区块链中的共识算法

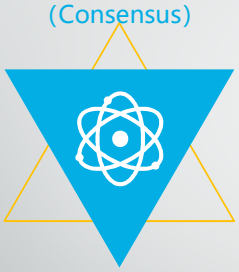
目录
CONTENTS

1 共识机制的相关知识
2 常见的共识算法
3 是否有完美的共识机制?
4 区块链分叉

1 共识机制的相关知识

3

共识
(Consensus)



- 共识机制

区块链作为一种按时间顺序存储数据的数据结构，可支持不同的共识机制。共识机制是区块链技术的重要组件。区块链共识机制的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：

1）一致性。所有诚实节点保存的区块链的前缀部分完全相同。

2）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。

共识机制

密码算法

4

共识机制的相关知识

CAP定律

CAP原则又称CAP定理，指的是在一个分布式系统中

Consistency（一致性） **Availability（可用性）**

Partition tolerance（分区容错性）

三者不可兼得。

分布式系统的CAP理论：理论首先把分布式系统中的三个特性进行了如下归纳：

一致性（C）：在分布式系统中的所有数据备份，在同一时刻是否同样的值。（等同于所有节点访问同一份最新的数据副本）

可用性（A）：在集群中一部分节点故障后，集群整体是否还能响应客户端的读写请求。（对数据更新具备高可用性）

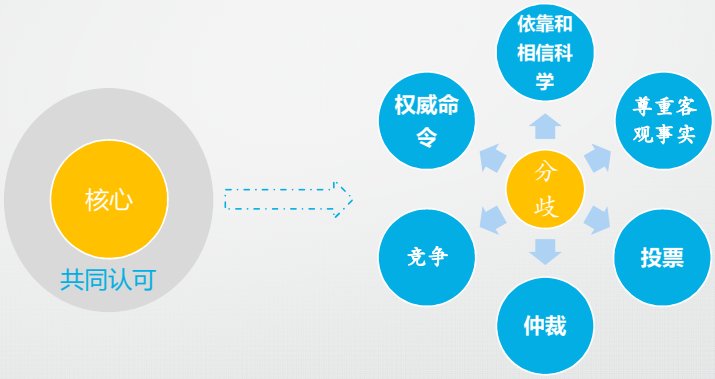
分区容错性（P）：以实际效果而言，分区相当于对通信的时限要求。系统如果不能在时限内达成数据一致性，就意味着发生了分区的情况，必须就当前操作在C和A之间做出选择。

5

共识机制的相关知识

几种生活中的共识机制

顾名思义，“共识”的核心是共同认可，它的反面就是有分歧。



6

共识机制的相关知识

- **从以下4个维度评价各共识机制的技术水平：**

1）安全性。即是否可以防止二次支付、自私挖矿等攻击，是否有良好的容错能力。

2）扩展性。即是否支持网络节点扩展。扩展性是区块链设计要考虑的关键因素之一。

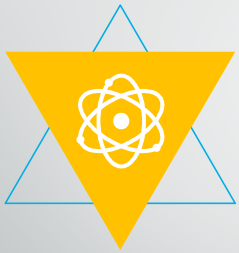
3）性能效率。目队交易达成共识被记录在区块链中至被最终确认的时间延迟，也可以理解为系统每秒可处理确认的交易数量。

4）资源消耗。即在达成共识的过程中，系统所要耗费的计算资源大小，包括CPU、内存等。区块链上的共识机制借助计算资源或者网络通信资源达成共识。

7

共识机制的相关知识

对共识机制的总结



达成共识越分散的过程，其效率就越低，但满意度越高，因此也越稳定；相反，达成共识越集中的过程，效率越高，也越容易出现独裁和腐败现象。

达成共识常用的一种方法就是通过物质上的激励以对某个事件达成共识；但是这种共识存在的问题就是容易被外界其它更大的物质激励所破坏。

8

2 常见的共识算法

9

常见的共识算法

非拜占庭错误：出现故障(crash 或 fail-stop，即不响应)但不会伪造信息

共识算法

- 针对非拜占庭错误的
  - 性能较高，但容错性较差，
    - Paxos
    - Raft
    - .....
- 针对拜占庭错误的
  - 往往容错性较高，但是性能相对较差
    - 工作量证明（POW）
    - 权益证明（POS）
    - 股份授权证明（DPOS）
    - 实用拜占庭容错算法（PBFT）
    - .....

