中山大學
SUN YAT-SEN UNIVERSITY

区块链的基础知识

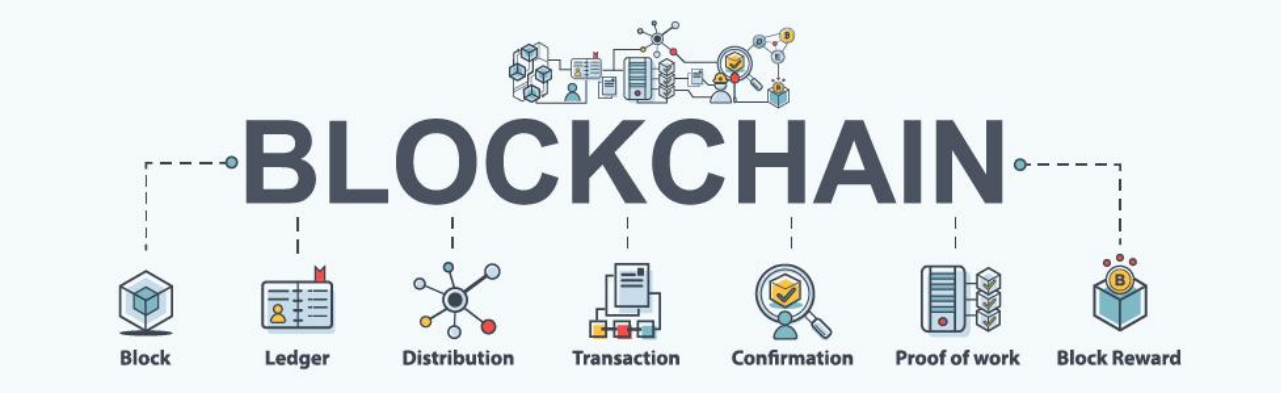
区块链为何能成为 Web3 与 元宇宙的基础设施

吴嘉婧
中山大学 软件工程学院
2024年 11 月 19日

提纲

区块链技术简介

- 区块链是什么?
- 区块链的基础知识



导言：人们在追求「去中心化」，到底在追求什么？

- 「中心化」有什么坏处?
  - 21年4月19日特斯拉用户维权事件
  - 22年河南村镇银行事件

新闻报道《四女弃驰女车主维权后续：厂家降低换车门槛》）

一名行业专家曾告诉财新记者，车主维权事件的根本原因是中国缺乏第三方争议处理机构，消费者遇到问题只能在汽车厂家、经销商和政府部门之间辗转，问题处理流程漫长也并不透明，容易让消费者产生愤怒情绪，进而采取极端方式维权。这名专家建议借鉴国外经验，成立第三方机构。

上海车展维权女子被行拘 特斯拉称对不合理诉求不妥协

汽车　2021/4/20 08:26 发布　2021/4/20 10:28 更新

上海市公安局青浦分局通报称，消费者应以合法合理途径表达诉求维护权益

4月20日，上海警方通报特斯拉车展事件。张某因扰乱公共秩序被处以行政拘留五日，李某因扰乱公共秩序被处以行政警告。图/截自上海市公安局官方微博

【财新网】（记者 刘雨锟）上海车展特斯拉维权车主因扰乱公共秩序被处以行政拘留

请朋友免费读财新

提纲

- 2个灵魂之问
  - Q1: 区块链是什么？（第3课）
  - Q2: 区块链技术会给世界带来什么的影响或改变？（第4课）

问题1：区块链是什么？

- 1.1 区块链背景与现状
- 1.2 区块链基本概念
- 1.3 区块链技术原理

问题1：区块链是什么？

- 1.1 区块链背景与现状
- 1.2 区块链基本概念
- 1.3 区块链技术原理

区块链近几年国家规划

- 从2016年开始进入国家规划
- 多地地方政府，采用多种方式引导和支持区块链技术和产业的发展

2019年10月25日，中央政治局第18次集体学习

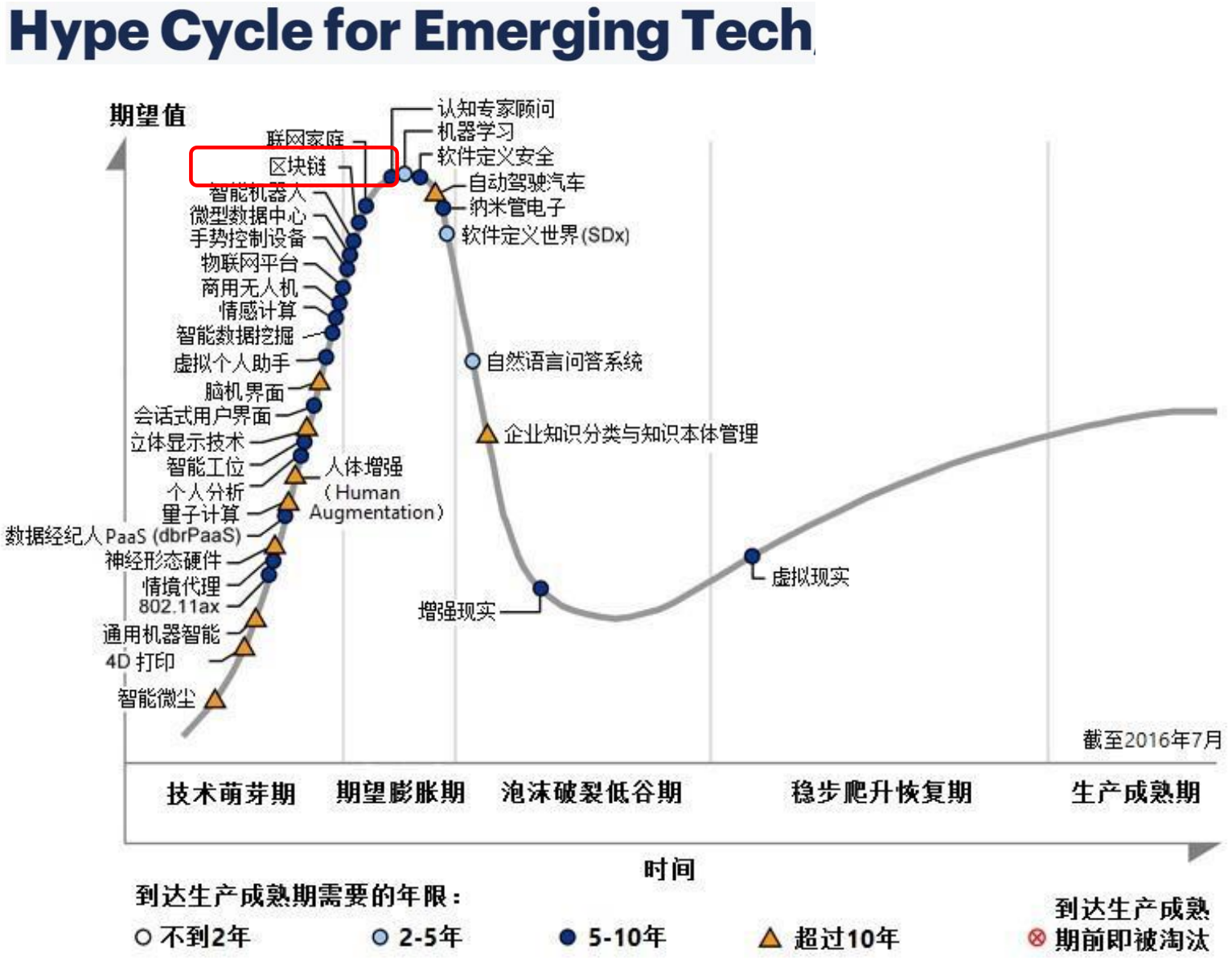
www.news.cn 新华网 NEWS www.xinhuanet.com
让新闻离你更近

