:6.

[4] 唐文剑，吕雯等编著．区块链将如何重新定义世界：机械工业出版社，2016.06

# [5] FISCO BCOS 2.0原理解析： 群组架构的设计，2019，<https://blog.csdn.net/FISCO_BCOS/article/details/89379067>

# [6] FISCO BCOS整体架构---群组架构,2019, TaroYoVen, <https://blog.csdn.net/TaroYoVen/article/details/89191066>

[7] 陈敏，张东，张引，亓开元编著,大数据浪潮 大数据整体解决方案及关键技术探索=BIG DATA INSPIRATION,华中科技大学出版社,2015.10,29-30