拜占庭错误：伪造信息恶意响应的情况

10

常见的共识算法

- 处理拜占庭错误的算法有两种思路

思路

一种是通过提高作恶节点的成本以降低作恶节点出现的概率，如工作量证明、权益证明等，其中工作量证明是通过算力，而权益证明则是通过持有权益。

另外一种是在允许一定的作恶节点出现的前提下，依然使得各节点之间达成一致性，如实用拜占庭容错算法等。

11

常见的共识算法

POW
工作量证明机制

英文Proof of Work简称POW，简单的说就是可以证明你付出了多少工作量的证明。在比特币网络中，要想得到比特币就需要先利用自己服务器的算力抢夺记账权，等记账权抢到手之后，矿工还有个工作就是要把10分钟内发生的所有交易记录按照时间的顺序记录在账本上，然后同步给这个网络上的所有用户。矿工付出劳动抢记账权和记录交易，并且这个劳动也在全网得到大家的认可，达成了共识的机制。

Proof of Work（PoW）的机制来实现共识，该机制于 1998 年在 B-money 设计中提出。

使用PoW的知名区块链项目：

- 比特币
- 以太坊的前三个阶段（Frontier前沿、Homestead家园、Metropolis大都会）

常见的共识算法

POW 工作量证明机制

1）去中心化，将记账权公平的分派到其他节点。你能够获得的币的数量，取决于你挖矿贡献的有效工作。

2）安全性高，破坏系统需要投入极大的成本，如果想作弊，要有压倒大多数人的算力（51%攻击）。

优点 缺点

1）挖矿造成大量的资源浪费，目前bitcoin已经吸引全球大部分的算力，其它再用Pow共识机制的区块链应用很难获得相同的算力来保障自身的安全。

2）网络性能太低，需要等待多个确认，容易产生分叉，区块的确认共识达成的周期较长（10分钟），现在每秒交易量上限是7笔。

3）PoW共识算法算力集中化，慢慢的偏离了原来的去中心化轨道。从比特币扩容之争可以看到，算力高的大型矿池是主人，而持币的人没有参与决定的权利，比特币分叉出很多儿子，即将失去“去中心化”的标签。

常见的共识算法

POS
权益证明机制

英文Proof of Stake简称POS权益证明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即可。它将PoW中的算力改为系统权益，拥有权益越大则成为下一个记账人的概率越大。这种机制的优点是不像Pow那么费电。

一句话介绍：持有越多，获得越多。

权益证明（Proof of Stake，简称PoS）由Quantum Mechanic 2011年在比特币论坛讲座上首先提出，后经Peercoin（点点币）和NXT（未来币）以不同思路实现。

以太坊 2.0 采用权益证明机制（POS）

常见的共识算法

POS 权益证明机制

1）在一定程度上缩短了共识达成的时间。

2）不再需要大量消耗能源挖矿。

1）还是需要挖矿，本质上没有解决商业应用的痛点；

2）所有的确认都只是一个概率上的表达，而不是一个确定性的事情，理论上有可能存在其他攻击影响。例如，以太坊的DAO攻击事件造成以太坊硬分叉，而ETC由此事件出现，事实上证明了此次硬分叉的失败。

3）极端的情况下会带来中心化的结果。PoS机制由股东自己保证安全，工作原理是利益捆绑。在这个模式下，不持有 PoS 的人无法对 PoS 构成威胁。PoS 的安全取决于持有者，和其他任何因素无关。

常见的共识算法

DPOS
股份授权证明机制

Delegated Proof of Stake
DPoS在PoS的基础上，将记账人的角色专业化，先通过权益来选出记账人，然后记账人之间再轮流记账。这种方式依然没有解决最终性问题。类似于董事会投票，持币者投出一定数量的节点，代理他们进行验证和记账。

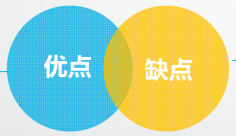
BitShares（比特股）社区首先提出了股份授权证明（简称DPoS）机制，它与PoS的主要区别在于节点选举若干代理人，由代理人验证和记账，但其合规监管、性能、资源消耗和容错性与PoS相似。类似于董事会投票，持币者投出一定数量的节点，进行代理验证和记账。