习近平在中央政治局第十八次集体学习时强调 把区块链作为核心技术自主创新重要突破口 加快推动区块链技术和产业创新发展

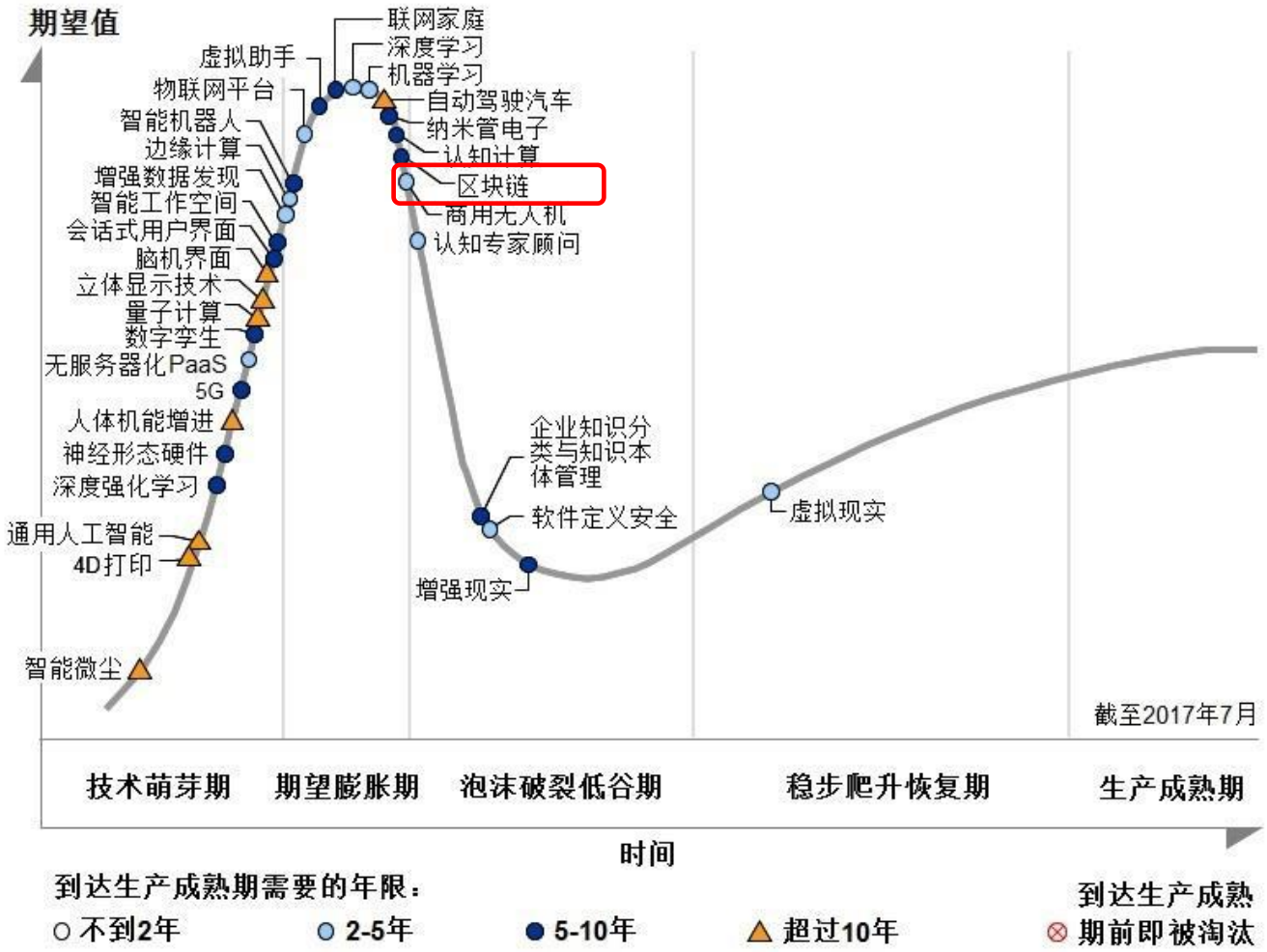
2019-10-25 18:14:26 来源：新华网

Gartner 新兴技术成熟度曲线-2016

- Gartner 新兴科技 技术成熟度曲线
- 该曲线重点关注了那些有望在未来五到十年内拥有高度竞争优势的科技

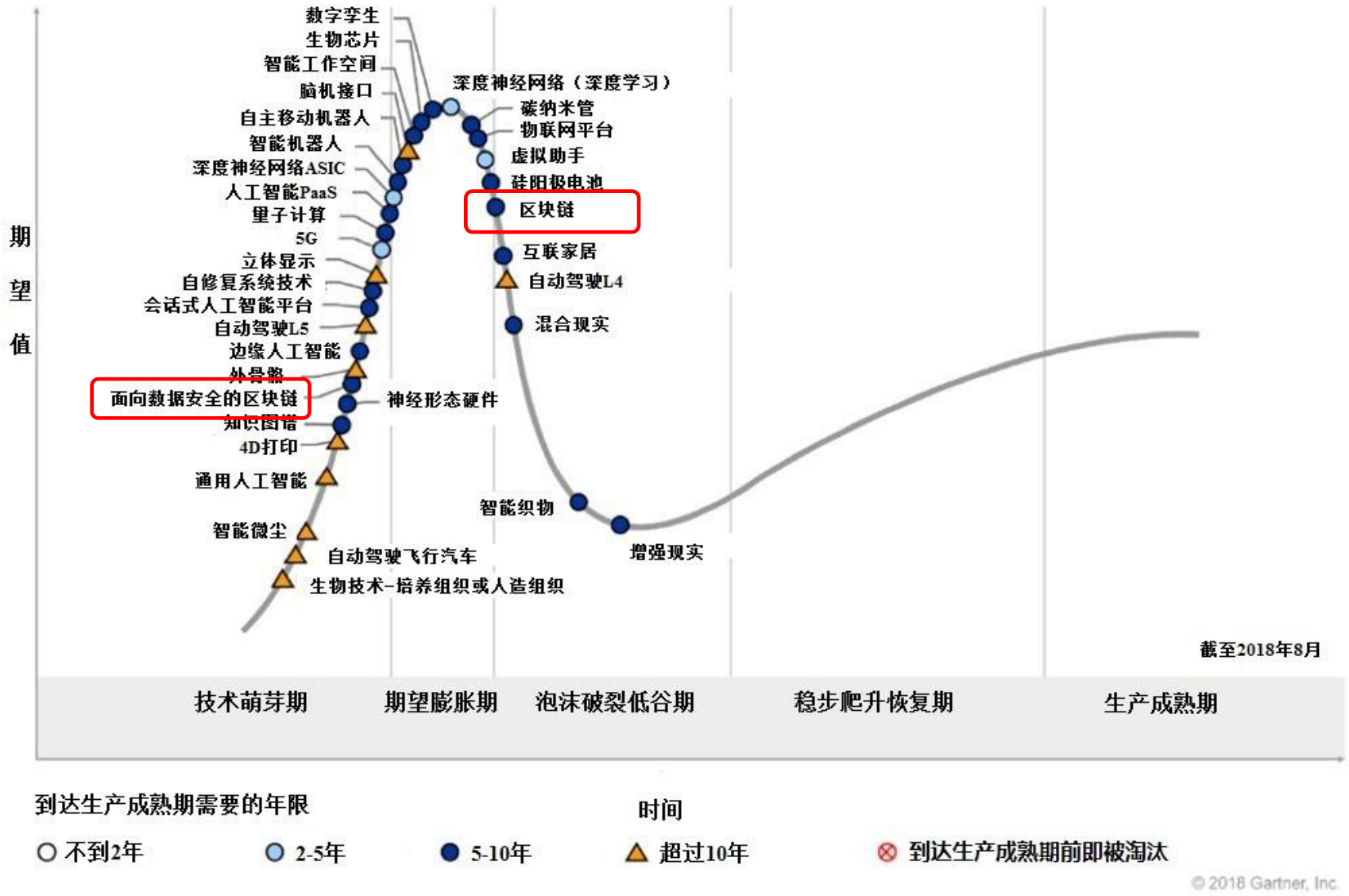


Gartner 新兴技术成熟度曲线-2017

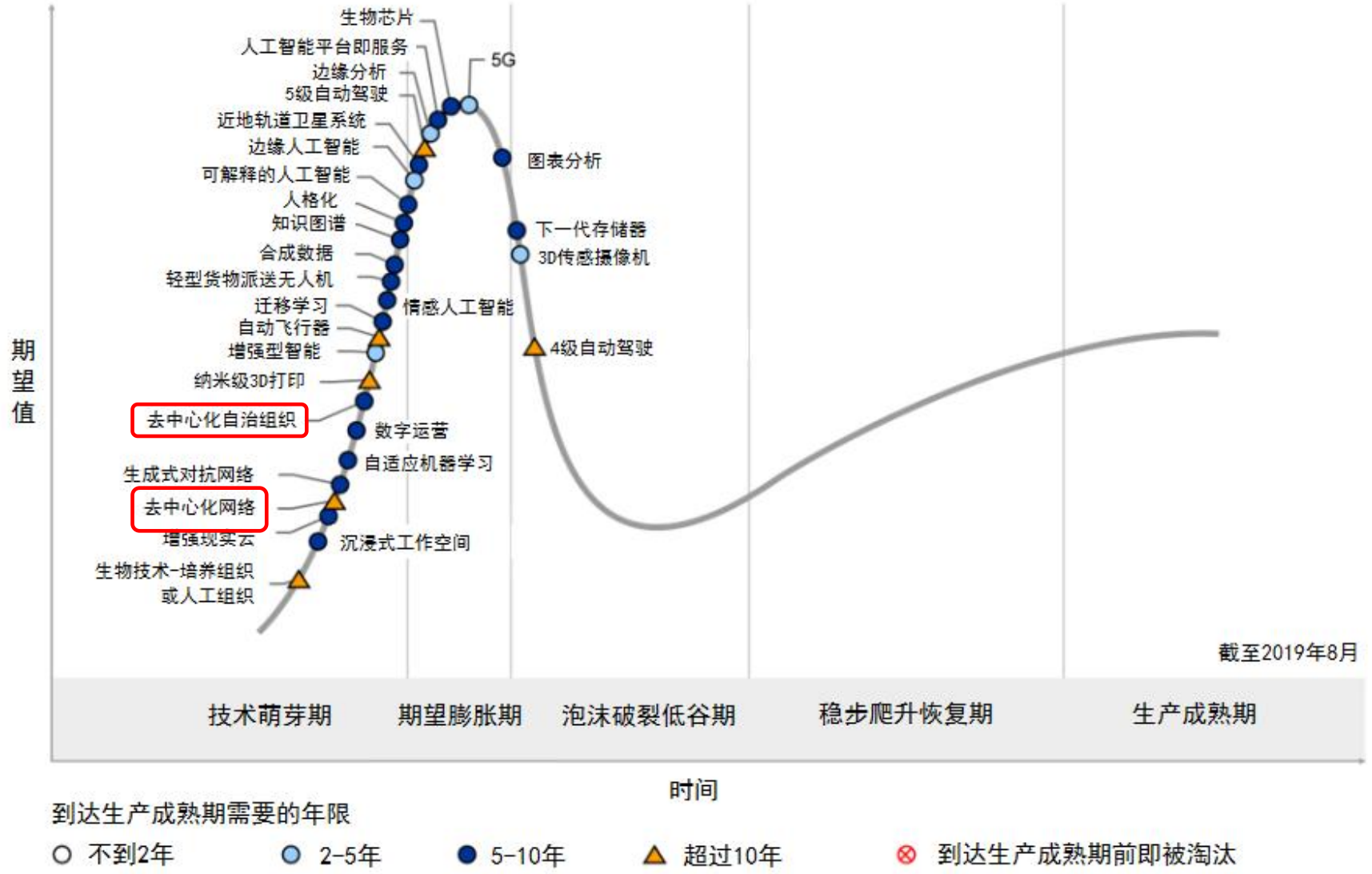


来源：Gartner（2017年7月）

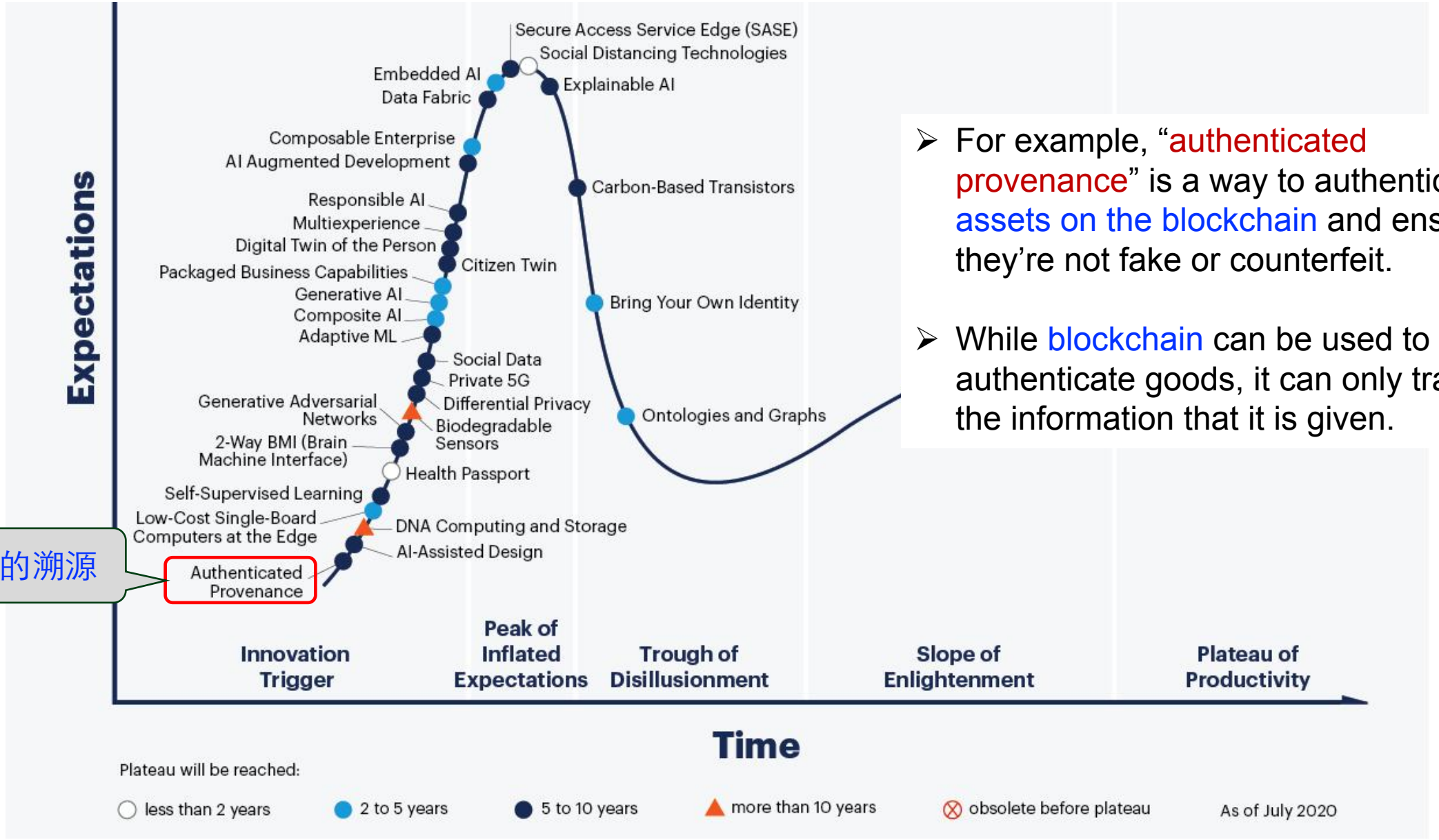
Gartner 新兴技术成熟度曲线-2018



Gartner 新兴技术成熟度曲线-2019



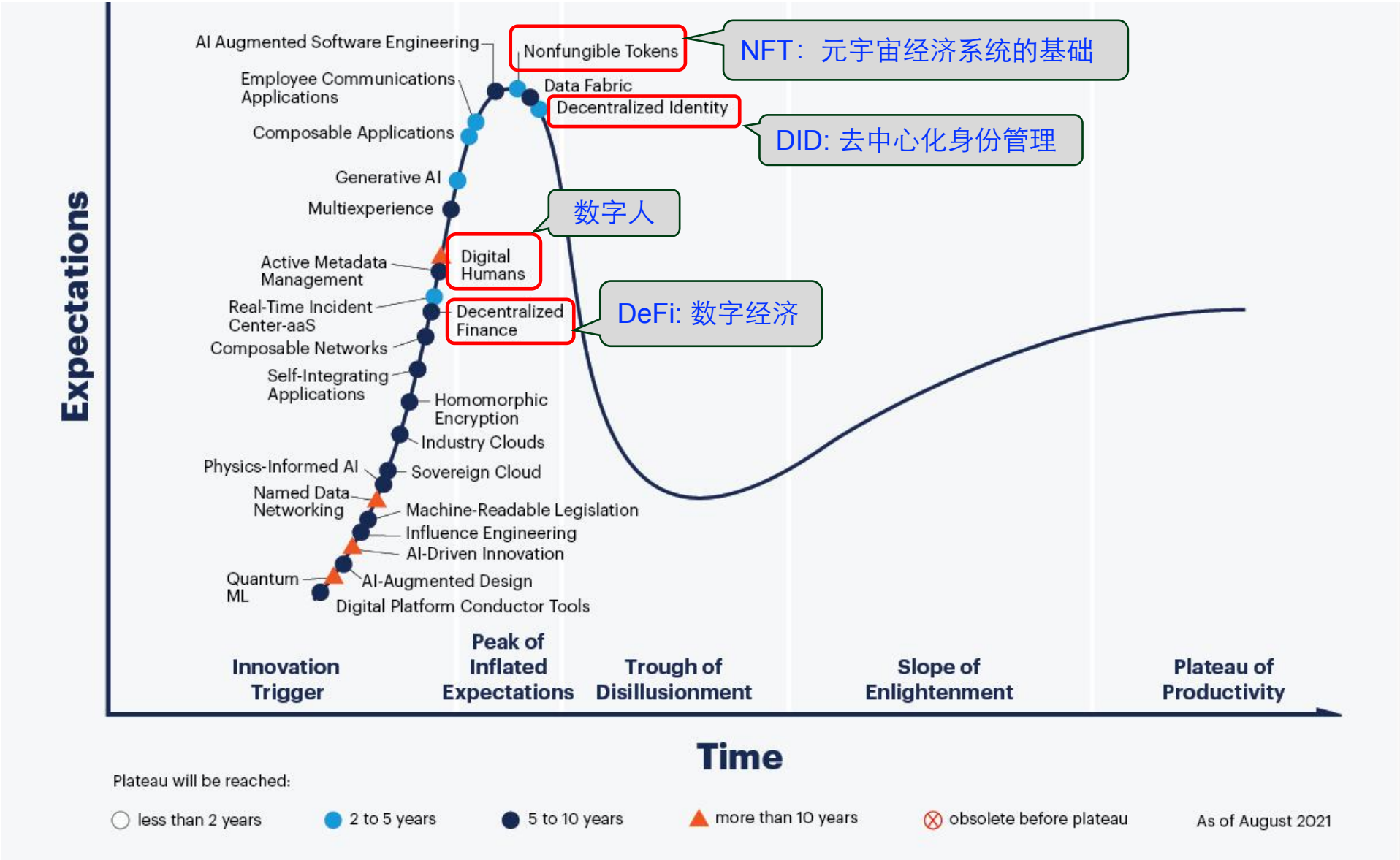
Gartner 新兴技术成熟度曲线-2020



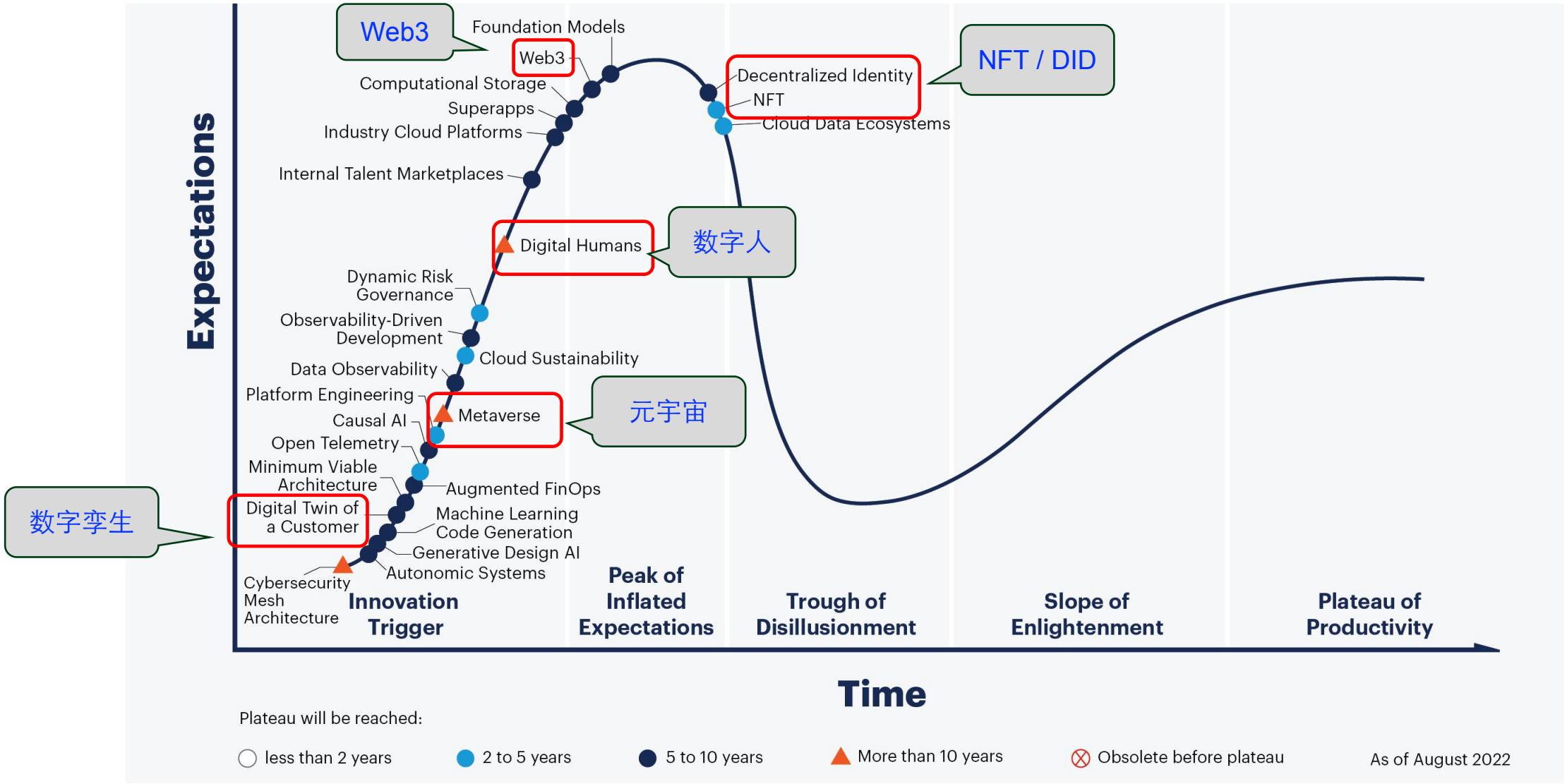
可认证的溯源

- For example, “authenticated provenance” is a way to authenticate assets on the blockchain and ensure they’re not fake or counterfeit.
- While blockchain can be used to authenticate goods, it can only track the information that it is given.

