十分钟读懂 Hashgraph

Hashgraph 是一种全新的分布式账本共识机制技术和数据结构，与区块链技术相比，其更快、更公平和更安全。InterValue 在共识机制方面，重点借鉴 Hashgraph 这些优点，设计并实现更先进的基于 HashNet 的全新共识机制解决现有区块链基础设施存在的各类问题。下面，InterValue 带大家十分钟读懂：什么是 Hashgraph？

一、DAG 共识和 HashGraph

所谓区块链的共识机制就是维护一个大家都认可的交易顺序。

中本聪在比特币网络中设计了 PoW 机制，矿工通过竞争一个时间段内的交易打包权力，获胜的矿工根据手续费高低挑选这个时间段内发生的交易的交易顺序，并且把这些交易打包到一个区块中，区块跟区块之间链接完成这个网络的交易时间顺序，完成比特币网络的共识。

不管是 POW、POS 还是 DPOS，这些共识算法通过竞争获得产生区块的方法确实解决了共识问题，却不能称得上优雅，每一个区块的形成过程似乎都是在把大部分交易拒之门外，留下一些满足矿工口味的交易打包到区块中。

基于区块的共识的问题

- **不够快**

矿工之间通过竞争一段时间的交易打包权获得激励，每笔交易被确认，首先需要足够幸运或者足够多手续费才有可能被矿工选中，交易即时被矿工选中，还要等待一个出块时间，这个就是出块延迟的问

题。比特币需要十分钟才能被打包到区块，以太坊需要十几秒才被得到确认。不管是比特币还是以太坊，性能上离大规模商用还有很远距离。

- **不公平**

矿工只会为了获得 coinbase 激励和交易手续费，然后埋头苦干不去作恶。这么想就太 too young,too simple。矿工可能在一些优质 ICO 活动中，矿工可以优先确认自己的交易，这就比老老实实挖矿划算多了。当引入未来比较明显的激励的时候，矿工完全会作恶，在设计之处，模型里只有当下的激励，但如果考虑到未来更大的激励的时候，矿工就会超出模型设计时的角色定位。

二、DAG 是不是为了性能牺牲了安全

需要首先讲讲 iota 和 bytebal。

不少人质疑这些 DAG 应用的安全性，是不是还满足去中心化的共识。包括 iota 是不是过多依赖还没开源的 validator，bytebal 的 witness 节点有没有作恶空间。

矿工在一个很长的出块时间里，矿工有足够空间从众多交易中选择对自己有利的交易来打包。在 byteball 的 witness 节点中 ,witness 节点能做的事情很有限，收到一笔交易后，能做的只是记录交易的时间戳，因为没有出块时间留出来的时间空挡，需要马上处理下一笔交易。所以 byteball 创始人一直强调不要把 witness 跟矿工混淆，witness 扮演的更多的是一个 checkpoint 的角色 ,只是帮系统记录交易发生的先后顺序，没有留出作恶的空间。

HashGraph 是 Swirds 公司抱有专利的一种分布式账本共识，也没有区块概念，交易跟交易直接组成 DAG。目前 Swirds 公司没有做关于 ICO 的打算，目前更多在以技术服务商的身份在供应链和物联网方向做应用落地。

HashGraph 通过 gossip of gossip 协议，让每个节点都维护着所有节点跟其他节点的通信历史，每个节点在完成拜占庭协议时，居然不需要经过网络多轮通讯，节点本地环境就可以直接模拟拜占庭决议。

性能角度，目前 hashgraph 共识已经满足了几十万的并发，性能瓶颈已经不是协议本身，而是到了网络 IO 层。

安全角度，hashgraph 的数学上可以证明满足异步拜占庭容错，至少跟比特币一样安全。

公平角度，没有矿工这种超级权利的角色存在。

Leemon Baird 有提到 ,hashgraph 的共识也很适合构建公有链。

三、Hashgraph 的特点

Hashgraph 是“互联网和分散技术的未来”，被设计成一种可替代区块链的高级一致性机制/数据结构。

- 超快速交易
- 公平：用数学通过一致的时间节点确保公平运用数学理论通过一致性时间戳证明得到的公平意味着任何人都不能操纵交易的秩序。
- 安全 :银行级安全(异步拜占庭容错 ,Asynchronous Byzantine Fault Tolerant），排除了不良行为，防止其达成共识。

- 独特性：Hashgraph 使用虚拟投票和小道消息而非 POW 或 POS 来达成分布式一致性，这是非常有效的。

四、Hashgraph 初认识

Hashgraph 是一种数据结构和共识算法。Hashgraph 不是数字货币，也不是区块链（因为它其实是 DAG 图，并非是链式结构），严格说也不单单是协议。Hashgraph 更像是一个底层的出块层而非一个完整的系统。Hashgraph 能为分布式 APP 提供高效、公平、安全的基础设施。高吞吐量和异步拜占庭容错（ ABFT ）的特点，使得 Hashgraph 在公链和私有链领域都有潜在的使用价值，并且，在保证去中心化的同时不需要繁重的工作量证明。