例子：BitShares，Steem，EOS

常见的共识算法

DPOS 股份授权证明机制

大幅缩小参与验证和记账节点的数量，可以达到秒级的共识验证。



整个共识机制还是依赖于代币，而很多商业应用是不需要代币的。

常见的共识算法

PBFT
拜占庭容错共识机制

PBFT是英文Practical Byzantine Fault Tolerance的缩写，意为实用拜占庭容错共识机制。PBFT是一种状态机副本复制算法，即服务作为状态机进行建模，状态机在分布式系统的不同节点进行副本复制。每个状态机的副本都保存了服务的状态，同时也实现了服务的操作。同所有的状态机副本复制技术一样，PBFT对每个副本节点提出了两个限定条件：

（1）所有节点必须是确定性的。也就是说，在给定状态和参数相同的情况下，操作执行的结果必须相同；

（2）所有节点必须从相同的状态开始执行。在这两个限定条件下，即使失效的副本节点存在，PBFT算法对所有非失效副本节点的请求执行总顺序达成一致，从而保证安全性。

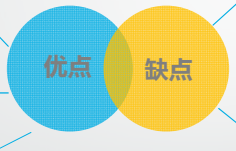
例如：Fabric

PBFT拜占庭容错共识机制

1）系统运转可以脱离币的存在，pbft算法共识各节点由业务的参与方或者监管方组成，安全性与稳定性由业务相关方保证。

2）共识的时延大约在2~5秒钟，基本达到商用实时处理的要求。

3）共识效率高，可满足高频交易量的需求。



1)当有1/3或以上记账人停止工作后，系统将无法提供服务；

2)当有1/3或以上记账人联合作恶，且其它所有的记账人被恰好分割为两个网络孤岛时，恶意记账人可以使系统出现分叉，但是会留下密码学证据

DBFT
授权拜占庭容错

delegated BFT 由权益来选出记账人，然后记账人之间通过拜占庭容错算法来达成共识。DBFT机制最核心的一点，就是最大限度地确保系统的最终性，使区块链能够适用于真正的金融应用场景。

2016年4月，小蚁公司发布共识算法白皮书，描述了一种通用共识机制——授权拜占庭容错，提出了一种改进的拜占庭容错算法，使其能够适用于区块链系统。

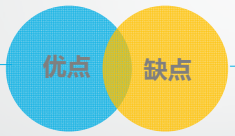
授权拜占庭容错算法在使用拜占庭容错算法的基础上，进行了以下改进：

- 将C/S架构的请求响应模式改进为适合P2P网络的对等节点模式；
- 将静态的共识参与节点改进为可动态进入、退出的共识参与节点；
- 为共识参与节点的产生设计了一套基于持有权益比例的投票机制，通过投票决定共识参与节点（记账节点）；
- 在区块链中引入数字证书，解决了投票中对记账节点真实身份的认证问题。

常见的共识算法

DBFT授权拜占庭容错

专业化的记账人；可以容忍任何类型的错误；记账由多人协同完成；每一个区块都有最终性，不会分叉；算法的可靠性有严格的数字证明。



当1/3及以上的记账人停止工作后，系统将无法提供服务；当1/3及以上的记账人联合作恶，且其他所有的记账人被恰好分割为两个网络孤岛时，恶意记账人可以使系统出现分叉，但是会留下密码学证据。

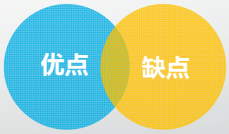
例子：NEO

常见的共识算法

PoI
重要度证明共识算法

PoI(Proof of Importance)重要度证明共识算法引了账户重要程度的概念，使用账户重要性评分来分配记账权的概率。

低能耗，速度快，公平



缺少社区共识，
账户重要性≠设备贡献度

常见的共识算法

POP算法

POP(Proof of Participation)算法，POP 将 POI和DPOS 的思想结合

基本原理

这是标准链（CZR）的创新，基于账户参与度的 PoP(Proof of Participation)算法，POP 将 POI 和DPOS 的思想结合，既能确保对设备的公平性，又拥有社区的共识。

优点 ← 低功耗、速度更快，更加安全，既能确保公平性，又拥有社区的共识。

常见的共识算法

PAXOS 共识机制

PAXOS是个分布式一致性协议，它的事件需要多个节点共同参与，一个事件完成是指多个节点上均完成了自身负责的单机子事件(把这样的事件称为"分布式事件")，这样的出某个节点上它们的单机子事件的先后，也不能根据某个节点上两个单机子事分布式事件可以看作是多个单机子事件的复合，但是即不能从两个分布式事件的先后推导件的先后断言它们对应的分布式事件的先后。