Gartner 新兴技术成熟度曲线-2021



Gartner 新兴技术成熟度曲线-2022



问题1：区块链是什么？

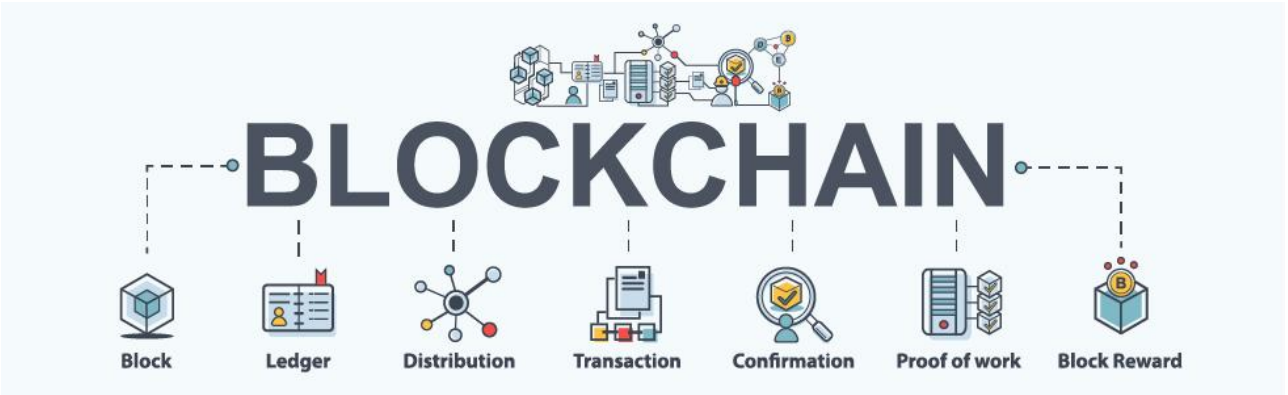
- 1.1 区块链背景与现状

- 1.2 区块链基本概念

- 1.3 区块链技术原理

以比特币为例

- 为什么要从比特币讲起?
- 跟区块链什么关系?

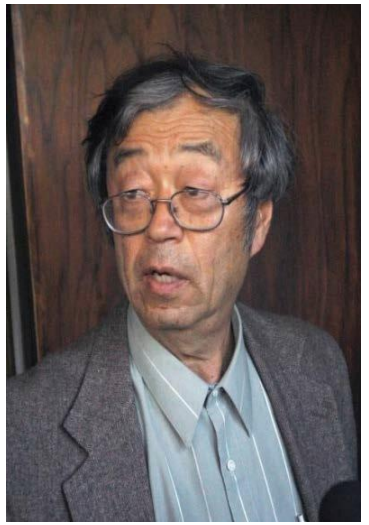


比特币诞生

Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending.



Not me !

Dorian Satoshi Nakamoto
(not actually Satoshi Nakamoto)

He graduated in physics from California Polytechnic and worked on classified defense projects.

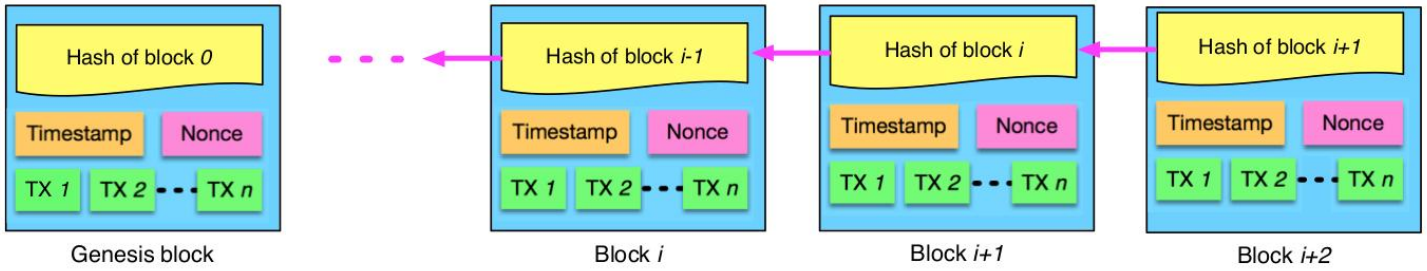
- 2009年1月，创世区块诞生。
- 一种完全基于点对点（P2P）的电子现金系统， 使得全部支付都可以由交易双方直接进行，完全 摆脱了第三方，创造了一种全新的货币体系。

  - 区块链技术随着比特币的出现而面世
  - 区块链 是 技术载体，比特币 是 产品

比特币的底层技术 —— 区块链初识

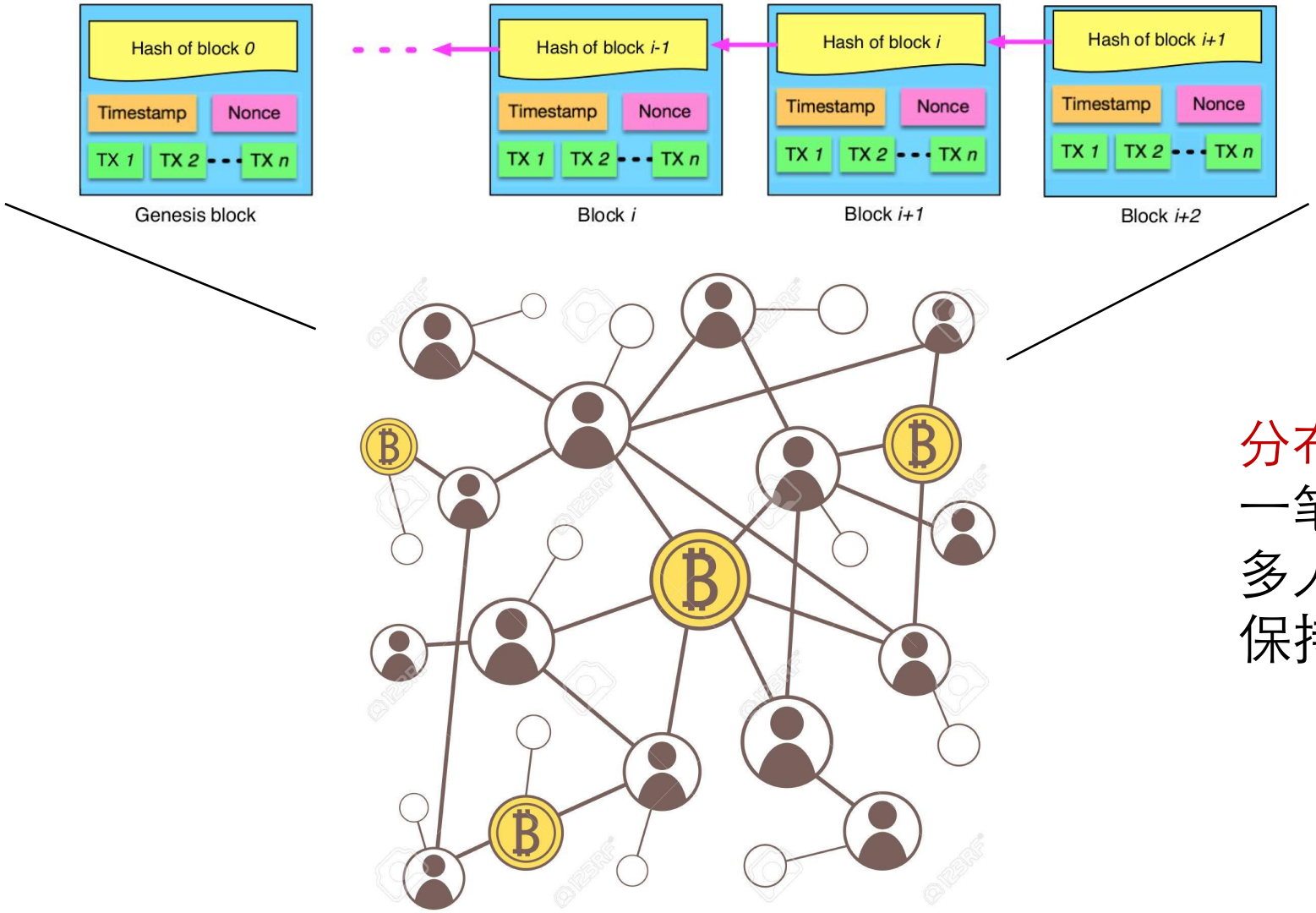
❑ 区块链 的 定义

- 按照时间顺序将数据区块以顺序相连的方式组合成的一种链式数据结构



- ❑ 区块链是一个**分布式的账本数据库**
- ❑ 网络中的每个节点都有一本完整的账本
- ❑ 链上数据无法篡改
- ❑ 去中心化，降低成本，提高效率

比特币 的 区块链



分布式账本：
一笔数据，
多人记录，
保持同步

问题1：区块链是什么？

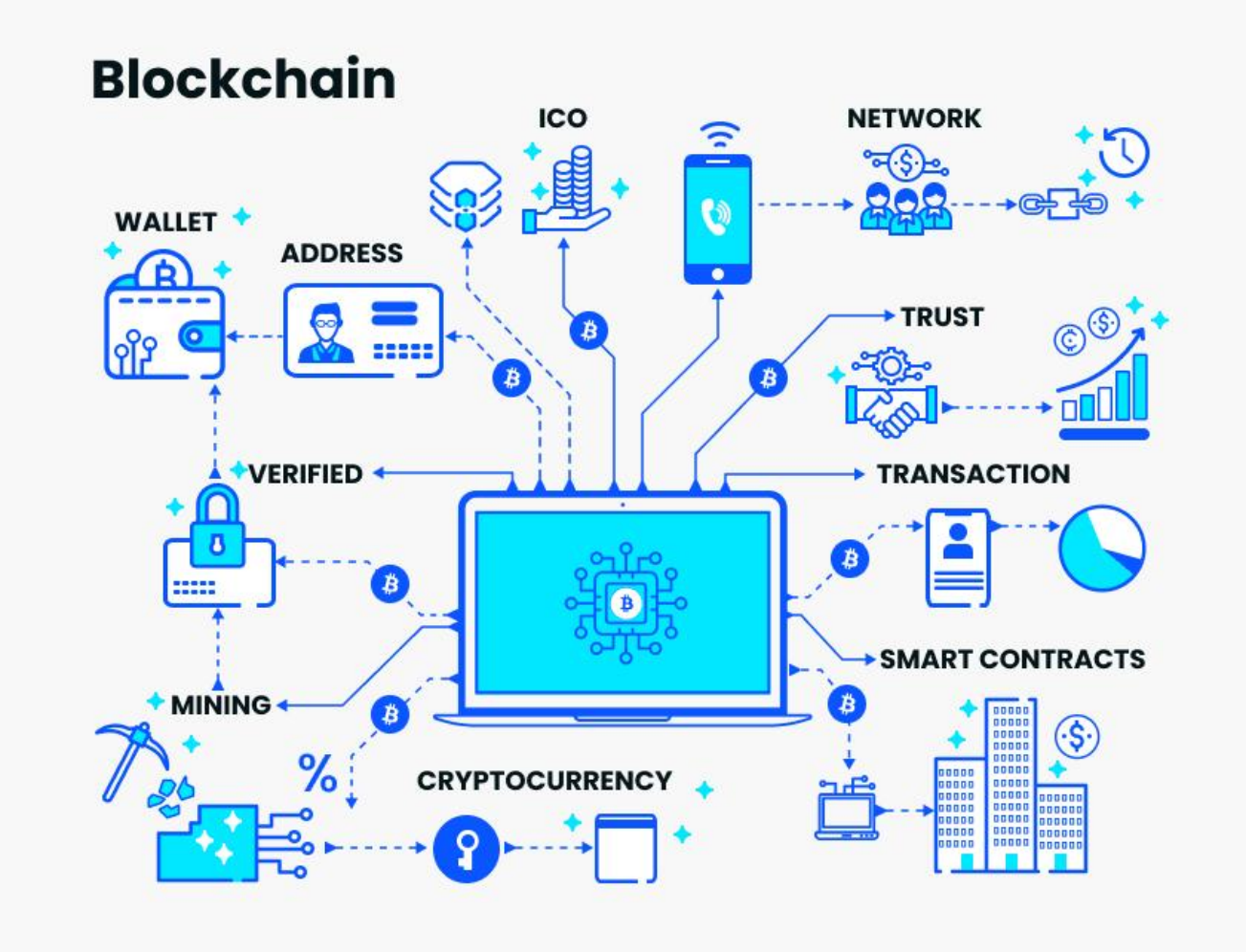
- 1.1 区块链背景与现状
- 1.2 区块链基本概念

- 1.3 区块链技术原理

导言：区块链特性

Anonymous
匿名性

Decentralized
分布式



Consensus
一致性

Immutable
不可篡改

导言：区块链技术的核心问题

- 如何确保“链上数据”不可篡改?
- 如何在 分布式 / 恶意攻击者存在 的环境中对账本状态达成 共识?