Hashgraph 用的是 gossip protocol，原理简单说来就是消息像八卦一样告诉自己的邻居 ,邻居再告诉他的邻居 ,这样一直广播出去，直到全网都知道这个消息。

- 1.Hashgraph 中的每个节点都可以将新创建的交易和从其他人那里接收到的交易的签名信息（event）传播给其他随机选择的邻居。（如何定义邻居呢？）
- 2.这些邻居将收到的事件与从其他节点收到的信息聚合成一个新事件，然后将其发送给其他随机选择的邻居。
- 这个过程一直持续到所有节点都知道在开始时创建或接收的信息。

由于八卦协议的快速收敛性，每条新信息都可以快速到达网络中的每个节点。

gossip protocol 的传播历史可以通过有向图来说明，每个节点维护一个图，表示每次交易的转发者/证人序列。

Hashgraph 假设：不到 1/3 的节点是拜占庭节点（作弊节点，延迟节点，丢消息节点）

如何确认交易呢：每个节点可以根据是否有超过全网 2/3 的节点（也就是目击者）来确定交易是否有效。

FLP 不可能定理：在网络可靠且存在节点失效的异步分布式系统中，不存在一个可以解决一致性问题的确定性算法。

在异步系统中，即使在仅有一个故障节点的简单情况下，不可能存在确定性的共识协议。在有拜占庭节点或恶意节点的场景下，共识协议要么是非确定性异步（典型的 PoW），要么是确定性非完全异步（典型的 PBFT）。

PBFT（实用拜占庭容错）大大优化了消息传播的复杂度（节点通信基本都是同步的），但是实际使用中差不多也就支持到 100 个节点就是极限了，因此 BFT 算法只适用于非公链场景。

Hashgraph 对共识定义做了一些放宽，本来应该在有限轮通信之后，会取得共识，Hashgraph 是在极小概率下共识算法可能会无线执行，但这一概率几乎为 0。

Hashgraph 的共识算法是非确定性的，但那是能保证最终确定性，同时因为所有节点都是对等节点，避免了潜在的 DDOS 攻击风险。

五、算法深入理解

- 谣言协议 （Gossip about Gossip）

- 虚拟投票（Virtual Voting）

  一传十十传百的谣言协议，最后让整个网络都被消息所传达到。

  通信机制保证了消息在这个节点网络中被传达，但要取得共识，还需要虚拟投票机制。

  这是一个非常复杂的算法，有太多新的概念：

- 事件（event）

  这个很好理解，就好比区块链中的区块，event 是一个包含有两个哈希指针的数据结构，并且可以包括 0 个或若干交易信息，节点在创建 event 的同时会加上 timestamp 并且对整个 event 数字签名。

- 绝对多数（supermajority）

  超过 2/3 以上节点的数量。

- 可见（seeing）

  当事件 B 可以沿着哈希指针找到事件 A，那么事件 B 就可见事件 A。两个 event 能通过哈希指针找到。

- 强可见（strongly seeing）

  当事件 B 能找到事件 A 的所有路径中跨越了绝对多数的节点 ,那么事件 B 强可见事件 A。白皮书中提到经过数学论证可以保证两个强可见的节点在虚拟投票时能获得一致的结果。简单理解就是事件 A 和 B 的发生次序（order）得到了全网节点的共识。

- 轮次（round）

  每个节点在同步到新 event 后，立即开始计算创建轮次。就是说所有

节点都听到点新的 gossip 了，好了，新的一轮听 gossip 传 gossip 游戏开始。

- 见证人（witness）

  每个节点在每个轮次中创建的第一个 event 就是见证人事件，即该轮次的祖先 event，节点可能在某个轮次中没有见证人事件，也就是说新的一轮游戏开始了，大家听到的第一个 gossip 就是这个 gossip 的见证人，这个 gossip 就是见证人 gossip。

- 知名见证人（famous witness）

  如果 R 轮的见证人能够被绝对多数的 R+1 轮的见证人可见，则它就是知名见证人。简单说就是某个 gossip 传到一定程度了，这一轮已经有很多节点都听说过这个 gossip 了，到下一轮的时候，达到了 2/3 以上节点都听说过这个 gossip 了，这一轮的这些节点都知名了，突破了临界点。

- 创建轮次（round created）

  一个事件的创建轮次是 R 或者 R+1，其中 R 是该事件父节点的最大轮次。当且仅当事件能强可见绝对多数的 R 轮见证人，则该事件的创建轮次为 R+1。这个定义一点都不好理解，简单来说，某个 gossip 在节点上传来传去，总得有个头才行，因为对一个没听过这个 gossip 的新节点而言，听到这个 gossip 的时候，自己签名加入 transaction 之后就是一个新的 gossip 了，整个 hashgraph 的目标是就这些 gossip 的发生次序在整个节点网络达成共识，同时实现异步 BFT。对于某个 gossip（等于 event，说成 gossip 更通俗易懂）