PAXOS算法是分布式技术大师Lamport提出的，主要目的是通过这个共识机制让参与分布式处理的每个参与者逐步达成一致意见，简单的说就是在一个选举过程中，让不同的选民最终做出一致性的决定。这个应用应该很受参与美国总统大选人选的欢迎吧。

常见的共识算法

PAXOS 共识机制

1998年Lamport提出Paxos算法，后续又增添多个改进版本的Paxos形成Paxos协议家族，且Paxos都有共同点是不容易工程实现。

Classic Paxos： LeaderLess，又名Basic Paxos，以下均为Paxos的变种，基于CAP定律，侧重了不同方向。

Cheap Paxos

Egalitarian Paxos： conflicts rare

Fast Paxos： Leader only when needed，conflicts common

Multi-Paxos： Leader driven

Byzanetine Paxos

常见的共识算法

RAFT 共识机制

Raft算法是对Paxos算法的一种简单实现，包括三个角色：leader，candiate，follower。

Raft机制可以理解成全网选出一个记账者leader，如果它稳定运行没有挂掉，就由这个节点记账，全网无条件接受他的记账结果，相信他是诚实的，假如这个节点挂掉了，那么大家是可以通过超时或网络探测感知，然后快速启动一轮投票，来从candiate选出一个新的记账节点，然后继续无条件的等它记账，这样就达成了容错性。

这在信任度较高，机构组成简单的的联盟链或者一个机构内的私有链里，是比PBFT更加高效的一种做法，因为不需要多步的反复确认，受网络影响的可能性也小很多。总的来说，Raft算法强调可用性和最终一致性，效率非常高，但是防欺诈性一般只能事后检查。

常见的共识算法

PoA
共识机制

权威证明（Proof of Authority，PoA）共识算法，节点只有被授权以后才能参与区块链共识。一旦被授权之后，共识节点享有公平的记账权利。因此，他们无需投入巨大的资源去竞争记账权。此外，记账权也和记账节点所拥有的数字资产无关。

验证者通过放置这个身份来获得担保网络的权利，从而换取区块奖励。
若是验证者在整个过程中有恶意行为，或与其他验证者勾结。那通过链上管理可以移除和替换恶意行为者。
现有的法律反欺诈保障会被用于整个网络的参与者免受验证者的恶意行为。

PoA算法出块效率高，不需要挖矿消耗大量资源，整个网络验证者互相监督，随时可以投票加入新的验证者或者剔出不合格验证者。

例子：唯链，TomoChain

常见的共识算法

DAG
(Directed Acycli Graph)

有向无环图DAG（Directed Acycli Graph）

，DAG共识算法的诞生是为了解决区块链的效率问题，通过DAG拓扑结构存储交易区块，支持网络中并行打包出块，提高交易容纳量。之后DAG不断演化逐渐形成了 blockless 的发展方向。从数据结构来看，DAG模式是一种典型的Gossip算法，即本质上为异步通讯，带来的最大的问题是一致性不可控，并且网络传输数据量会随着节点的增加而大幅增加。

DAG算法支持交易快速确认，低廉交易手续费，同时也剔除了矿工角色。但是目前来看安全性低于POW等机制，容易形成中心化，例如IOTA依赖validator，字节雪球则需要见证人节点。

例子：IOTA，Byteball（字节雪球）

常见的共识算法

RPCA

瑞波协议共识算法（Ripple Protocol Consensus Algorithm，RPCA）

瑞波协议共识算法使得一组节点能够基于由特殊信任节点达成共识。在瑞波网络中，每个服务节点都会维护一个信任节点列表且认为信任列表中的节点不会联合起来作弊。在共识过程中，各个需要共识的交易需要接受只接受来自信任节点列表中节点的投票，只有超过一定阈值后才能达成共识。瑞波协议共识算法比较高效，但是同样属于弱中心化且防攻击能力比较弱。

例子：Ripple

3 是否有完美的共识机制？

30

是否有完美的共识机制?

一个好的共识机制必然需要考虑以下几个问题

1.谁是区块链的实际拥有者？节点还是地址持有人，谁享有权益、担当风险、并应负责?