1.3 区块链的技术原理 —— 以比特币为例

- 以 比特币 为代表的 区块链技术，原理 主要包括
  - 1.3.1 密码学基础
  - 1.3.2 区块链数据结构
  - 1.3.3 （比特币的）共识机制

1.3 区块链的技术原理 —— 以比特币为例

- 以 比特币 为代表的 区块链技术，原理主要包括

- 1.3.1 密码学基础
- 1.3.2 区块链数据结构
- 1.3.3 （比特币的）共识机制

引言 —— 什么是密码学

著名的密码学者Ron Rivest解释道：

“密码学是研究如何在敌人存在的环境中通讯”

甲

乙

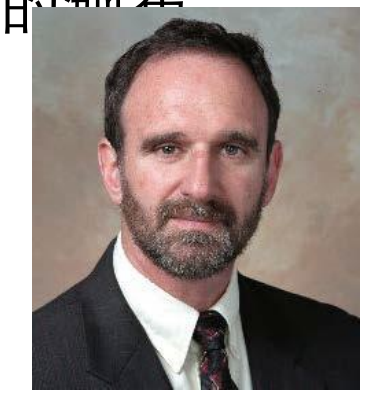
丙

密码学悠久历史

- 现代密码学（1976/1977年）
- 代表性事件
  - 公钥密码学的提[illegible]数据加密标准DES的颁布
- 意义：
  - 密码学成为了一门科学，研究从军事和外交走向了公开

左为Hellman，右为Diffie

Whitfield Diffie

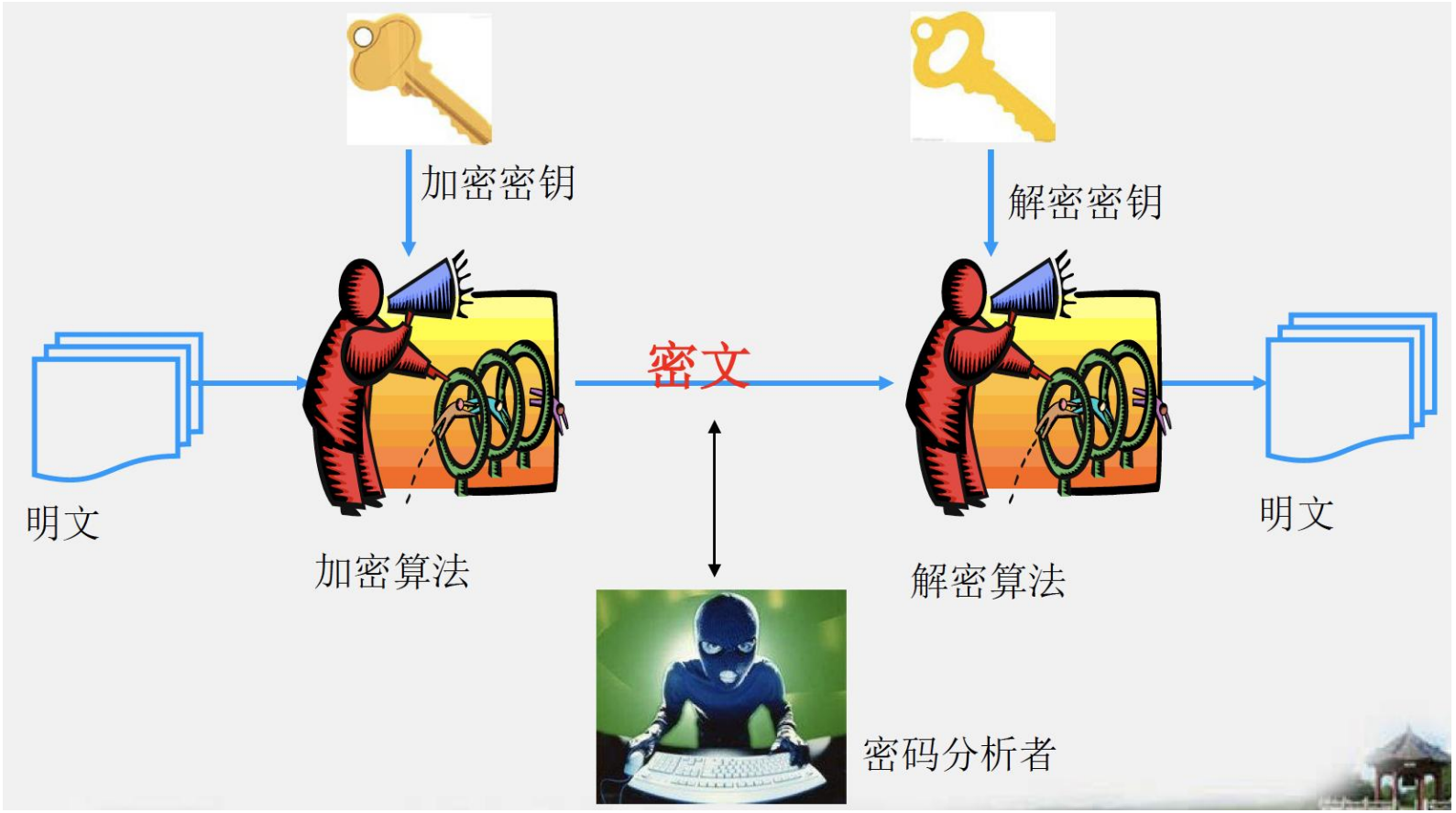


Martin Hellman

密码学主要研究内容

- 公钥加密
- 数字签名
- 私钥加密：分组密码，流密码
- Hash函数
- 伪随机数
- 安全协议：承诺，
- 零知识证明，
- 多方计算等

引言 —— 基本概念

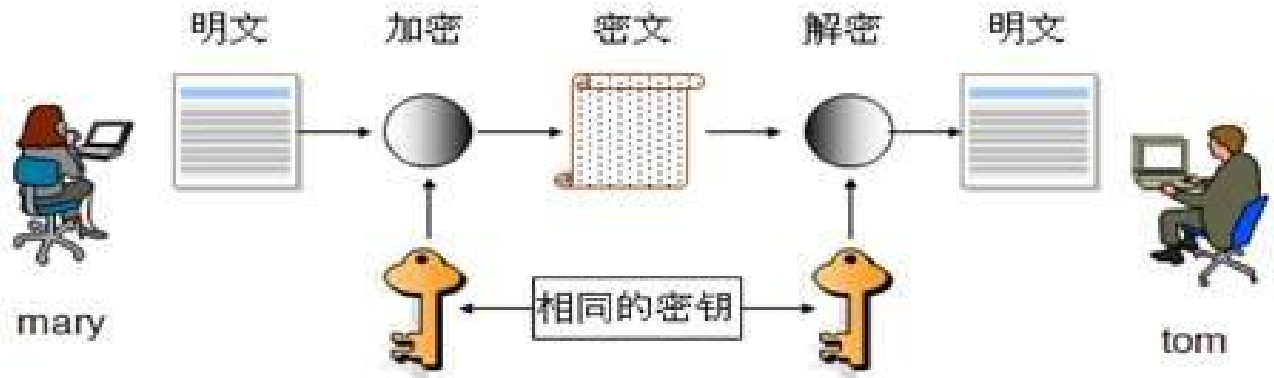


引言 —— 密码学基础

对称加密

使用相同的密钥

加密大量数据



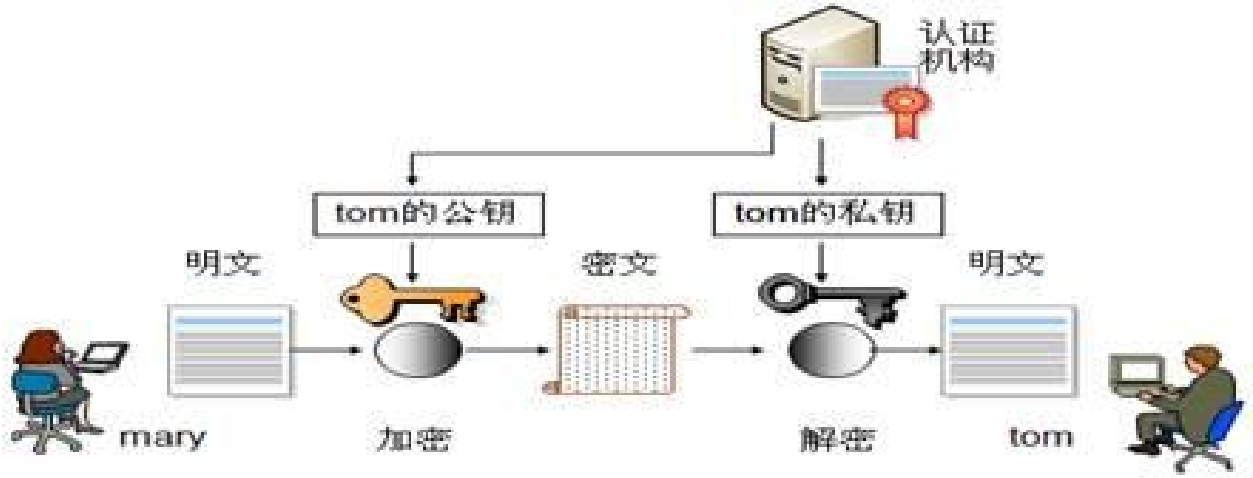
非对称加密

采用不同的密钥

加密少量数据

用于交换对称密钥

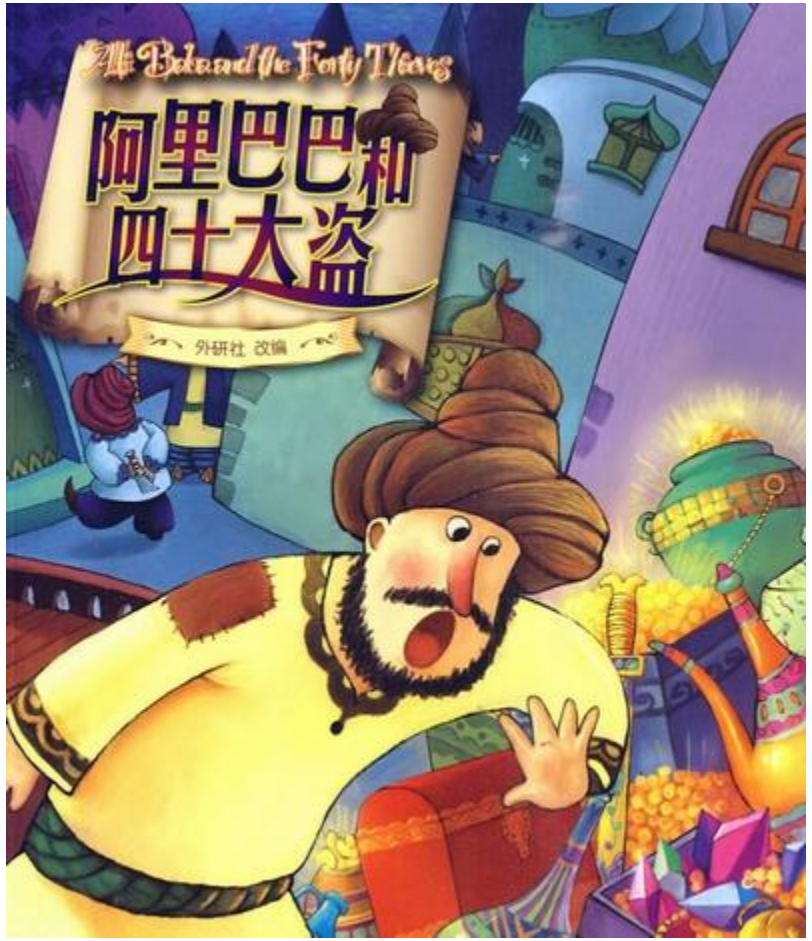
用于签名验签



引言 —— 博大精深的密码学算法

- 零知识证明Zero Knowledge
  - S.Goldwasser、S.Micali及C.Rackoff在20世纪80年代初提出
  - 证明者能够在不向验证者提供任何有用的信息的情况下，使验证者相信某个论断是正确的。
  - Zcash零币：比其他加密货币更强的隐私性。

零知识证明：阿里巴巴和四十大盗



- 阿里巴巴的零知识证明
  - 阿里巴巴想要向强盗证明自己拥有密码，又不想把密码告诉强盗。
  - 让强盗离开一箭之地，距离足够远听不到口令，足够近无法在龚剑下逃生。
  - 通过看强盗的手势展示开关门
  - 零知识（不提供石门口令）证明（阿里巴巴知道石门打开的方法）

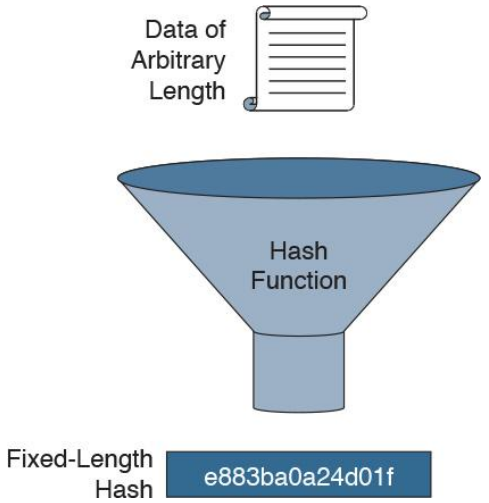
--1.3.1 密码学基础

PART 1: 哈希函数

PART 2: 数字签名

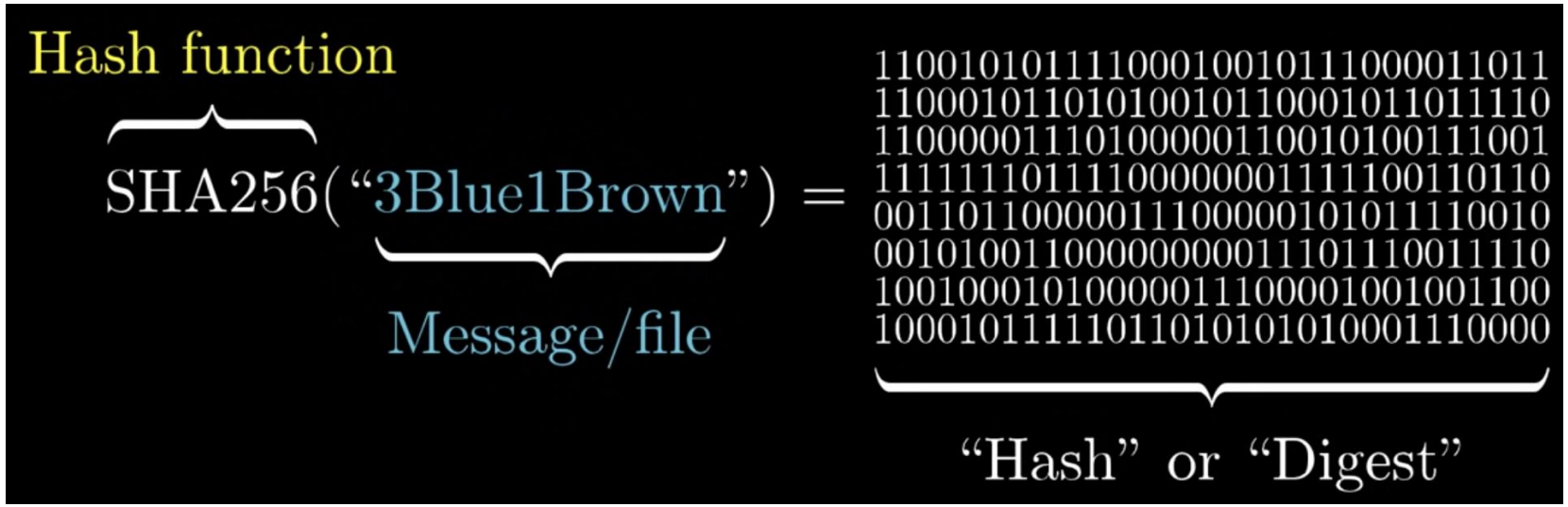
哈希函数

- 定义：
  - 哈希函数是将任意长度的消息映射成一个较短的定长输出消息的函数
  - 如下形式: h = Hash(M), M是变长的消息, h是定长的哈希值
- 目的：
  - 为文件、 消息或其它的分组数据产生 “数字指纹”