而言，节点从见证人到知名见证人的转变，就好比比特币区块链上确认数从 1 个到 6 个的过程，对于这个 gossip 的确认信心逐渐增强的过程，当知名见证人达到绝大多数的时候，这个 gossip 就被整个网络确认了，然后确认下一个 gossip，这样一个一个将不断新加入的 gossip 定好 order。

- 接受轮次（round received）

  如果 R 轮（创建轮次）中的所有知名见证人可见某一普通事件，则该事件的接受轮次就是 R 轮。可以简单理解，知名见证人就是那些信心满满的确认 gossip 的节点，为什么信心满满呢，就是大家都听说过这个 gossip 也没表示怀疑，那当然就可以信心满满。定义是复杂了点，简单说，在 R 轮次，知名见证人可见某一个 gossip，就说这个 gossip 的接收轮次是 R 轮。

  根据上面的定义和解释，再来看这些图，就简单清晰了。

  下面的图，很明显是自下向上不断生长，就像一棵 hash 树一样，gossip 不断在增长的网络中传播，新的 gossip 不断被确认。确认过程中，一个重要的概念是轮次，如何创建轮次，就是按照每个节点都听到新的 gossip 开始。其实这个轮次是逻辑意义上的，对于整个网络来说，才不管多少轮次，只要大部分节点都信心满满确认某个 gossip 的时候，这个 gossip 就板上钉钉了，紧接着是确认下一个 gossip 的事情了，这一茬就算过去了。

见证人，新的一轮游戏开始，大家听到的第一个 gossip 就是见证人事件，大家也顺便做个见证人。

不要以为让你做见证人就算了，还要看看你算不算知名见证人，这就是一个根据见证人多不多来衡量的。规范点说 ,要判断在这一轮，你这个节点是不是知名见证人，就要看下一轮的那些见证人是否绝大多数可见你这个见证人。这就是一个投票的过程。

如果绝大多数节点都投票说你这个节点是知名见证人，但统计票数和你见证的这个 gossip 成为板上钉钉还需要到下一轮才能见分晓。

计票阶段，一旦某个投票结果的计票数量超过绝对多数即认为该结果有效，也就是达成共识。根据数学理论证明，任何一个 R+2 轮见证人如果对投票结果做出了决定，那么这个结果就是全网的结论，如果这轮见证人无法做出决定，就由下一轮见证人计票决定，直到得出确切结论。

根据数学定理只要我们在每十轮增加一个随机轮次（ coin round ），则选举过程最终一定会结束（以概率 1 收敛，通俗点说就是几乎必然收敛，这是概率论中的概念）。在随机轮中，收集到绝对多数结果的见证人仅投票而不做决定，而其他见证人则根据数字签名的中间位进行随机投票。我们继续进行知名见证人的选举，结果如下：

一旦某个轮次确定了所有的知名见证人，就可以为这一轮次中的其他普通事件确定接受轮次和共识时间戳（ consensus timestamp ）。我们可以看到黑色事件可以被第二轮的所有知名见

证人可见，因此它的接受轮次就是 2。就是确定说某个节点上的 gossip 被多少轮接受了。

现在我们开始确定黑色事件的共识时间戳用于后续确定共识顺序，寻找 A 节点最早的事件 X，它既是 A2 的祖先也是黑色事件的儿子，同理寻找 B 节点的 Y 和 D 节点的 Z。然后将 XYZ 事件的时间戳依次排序并取中位数作为黑色节点的共识时间戳。然后我们继续确定其他节点的接受轮次。

现在我们确定了 10 个接受轮次为 2 的事件，我们将为其排序得到全网公认的顺序，即共识顺序。我们按照以下优先级进行排序：

- 接受轮次
- 共识时间戳
- 按事件签名和某随机数异或的结果排序，这个随机数通过该轮所有知名见证人的数字签名进行异或运算得到

总结：

- gossip 在节点上传播，听 gossip 传 gossip 游戏开始。
- 每个节点在同步到新 gossip 后，立即开始计算创建轮次。就是说所有节点都听到点新的 gossip 了，好了，新的一轮听 gossip 传 gossip 游戏开始。
- 节点成为 gossip 见证人
- 节点成为 gossip 知名见证人
- 确定 gossip 的接受轮次
- 投票选出被确认的 gossip

很明显，看完整个共识算法，单个节点需要保存全网数据。

Hashgraph 是一个有趣的共识协议，已被证明在许多环境中产生高吞吐量。Hashgraph 在其当前运营的许可设置内是快速、公平和安全的。但是，如果在公共环境中使用，将面临可能无法维持其安全性和性能。