2.从技术层面考察共识机制是否具有足够的稳定性、一致性来对抗网络同步问题和恶意节点。

3.是否需要耗费资源维护共识机制？如需要，应以何种方式补偿资源消耗者的付出?

4.共识机制与治理机制。当实际拥有者让度部分权力的时候，例如记账权利，除补偿执行人的资源消耗外，是否还应同时让度其他权利，例如投票权？如何保证实际拥有者始终能保持对区块链的控制权?

31

是否有完美的共识机制?

不同类型的区块链的共识机制

私有链

封闭生态的存储网络，所有节点都是可信任的。
由于私有链是封闭生态的存储网络，也就是说内部节点都是封闭系统自行创建的，并不存在内部节点的信任问题，所以根本不存在所谓拜占庭将军问题。针对私有链，只需要做到少数服从多数原则即可并不需要使用任何共识机制来达成一致。

联盟链

半封闭生态的交易网络，存在对等的不信任节点。

公有链

开放生态的交易网络，这层主要是为行业链和私有链提供全球交易网络。

公有链而言，由于所有节点都是平等的，也就是完全去中心化的组织，笔者认为使用工作量证明机制至少目前来讲是最公平也最客观的，因为它完全依赖于随机计算结果与其他任何主观因素都没有联系。

32

4 区块链分叉

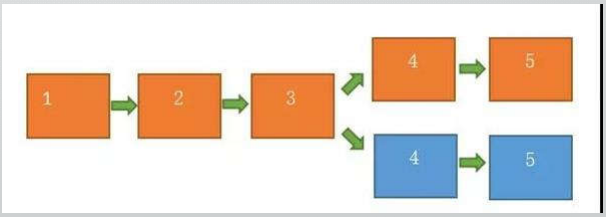
33

软分叉与硬分叉

什么是区块链分叉

由于每个矿工的区块数据都不一样，所以他们解题得出的结果也是不一样的，都是正确答案，只是区块不同。于是，区块链在这个时刻，出现了两个都满足要求的不同区块。那么，全体矿工这时该怎么办呢?

由于距离远近，不同的矿工看到这两个区块是有先后顺序的。通常情况下，矿工们会把自己先看到的区块复制过来，然后接着在这个区块开始新的挖矿工作。于是，出现了这样的情景：



34

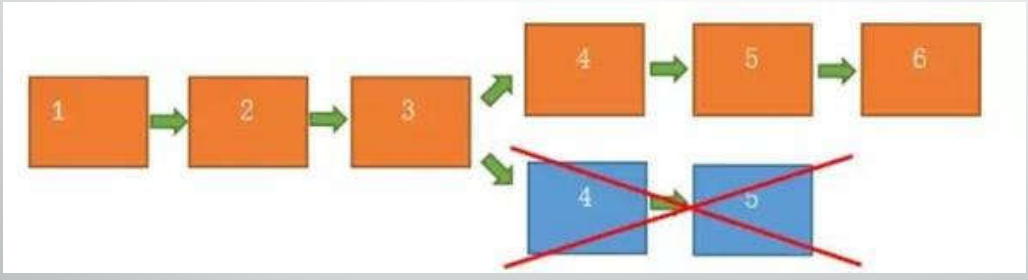
区块链分叉

- 遵守共同协议下偶然产生的分叉会自然消除
- 由于协议不同而引起的分叉
  - 硬分叉
  - 软分叉

35

遵守共同协议下偶然产生的分叉会自然消除

当矿工发现全网有一条更长的链时，他就会抛弃他当前的链，把新的更长的链全部复制回来，在这条链的基础上继续挖矿。所有矿工都这样操作，这条链就成为了主链，分叉出来被抛弃掉的链就消失了。