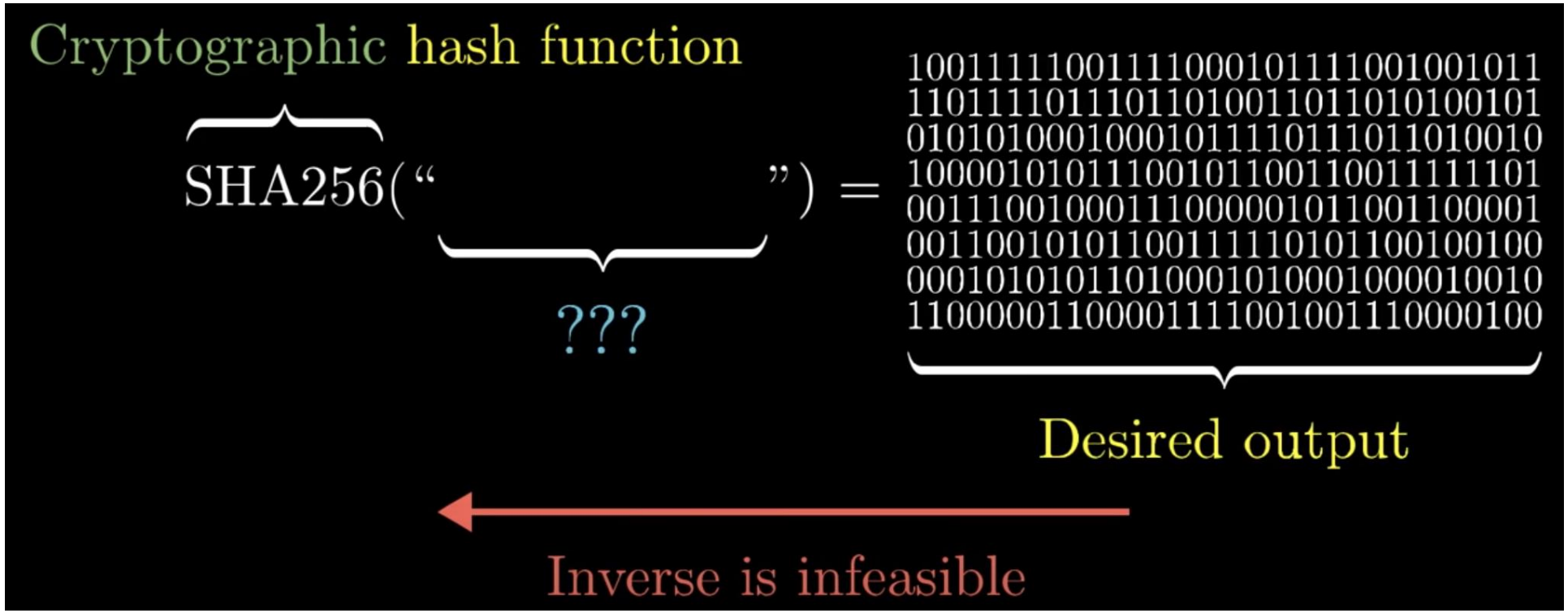


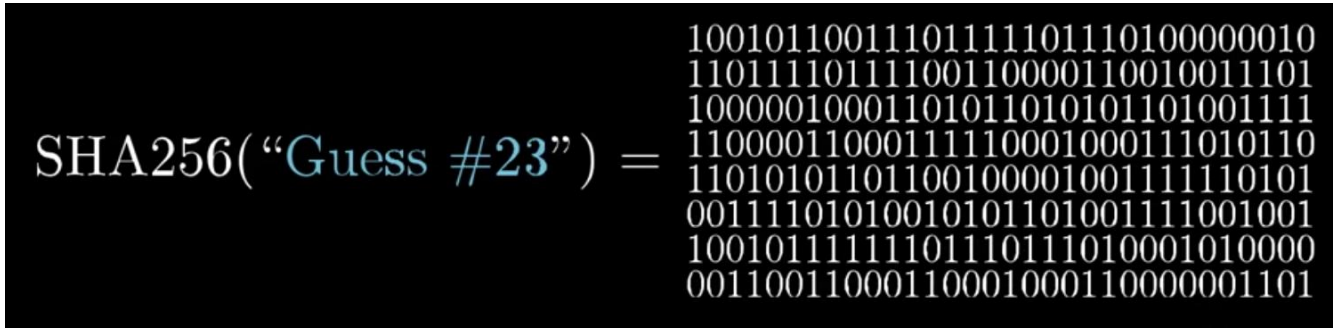
密码学 哈希函数

- 密码学**哈希函数（Hash Function）**，特性
  - 输入可以为 任意大小的 string
  - 输出固定大小(长度)，e.g., 256 bit-long
  - 有效计算：特定的输入字符串，合理时间内输出 —— *O*(n) 复杂度



密码学 哈希函数 的特性


Cryptographic hash function
SHA256(“ ”) =
???
10011111001111000101111001001011
11011110111011010011011010100101
01010100010001011110111011010010
10000101011100101100110011111101
00111001000111000001011001100001
00110010101100111110101100100100
00010101011010001010001000010010
11000001100001111001001110000100
Desired output
Inverse is infeasible


SHA256(“Guess #23”) =
10010110011101111101110100000010
11011110111100110000110010011101
10000010001101011010101101001111
11000011000111110001000111010110
11010101101100100001001111110101
00111101010010101101001111001001
10010111111101110111010001010000
00110011000110001000110000001101

比特币的密码学安全附加特性

- 如果达到*密码学安全*，采用的 哈希函数 还需如下附加特性
  - 碰撞阻力 (collision-resistance)
  - 隐秘性 (hiding)
  - 谜题友好 (puzzle-friendliness)

密码学特性1-碰撞阻力 (Collision-Resistance)

- 特性1—— 碰撞阻力 (Collision-Resistance)
  - 找不到碰撞，不代表碰撞不存在

Theorem: 假如有n+1个元素放到n个集合中去，其中必定有一个集合里至少有两个元素。

10只鸽子放进9个鸽笼，那么一定有一个鸽笼放进了至少两只鸽子。

密码学特性2 – 隐秘性 (hiding)

- 作用
  - 保证：如果仅仅知道哈希函数的输出 y = H(x)，则没有可行的办法算出输入值 x.
  - 要求：
    - x 需要取值自一个很广泛的集合
    - 仅仅通过尝试几个特定的 x，找不到特定的输出值
  - 如果：x 的取值并非来自分散的集合，怎么办?
    - e.g.,抛硬币实验：H (正面朝上) = “正面”, H (正面朝下) = “反面”。

密码学特性3

- 特性3 --谜题友好 -- 的应用
  - 什么是 谜题搜索?

搜索谜题 搜索谜题构成：

- 一个哈希函数H。
- 从高阶最小熵分布选出的一个取值，id（我们称其为谜题ID）。
- 目标集合Y。

该谜题的解决方法为一个解，x，应该满足以下公式：

$$H(id \| x) \in Y$$

要求找到一个位于集合Y内的输出值

  - Y 集合的大小决定了谜题难度
    - If Y集合的排列组合数==n，（n为谜题输出字符串的种类数），难度为0。
    - If Y集合的排列组合数==1，难度最大。

密码学特性3

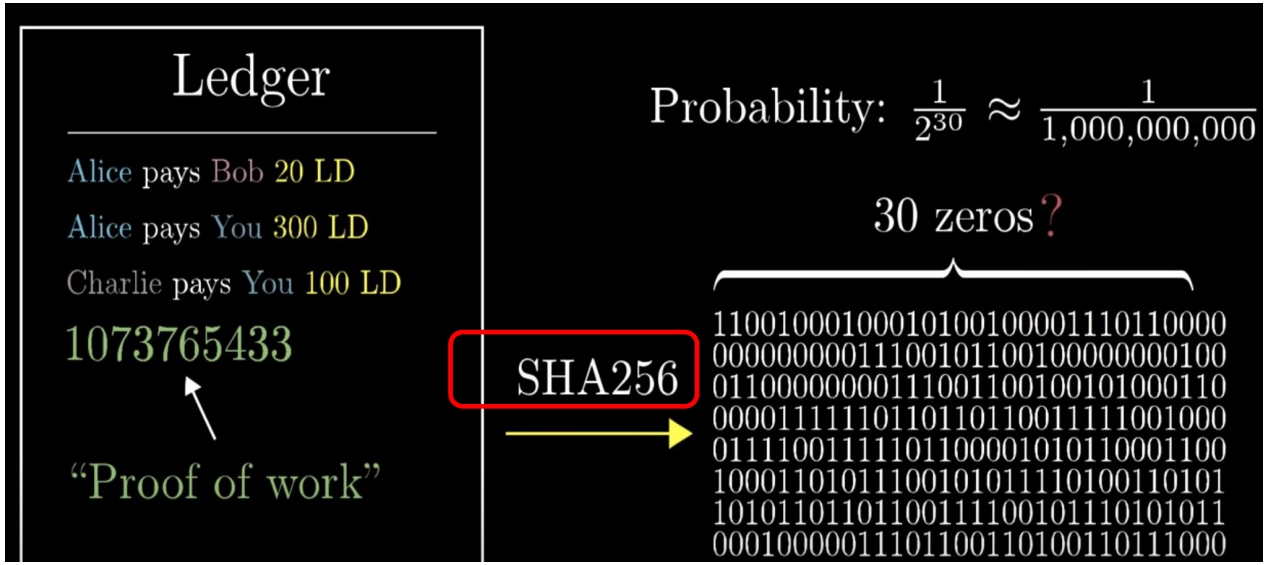
- 特性3 --谜题友好 -- 的作用
  - 谜题搜索 —— Bitcoin's mining：挖矿就是“解数学难题”？
  - 如果一个 H(id || x) 具备此特性，则 对于这个谜题没有一个解决策略，比只是随机地尝试 x 会更好。
  - 那么，我们可以为 Bitcoin 设计一个谜题搜索，来保证所有参与者的公平性

密码学特性3

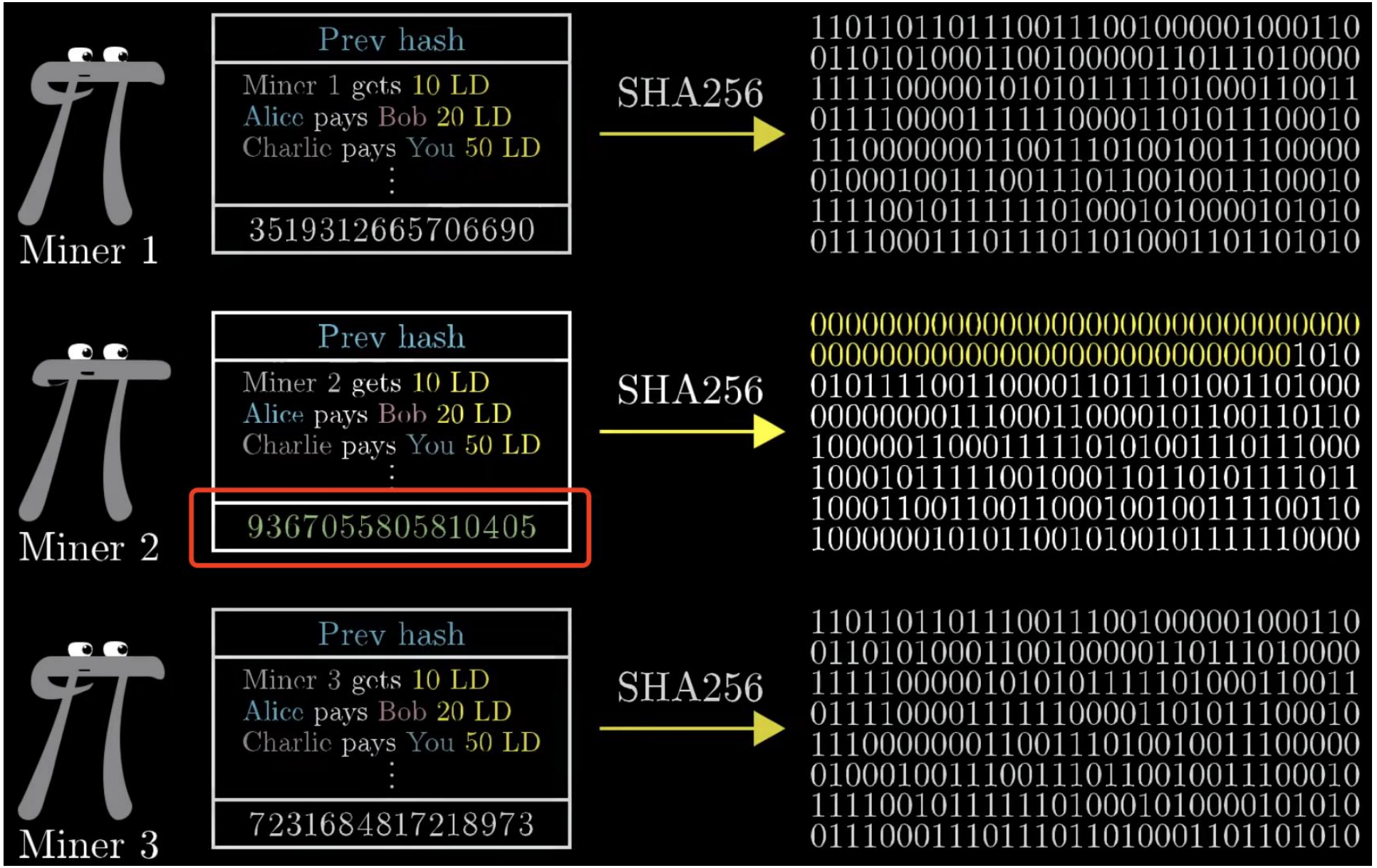
❑特性3 --谜题友好 -- 的作用

- Mining 对每一个 miner 都是公平的： 大家都不停地寻找一个合格的解 x —— nonce!

- H (header || TXs || nonce) < target



比特币的“密码学哈希函数”的应用：哈希计算



密码学特性3

- Mining 的比喻
  - H (header || TXs || nonce) < target
  - 策略：狂轰滥炸，鸟枪法

nonce

Hash Function

解空间 Y

--1.3.1 密码学基础

PART 1: 哈希函数

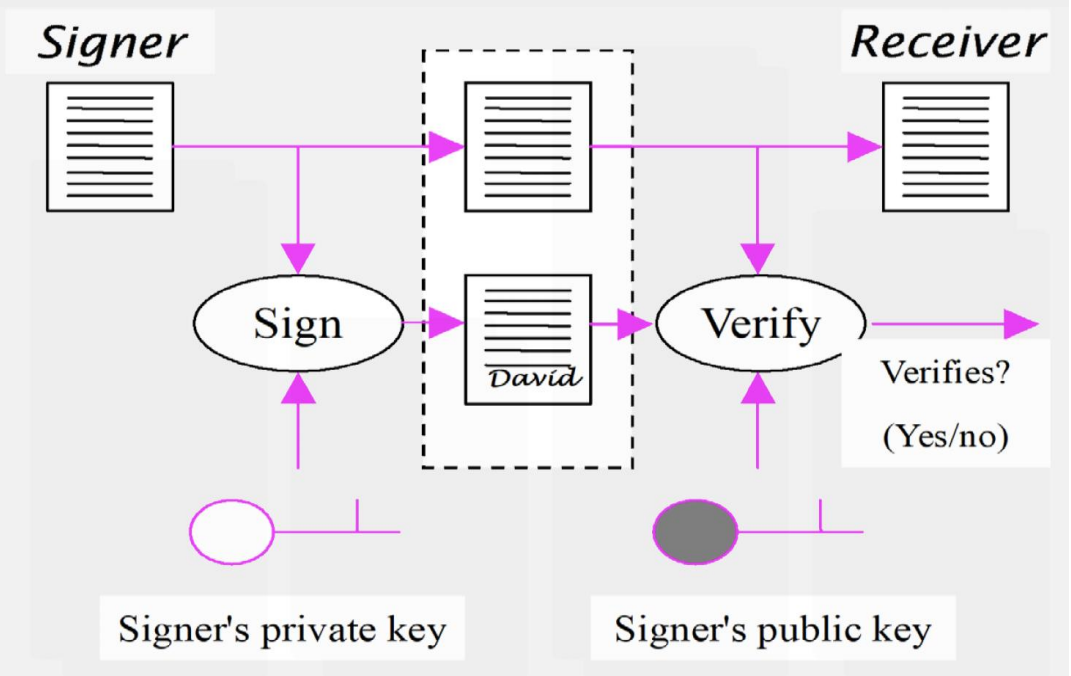
PART 2: 数字签名

引言 —— 比特币用户有账户吗？

- 银行账户
  - 中心化账户管理
- 比特币的用户没有「账户」
  - 用户自己开“账户” —— <public key, secret key>
  - <pk, sk> 来源于 非对称加密

