**NEO**

**DBFT**

DBFT的算法：参与记账的是超级节点，普通节点可以看到共识过程，并同步账本信息，但不参与记账。总共n个超级节点分为一个议长和n-1个议员，议长会轮流当选。每次记账时，先有议长发起区块提案（拟记账的区块内容），一旦有至少（2n+1)/3个记账节点（议长加议员）同意了这个提案，那么这个提案就成为最终发布的区块，并且该区块是不可逆的，所有里面的交易都是百分之百确认的。如果在一定时间内还未达成一致的提案，或者发现有非法交易的话，可以由其他记账节点重新发起提案，并且当至少[(2n+1)/3]个记账节点同意后，就形成最终确定的区块了。

通过选举一部分节点来参与共识，提高出块效率，增大了tps。

NEO经常被质疑过于中心化，说的就是DBFT共识机制下投票选出的仅有7个共识节点。这些代理节点是通过静态选出的，并完全由项目方部署（DPOS下的记账节点是会动态变化的）

1. DBFT是一种弱中心化的共识机制，与dpos类似；而pos和pow是完全去中心化的；
2. DBFT的好处是，具有最终性，也就是说只要出来的区块，一定是不可逆的，不会导致分叉；而dpos、pos、pow理论上都是存在分叉导致交易会被取消的可能性。
3. DBFT的问题在于，安全性较低，只要有(n-1)/3个节点是恶意节点，就会使得系统无法正常运行；而pow、pos都可以容忍低于50%的算力（币天），Eos的dpos，也可以保证50%以下的安全性，以及只要不超过2/3的节点为恶意节点，就不会进入不可逆状态。
4. DBFT还有一个问题是，相比dpos，出块速度还是较慢，并且eos的dpos可以在一秒内进入不可逆状态（等于DBFT的最终性）。

**EOS**

**BFT-DPOS共识机制总结**

BFT + DPOS + 小块：

    该协议是为了加快出块速度，该改进的核心思想是让同一个节点产生6个小块。这个能提升性能的核心原因是6个小块的产生没有等待确认的环节。在BFT+DPOS算法中，一个节点要创建新块，必须等上一个块被确认(或者超时)，因为它不确定上一个节点是否作恶是否会被确认，所以它只能等。但是如果上一个块时也是该节点产生的，它自然不需要等上一块确认，因为它知道上一个块是真实的，不是作恶的块，最后肯定会被确认，所以安心生产下一块。

**而在EOS的机制下，节点是定向广播的。21节点的位置是透明的，会选择最短路径来规定广播顺序。**

优点：21个超级节点（主力见证人节点） + 100个备选见证人节点；  
0.5秒出块时间 + 1秒全网确认；  
每个主力见证人节点通过协商方式确定各自出块顺序，并且每轮产生6个区块以减少网络延时的影响，见证人间按顺序处理交易，可尽量减少地理影响；  
当21个主力见证人的15个确认交易后，交易即不可逆转；  
当达到不可逆转状态后，就无法分叉。

**缺点：**

1.不是完全去中心化，可能会有多个中心之间共同串通而损害整个社区利益的行为。  
2.依赖于投票机制  
投票制度其实有以下问题，首先有可能最后投票的参与度会很低，影响投票结果。其次也会可能有这种情况，例如用户把币都存在了交易所，交易所有可能会代替他们去投票，但是用户并不是很在意到底交易所会把票投向何处。也就是说有时候代币持有者的兴趣点和用户的是可能不完全一样的。

##### Zilliqa共识

* 我们用数字签名替换pbft中消息认证来有效地减少O（n）的通信开销。
* 与此同时，为了让其他节点能够验证协议，一种典型的方法是从诚实的多数收集签名并将它们附加到协议中，从而导致协商规模与协商组的大小成线性关系。 为了改善这一点，我们使用EC-Schnorr多重签名来将几个签名聚合成O（1） - 大小多重签名

在ZILLIQA中，我们使用PBFT作为基础共识协议，并采用两轮EC-Schnorr多重签名来替换PBFT中的准备阶段和提交阶段。 下面将解释对PBFT阶段的不同修改。

* 预准备阶段: 与标准PBFT中一样，领导者将TX-Block或声明（由领导者签名）分发给共识组中的所有节点。
* 准备阶段：所有诚实的节点检查TX块的有效性，并且领导者收集来自超过2/3\*n个节点的响应。 这保证领导者提出的陈述是安全的并且与以前的所有历史一致。 签名是使用EC-Schnorr多重签名生成的。 领导者还构建签署TX块的节点的位图
* 提交阶段：为了确保超过2/3\*n的节点知道超过2/3\*n节点验证了TX-Block的事实。我们进行第二轮EC-Schnorr多重签名。 正在签署的声明是上一轮生成的多重签名。

在三个阶段结束时，就领导者提出的TX-Block将达成共识。

减少通信的方法：