36

遵守共同协议下偶然产生的分叉会自然消除

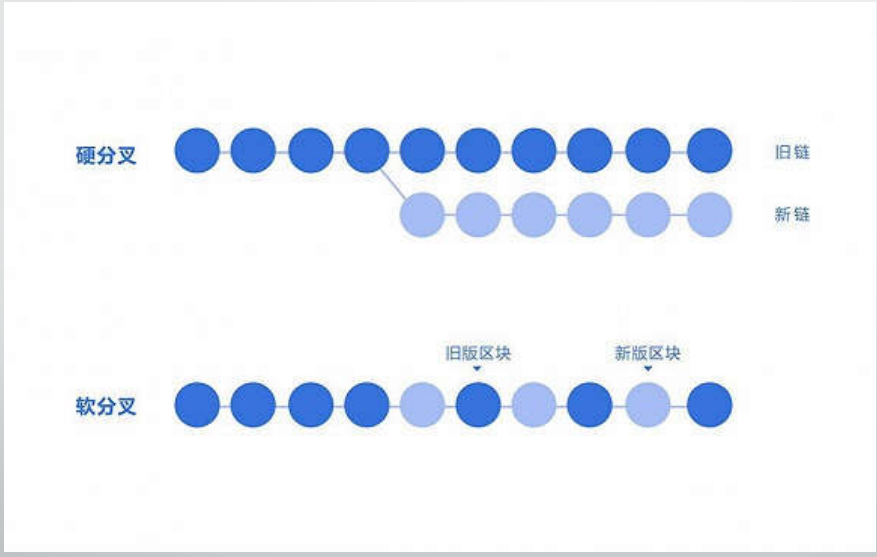
比特币的六次确认

承载比特币应用的区块链，一般认为一个区块后面再链了6个区块后，就不可能被“颠覆”了，故称为“六次确认”。

挖到一个新区块别高兴太早，等后面链了6个其他区块的时候，再庆祝吧。
承载比特币应用的区块链，平均10分钟生成一个区块，“六次确认”大概需要经历1个小时。

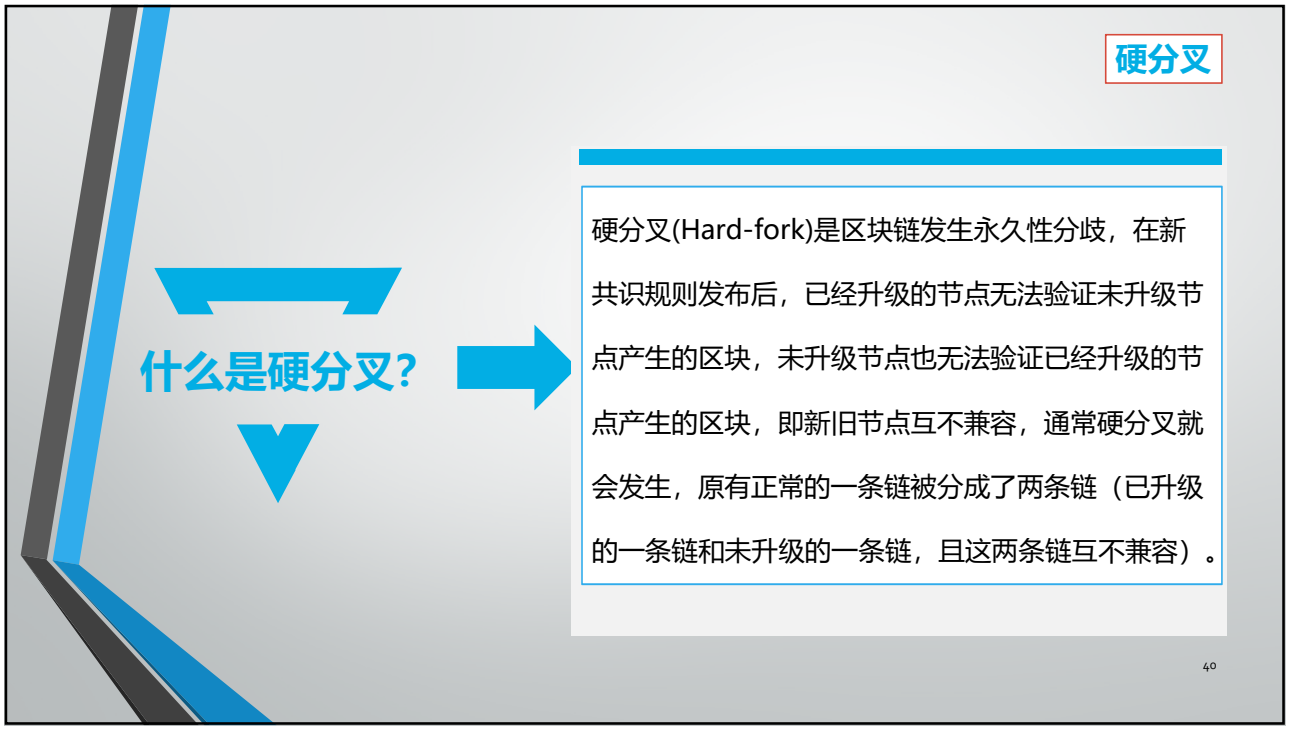
37

由于协议不同而引起的分叉–硬分叉和软分叉



38


什么是硬分叉和软分叉
未升级节点
原链
遵循旧规则
币A
硬分叉
升级后节点
分叉后的链
遵循新规则
币B
多数共识
少数（或没有）共识
原链
遵循旧规则
当所有矿工和节点升级时，原始链条“消失”，因为没有块再产生
软分叉
升级后节点
分叉后的链
遵循新规则
币A
39