加密货币中的公钥私钥

- 加密货币 在公链上的（转账）数据并不加密
- 那么，密码学公私钥 有什么用？—— 数字签名
  - 签名 Sign (message, sk) = Signature
  - Verify (message, Signature, pk) = Ture / False



数字签名方案

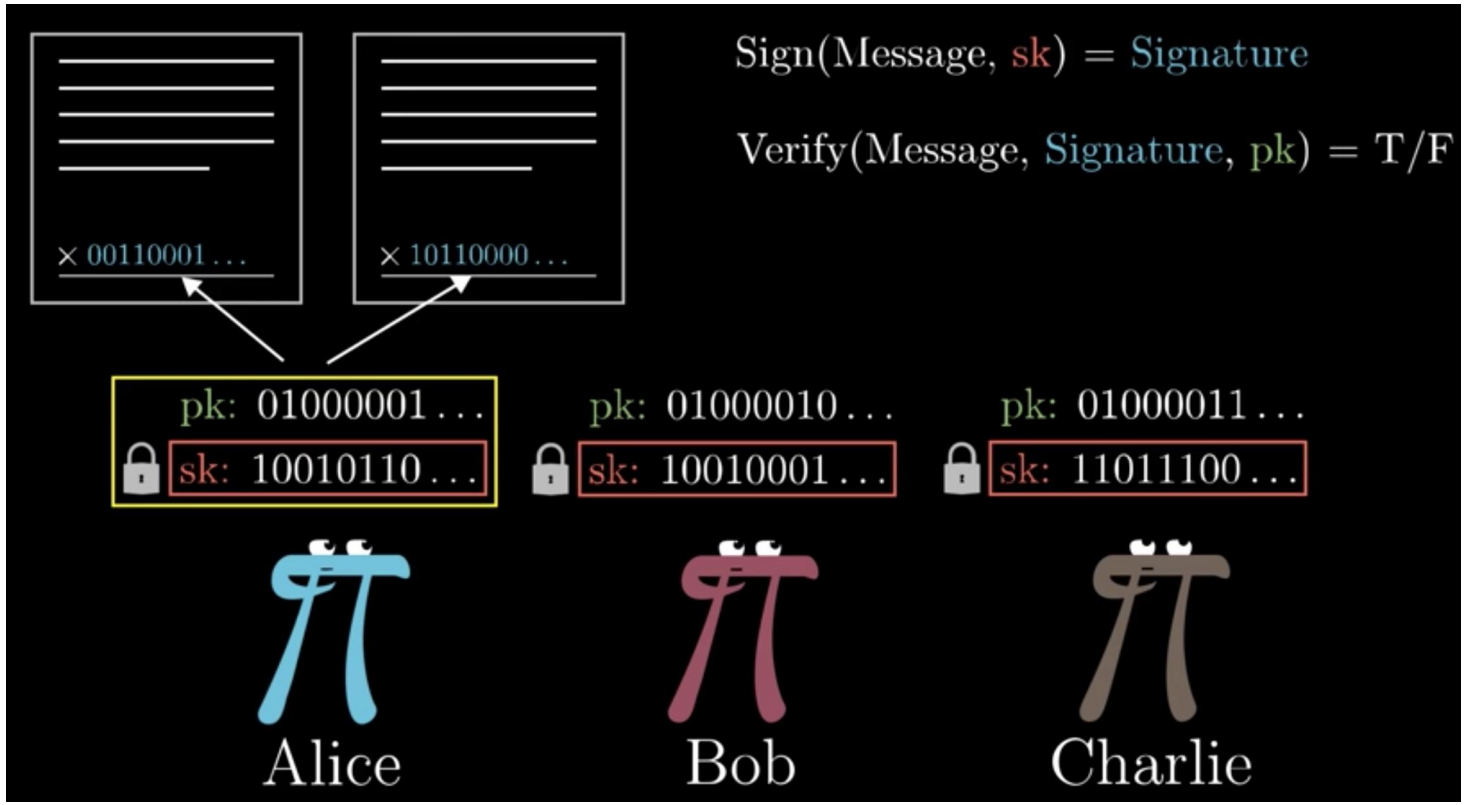
- 由三个算法构成
  - (sk, pk) := generateKeys (keysize)
    - 把 keysize 作为输入，来产生一对公钥和私钥
    - 私钥 sk 被安全保存，并用来签名一段消息；
    - 公钥 pk 是人人都可以找到的，拿到它用来验证你的签名。
  - sig := sign(sk, msg) —— 签名过程
    - 把一段消息 message 和私钥 sk 作为输入，输出是 签名sig
  - isValid := verify(pk, message, sig) —— 验证过程
    - 通过把一段消息和签名消息与公钥作为输入，
    - 如果返回是真，证明签名属实；否则，证明签名的消息为假。

比特币中签名

- 每笔 交易 的 发起方 用他自己的 私钥（sk） 签名
- 其他人 通过 发起方 的 公钥（pk） 验证 交易 的合法性

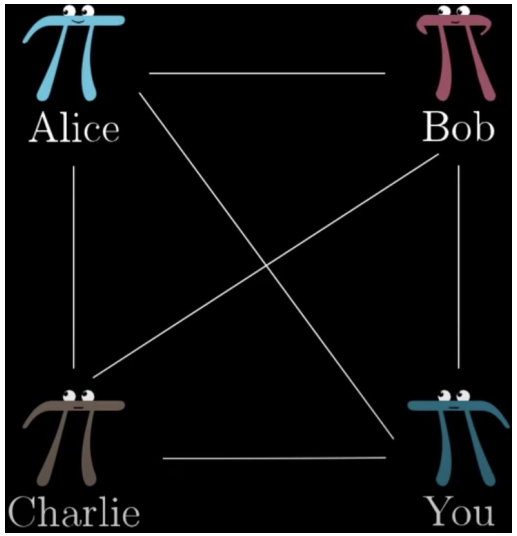
交易:

- Alice 转10 BTC 给 Bob
- Alice 需要使用私钥签名



比特币中：公钥即身份

- 为什么说 “公钥即身份”？
  - Bitcoin 用户自己开账户 —— <pk, sk>
  - 其他用户看到一个签名，并被“发起者”的 pk 验证
  - pk 就可以代表 一笔交易的“发起者”的 身份



- 去中心化身份管理
  - 随时定制新的随机身份
    - new <pk, sk> = generateKeys (keysize)
  - 匿名：一个人可以有多个 <pk, sk>

1.3 区块链的原理 —— 以比特币为例

- 以 比特币 为代表的 区块链技术，原理主要包括
  - 1.3.1 密码学基础
  - 1.3.2 区块链数据结构
  - 1.3.3 （比特币的）共识机制

引言：一个区块中到底有什么内容？

- 区块链的数据结构 == Block + Chain

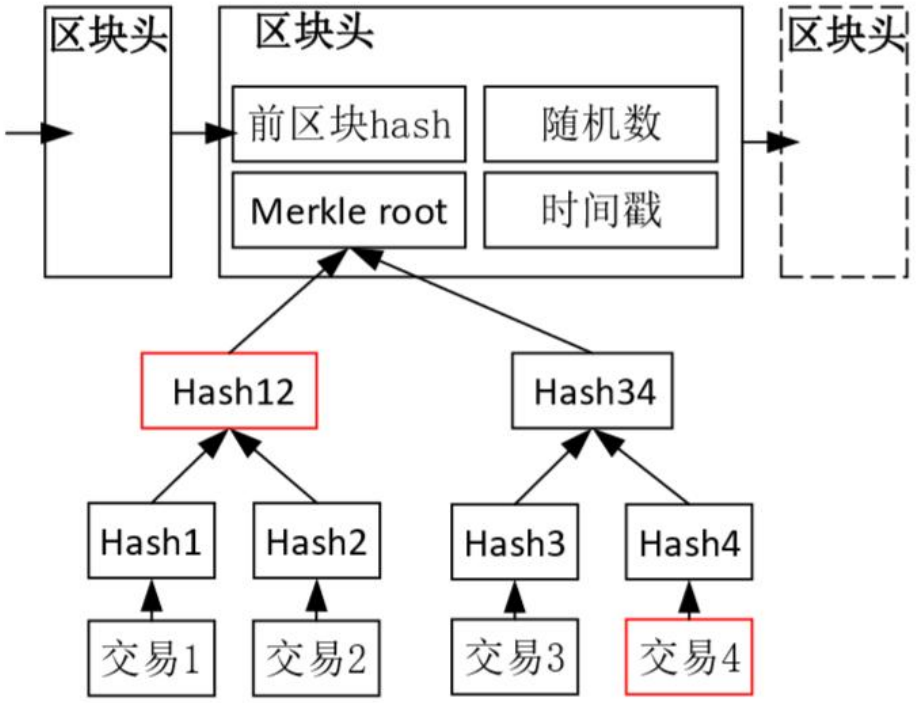
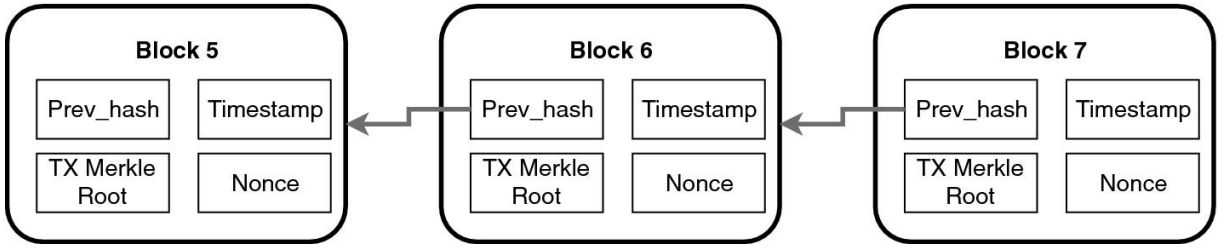


Fig. 2 Bitcoin blockchain data structructure

图 2 比特币区块链结构



Size	Field	Description
4 bytes	Version	A version number to track software/protocol upgrades
32 bytes	Previous Block Hash	A reference to the hash of the previous (parent) block in the chain
32 bytes	Merkle Root	A hash of the root of the merkle tree of this block's transactions
4 bytes	Timestamp	The approximate creation time of this block (seconds from Unix Epoch)
4 bytes	Difficulty Target	The proof-of-work algorithm difficulty target for this block
4 bytes	Nonce	A counter used for the proof-of-work algorithm

1.3 区块链技术原理

-- 1.3.2 区块链数据结构

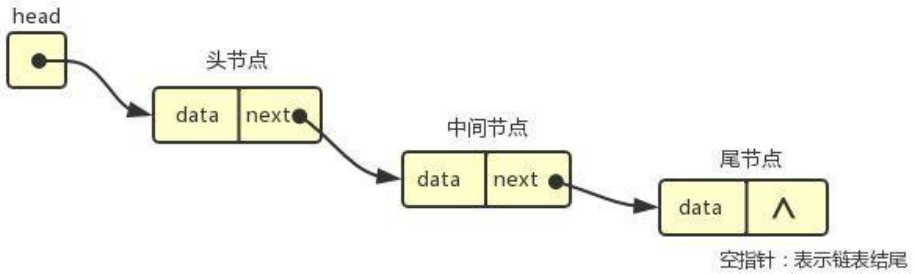
Part 1: 哈希指针

Part 2: 默尔克树

数据结构1 —— 哈希指针 是什么？

- 哈希指针: *ptr = HashFunc (Input Data)
  - 不仅告诉你 数据在哪里
  - 而且允许 验证该数据是否被修改过： 对数据加锁

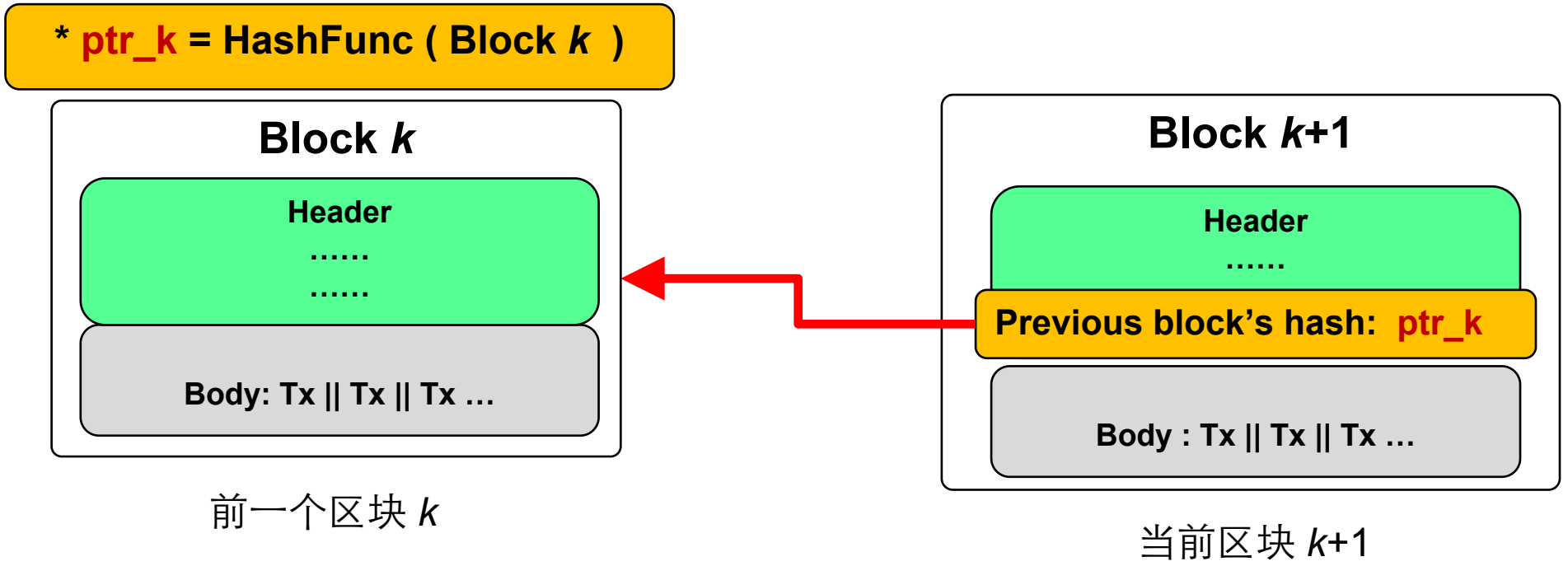
类比普通指针：



ptr: 哈希指针

数据结构1 —— 哈希指针 有什么用？

- 哈希指针 用来组成 —— 区块链的 “链”
  - 指向前一个区块的 “哈希指针” := HashFunc (前一个区块)



1.3 区块链技术原理

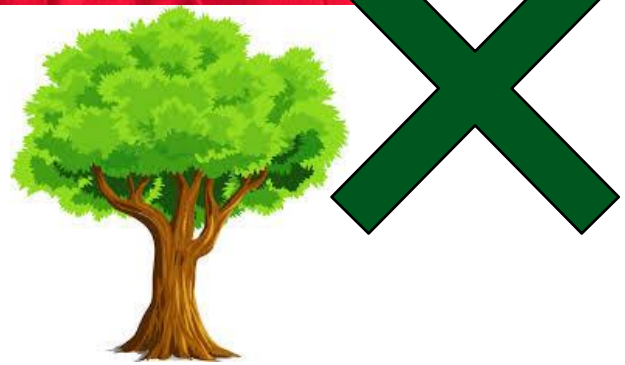
-- 1.3.2 区块链数据结构

Part 1: 哈希指针

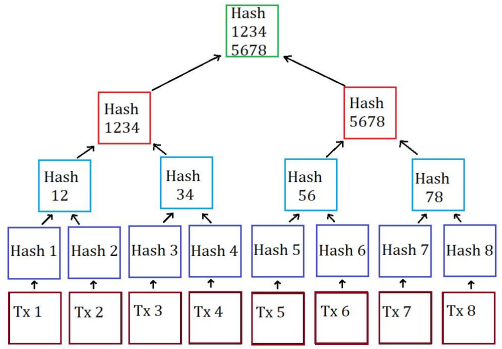
Part 2: 默尔克树

数据结构2 —— 默克尔树

❑ 不是 **Merkel + tree**



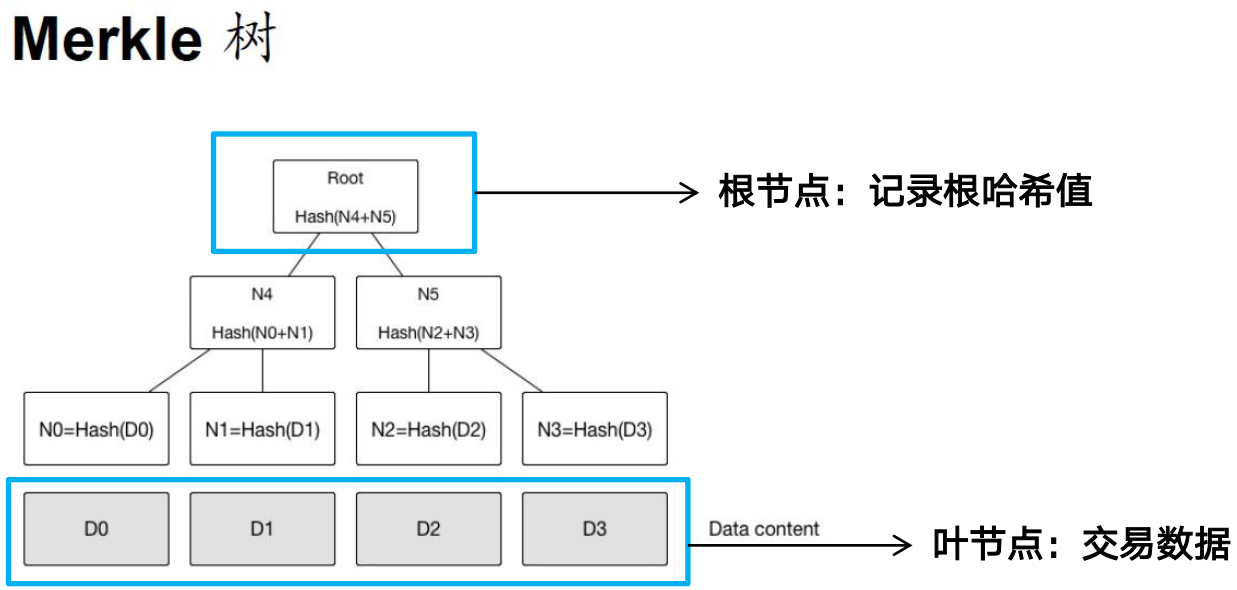
❑ 而是 **Ralph Merkle**



于 1979 年提出“默克尔树”

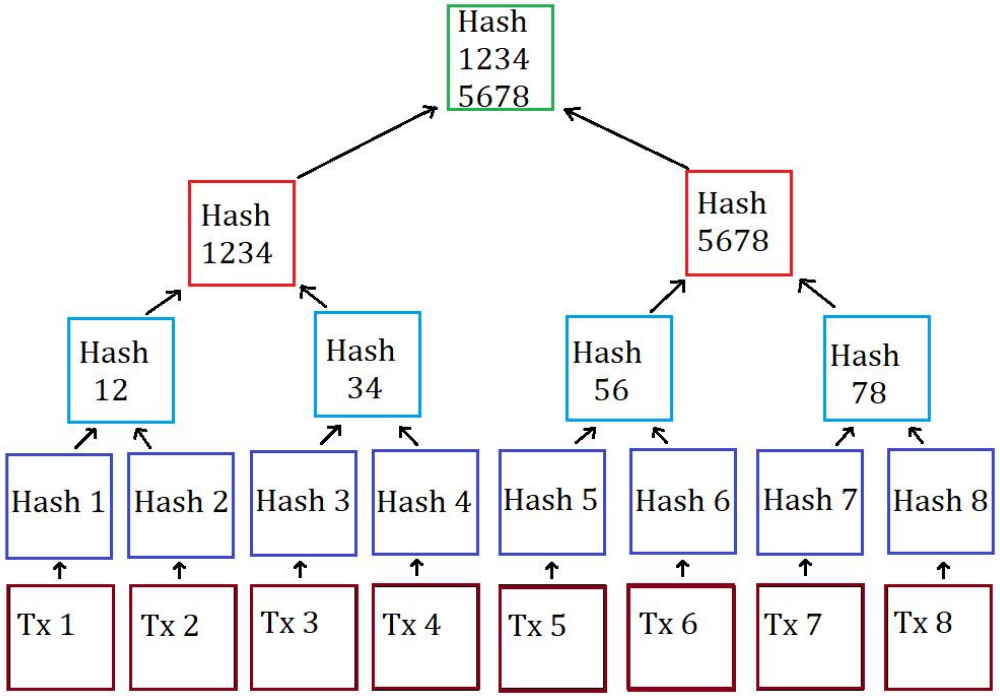
数据结构2 —— 默克尔树 是什么？

- 使用了 哈希指针 的二叉树 —— Merkle Tree
- 本质上它是一种二叉树，由一个根节点、一组中间节点和一组叶节点组成。
  - 底层数据的任何变动，都会传递到其父节点，一直到传递到 根节点（Merkle Root）



数据结构2 —— 默克尔树 有什么用？

- 默克尔根（Merkle Root）的哈希值
  - 代表一个区块中所有交易的哈希值
  - 一旦有交易被篡改了，根哈希值也会改变 —— 防篡改的原因
  - 所以说：根哈希值是一个区块中所有交易数据的“锁”



1.3 区块链的原理 —— 以比特币为例

- 以 比特币 为代表的 区块链技术，原理主要包括
  - 1.3.1 密码学基础
  - 1.3.2 区块链数据结构
  - 1.3.3 （比特币的）共识机制

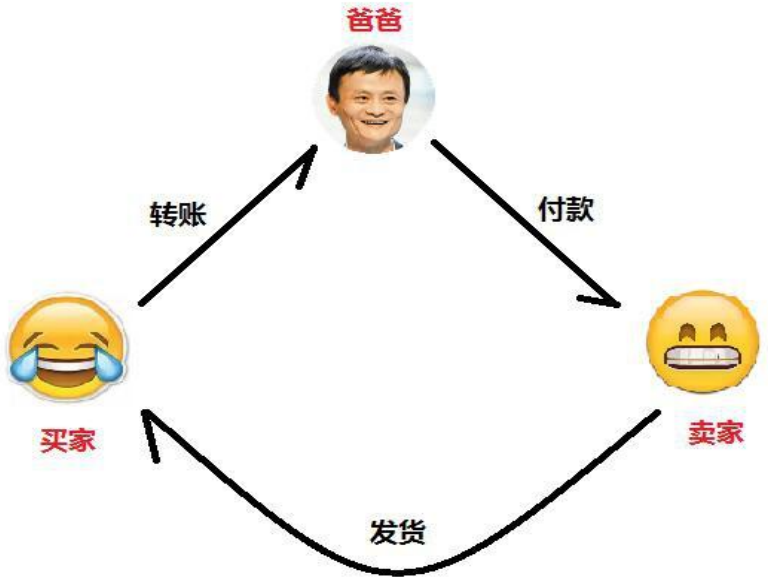
1.3 区块链技术原理

-- 1.3.3 （比特币）共识机制

PoW：挖矿

为什么需要共识 —— 分布式环境下如何记账？

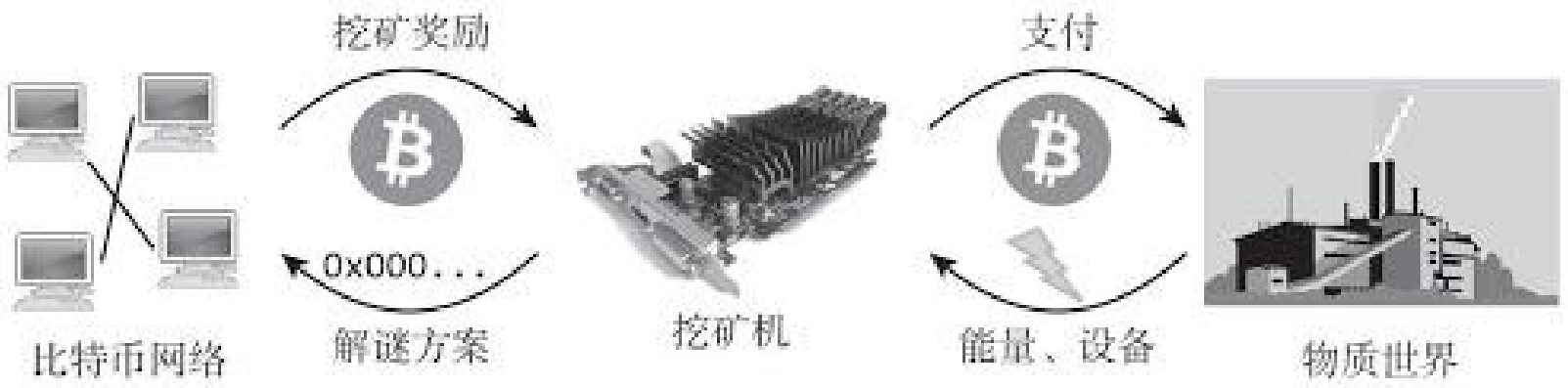
- 集中式：第三方中介负责记账
- 比特币：每10分钟产生一个块，谁负责打包这个块?
- 如果负责打包的用户是恶意的呢?



提高作恶的成本 —— 竞争机制

- 设计一套机制：争夺记账权 —— 挖矿
  - 记账有利润：**比特币奖励** + **交易手续费**
  - 很多人争夺记账权
  - 通过付出计算量解决一个难题，谁先解决谁获得记账权
  - 坏人作恶的成本变高

- 发布区块的过程就是比特币发行的过程



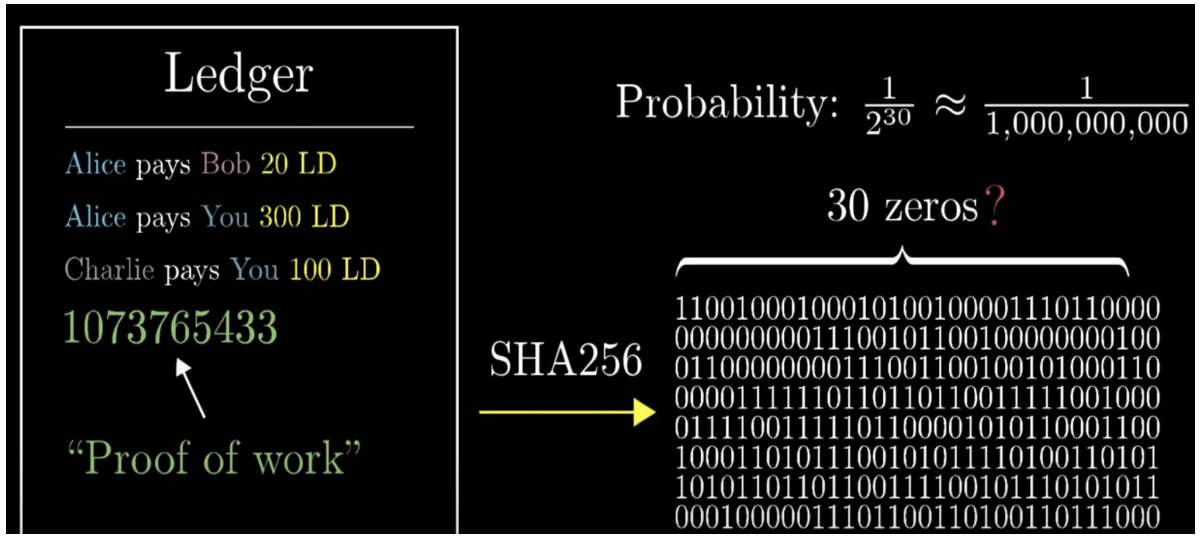
比特币的货币发行方式 — 挖矿 (mining)

挖矿 即 哈希 (计算)

比特币的 挖矿原理

❑ 挖矿成功 ——

- SHA256 (Markle Root+上一个区块Hash值+时间+Nonce) < **难度系数**



❖"**0000000000000f15673f1354**";

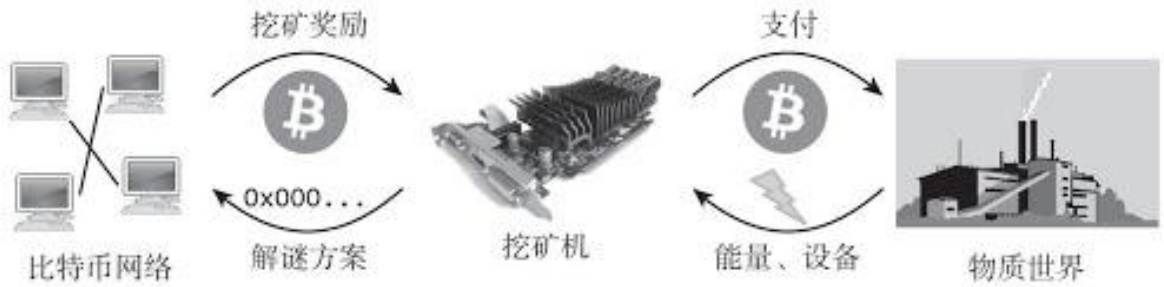
❖"**0**"的数目体现了**挖矿难度**

工作量证明 —— 资源消耗

Proof-of-Work

证明

资源消耗



- 比特币共识机制
  - 身份确认
  - UTXO交易模型
  - 交易信息记录
  - 工作量证明

比特币共识机制 关键词

- 比特币身份确认　身份确认：公私钥体系
- UTXO交易模型　交易服务
- 链式交易信息记录　记录管理
- 工作量证明(POW)　信任规则

为何需要共识？

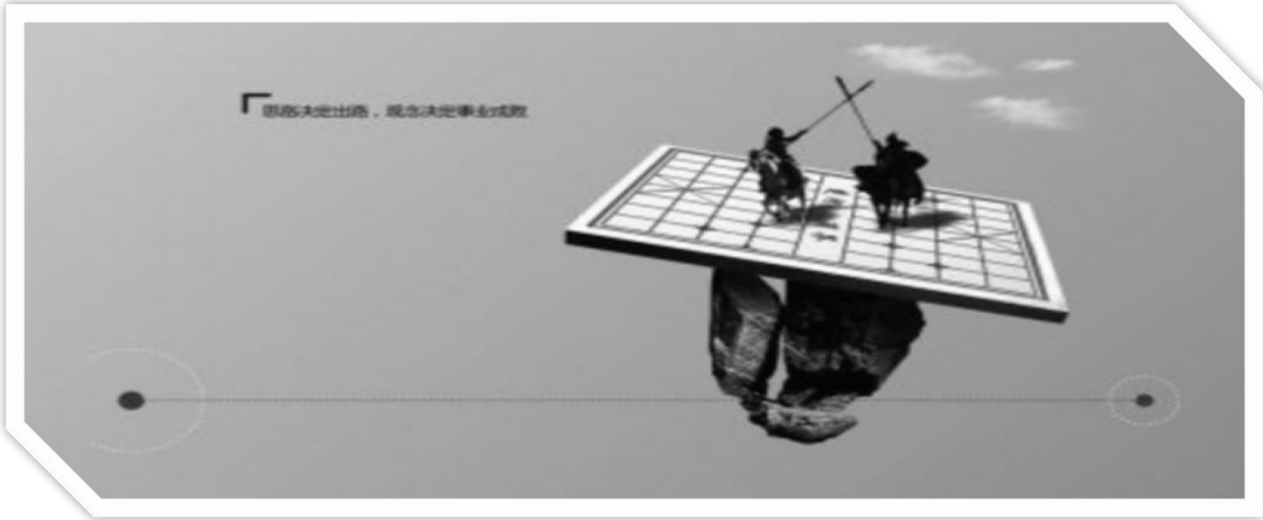
- 从交易的角度，不共识会有什么样的场景?
  - 一个提议 (e.g., a Tx, or a block) 广播出去，
    有人收到，有人没收到
  - 假如有人发送假消息，怎么办?
    - 假如有人尝试：一个 coin 花两次