硬分叉
什么是硬分叉?
硬分叉(Hard-fork)是区块链发生永久性分歧，在新共识规则发布后，已经升级的节点无法验证未升级节点产生的区块，未升级节点也无法验证已经升级的节点产生的区块，即新旧节点互不兼容，通常硬分叉就会发生，原有正常的一条链被分成了两条链（已升级的一条链和未升级的一条链，且这两条链互不兼容）。
40

著名的硬分叉事件1
-- The DAO 事件（ETH的产生）

DAO 是Decentralized Autonomous Organization（分布式自治组织）的简称，TheDAO是当时世界上基于以太坊区块链平台最大的众筹项目。其目的是让持有TheDAO代币的参与者通过投票的方式共同决定被投资项目，整个社区完全自制，并且通过代码编写的智能合来实现。于2016年5月28日完成众筹，共募集1150万以太币，在当时的价值达到1.49亿美元。

TheDAO事件是指2016年6月17日，DAO因为编写智能合约时不谨慎，存在漏洞，而黑客则利用该漏洞盗取了DAO，造成360多万个ETH被盗，是当时流通的以太币总额的15%。，按照当时的以太币价格，损失达到了6000万美元。

为了弥补这个损失，2016 年 7 月，以太坊团队修改了以太坊合约代码实行硬分叉，强行把The DAO 及其子 DAO 的所有资金全部转到一个特定的退款合约地址，进而“夺回”了黑客所控制 DAO 合约上的币。但这个修改被一部分矿工所拒绝，因而形成了两条链：

- **一条为原链(以太坊经典，ETC)**
- **一条为新的分叉链(ETH)**

他们各自代表了不同社区的共识和价值观。

41

著名的硬分叉事件2
--以太坊拜占庭与大都会君士坦丁堡硬分叉升级

- **2017年10月16日以太坊从第4370000号区块起顺利完成了代号为拜占庭的硬分叉。**
- **2019年2月28日以太坊完成了大都会君士坦丁堡硬叉更新完成。**
- **并不产生新的币种，只是升级客户端**

42

著名的硬分叉事件3
--比特币分叉产生众多衍生竞争币

硬分叉与IFO

硬分叉这种创造货币的方式和ICO非常类似，于是一个新的名词诞生了——**IFO（Initial Fork Offerings）**。矿工团队在创造分叉的同时，可以在分叉发生的区块中，利用自己的特权，分配一些货币给自己或其他人（直接写成CoinBase交易即可），然后再开放让所有人都可以参与挖矿。

随着越来越多的硬分叉发生，比特币的公信力是否还能像以前一样？
IFO会不会成为比特币的杀手？
还需要时间的考验。
（对其他币也面临同样的问题）

硬分叉解读

（1）区块头里能记录版本信息，所以理论上任何人都可以改程序，升级版本自己玩，只是没有人陪你玩罢了，没人陪你玩，你挖到的矿大家不认可，就只是浪费电。还是那句话，区块链的世界里，遵守规则才能让矿工的利益最大化。

（2）硬分叉，其实违背了区块链“不能修改”的技术本质，采用了人为手段“强制回滚”，楼主认为，这违背了区块链去中心化的技术本质。

45

什么是软分叉?

软分叉是指：协议发生了一些变化，但旧节点却不能发现这个协议的变化，从而继续接受新节点用新协议所挖出的区块。旧节点矿工将可能在他们不能完全理解和验证的新区块上继续添加区块。

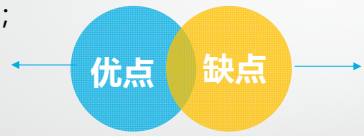
软分叉产生的原因：新的节点要求比老的节点要严格很多。

46

软分叉和硬分叉相比

软分叉总是只有一条链，没有分成两条链的风险；

软分叉不要求所有节点同一时间升级，允许逐步升级，且并不影响软分叉过程中的系统稳定性和有效性；



软分叉的前提是老的节点总是能够接受新节点的区块，这就要求把系统设计成向前兼容（forward compatible）；

软分叉总是建立在对老节点进行欺骗的基础上，它让老节点没有察觉实际上已经发生的变化，某种程度上违背了单点完整验证的原则

47

利益的驱动和博弈

尽管区块链基于一套严格的制度和规则，更强调“去中心化”和“去信任”，无需监管，但仍然有漏洞和bug出现。

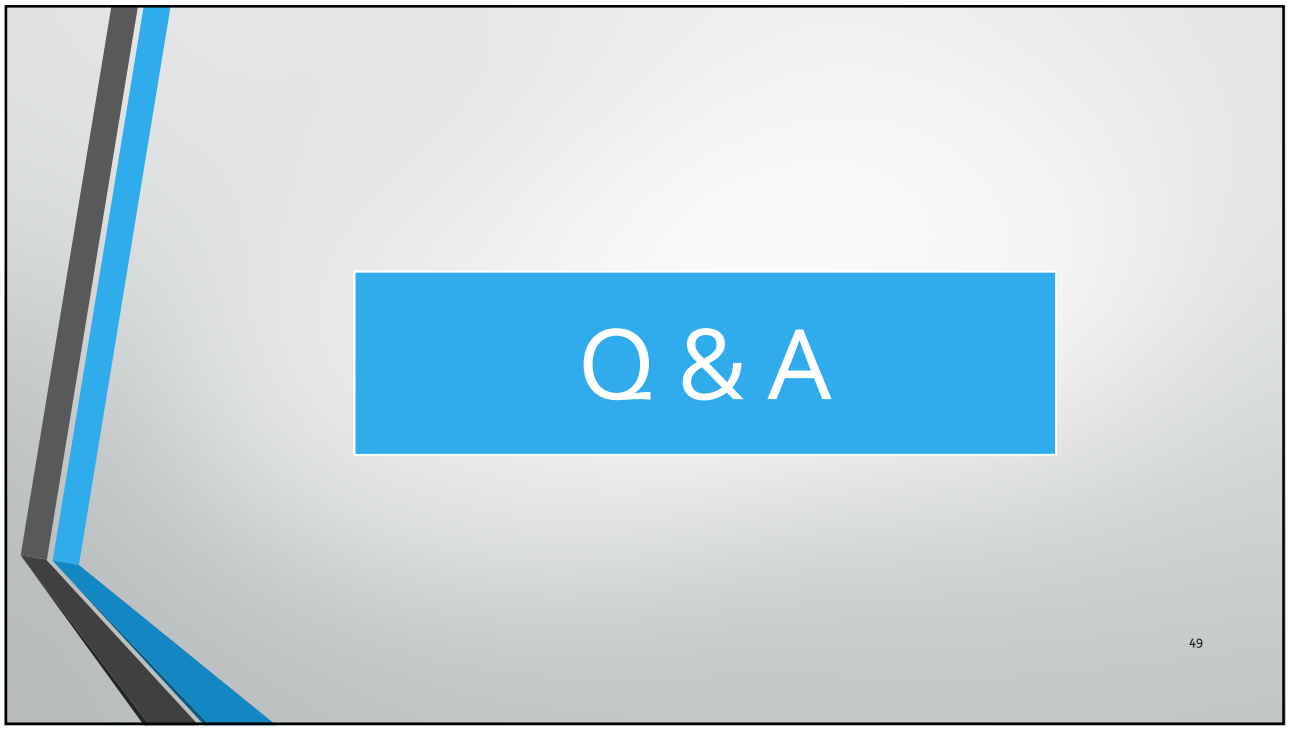
究其原因，就在于利益的驱动，也就是一部分矿工为了利益势必想尽办法，不择手段。

为了获得新区块以及奖励，在不满原有制度的情况下，反对者就会选择另谋出路、分道扬镳。

但要开辟新的道路，矿工们先要与平台这一管理者进行博弈，胜出后才能真正走自己的路。

综上所述，尽管区块链一直标榜公正、中立、自动等特点，但在私欲的驱动下，仍然围绕利益纠缠不清，犹如现实中的名利场。

48


Q & A
49