共识机制：区块链的核心引擎

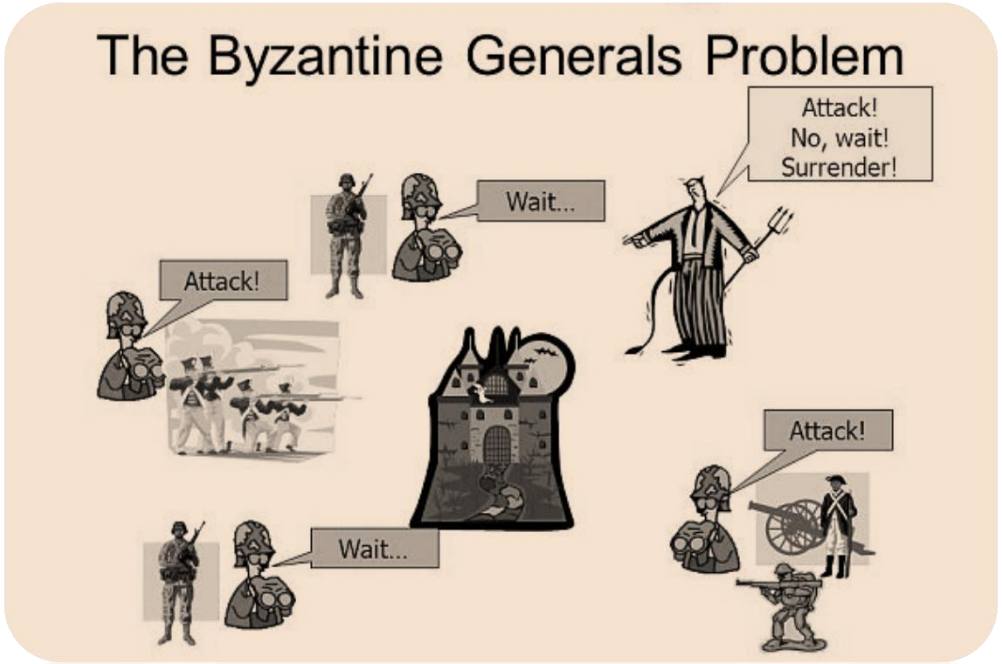
- 定义：
- 一种多方协作机制，用于协调多参与方达成共同接受的唯一结果，且保证此过程难以被欺骗，且持续稳定运行

共识模型里体现的博弈论

- 拥有记账权的人更倾向在维护整个体系过程中获利（纳什均衡+帕累托最优）
- 使用网络的人需要付出一定的成本（手续费、计算费）以免滥用 （避免公地悲剧）
- 少数人作恶的成功几率很低, 参考“赌徒破产问题”（Gambler‘s Ruin problem）
- 只有极端势力才有可能不顾一切的颠覆这个体系
- 整个局势不存在“确定性”，一直在动态的多方博弈

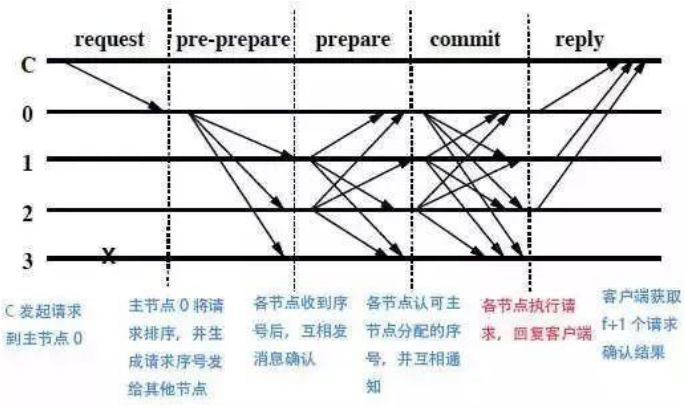


拜占庭将军问题和解决方案 **(Byzantine Fault Tolerance)**



预设条件

- 至少4个以上参与者
- 每轮次有一个发令者
- 少于1/3的参与者作恶或失效
- 极大概率可达的网络（区块链网络）
- 可控的网络规模（少于100参与者）



BFT拜占庭将军问题：Lamport，Shostak 和 Pease 于1982年的一篇学术论文中引入，Miguel Castro 和 Babara Liskov 在1999年提出 PBFT，放松了约束来解决拜占庭问题。Liskov于2008年获得了图灵奖

拜占庭将军问题相关的图灵奖得主

Lamport分布式计算理论奠定了这门学科的基础。他在1978年发表的论文《分布式系统内的时间、时钟事件顺序（Time, Clocks, and the Ordering of Events in a Distributed System）》成为计算机科学史上被引用最多的文献。他为“并发系统的规范与验证”研究贡献了核心原理。

2008年，美国计算机协会(ACM) 宣布Barbara为当年年度图灵奖获得者，以表彰其在程序设计语言与系统设计，特别是在数据抽象、容错和分布式计算领域的实践和理论基础方面的贡献。

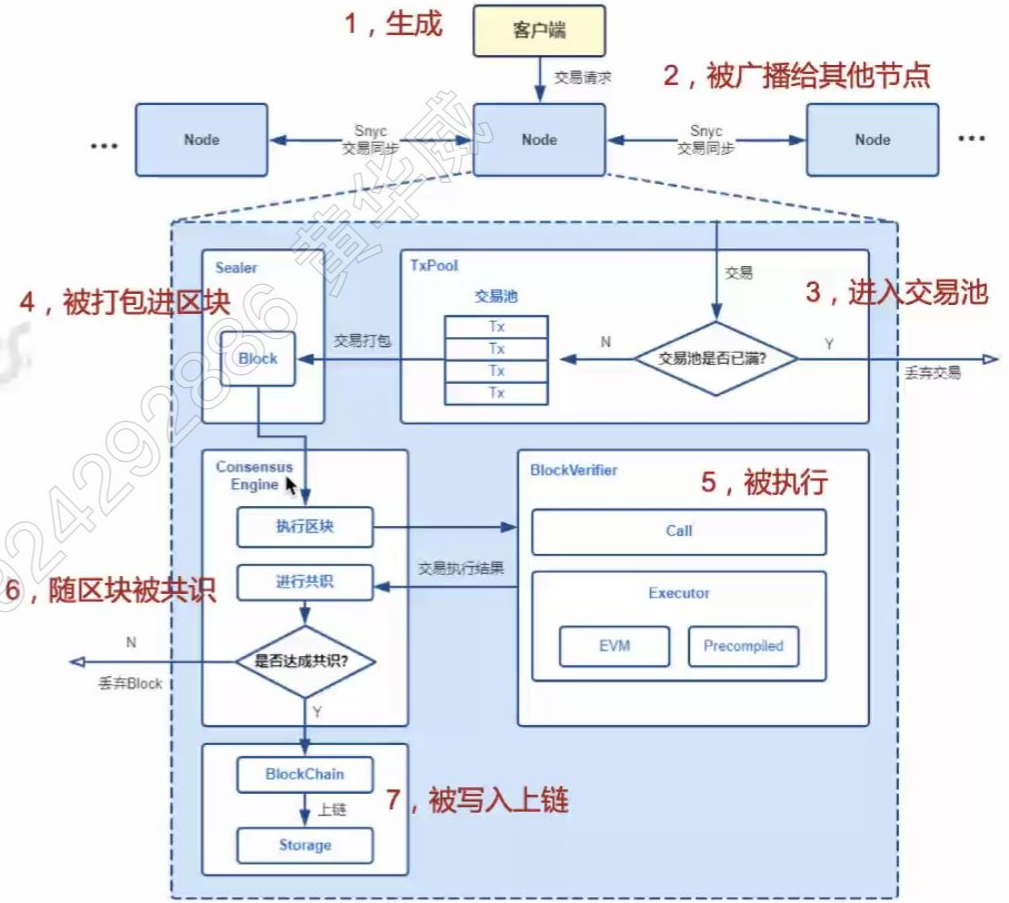
比特币共识机制

- ~~比特币身份确认~~ 身份确认
- UTXO交易模型 交易服务
- 链式交易信息记录 记录管理
- 工作量证明(POW) 信任规则

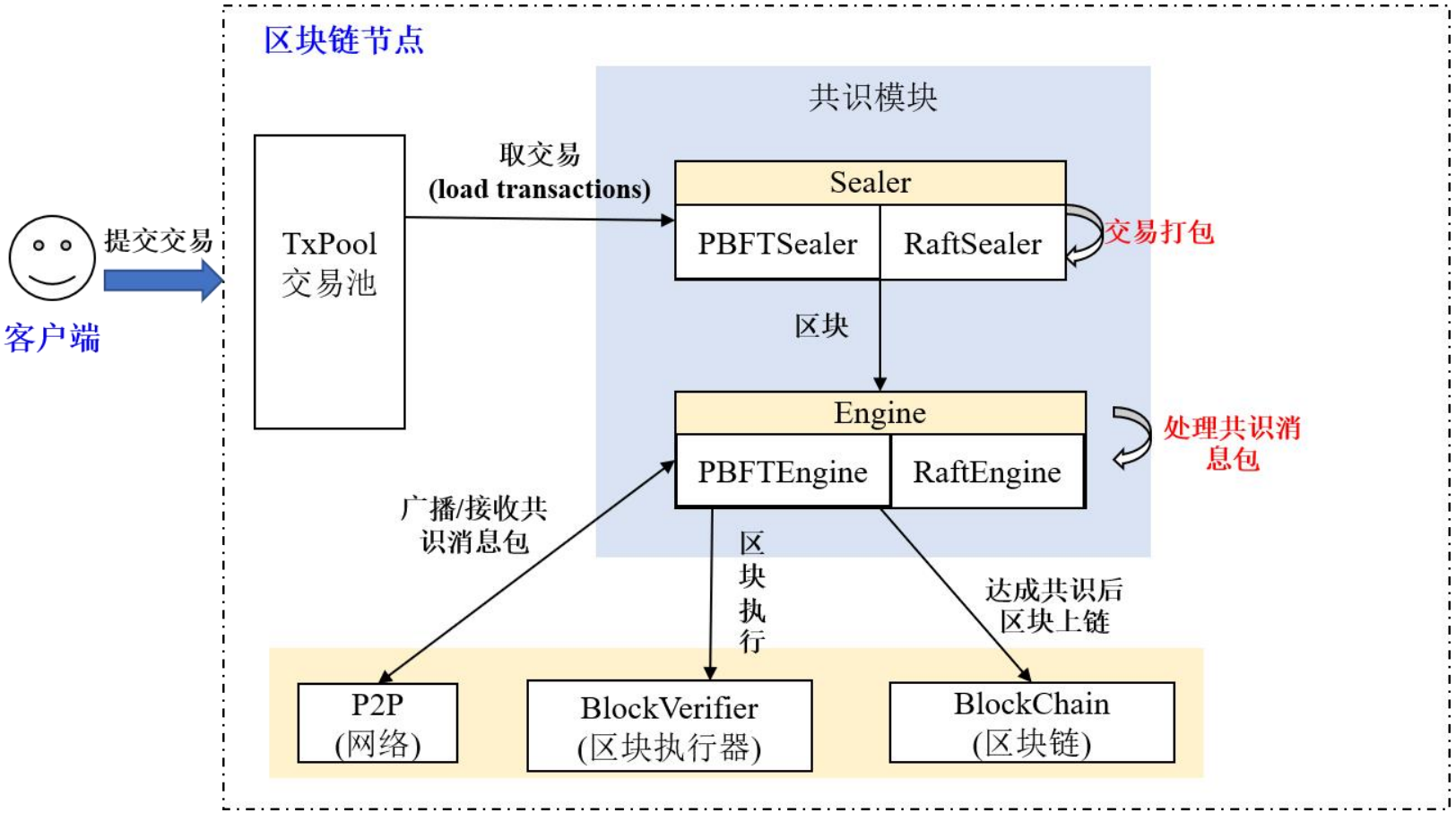
补充知识：交易生命周期

交易生命周期

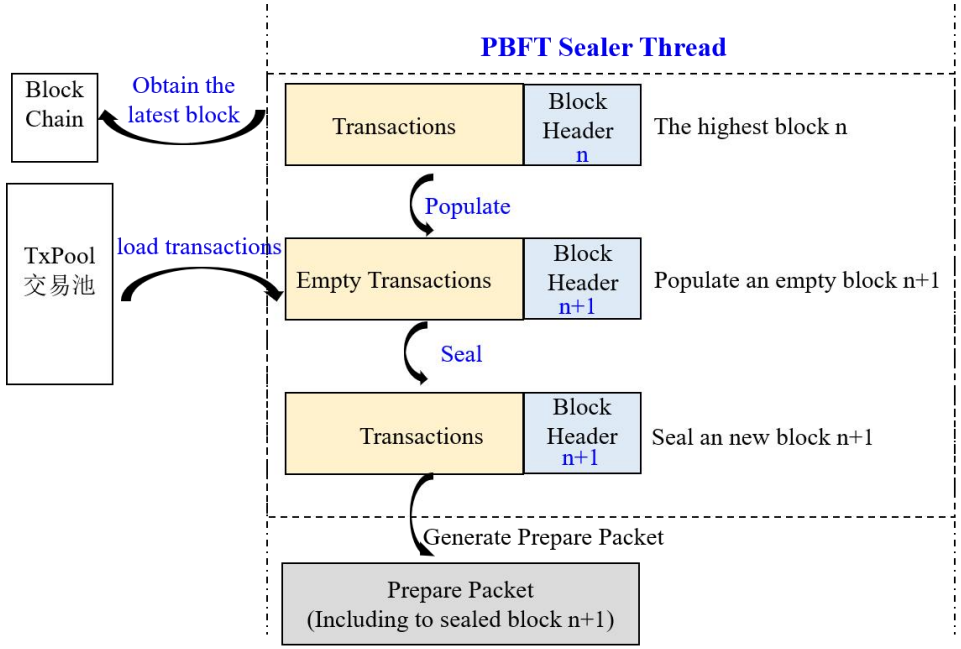
- 网络模块：广播交易和区块
- 交易池模块：缓存交易
- 打包共识模块：交易打包，节点间共识
- 交易执行模块：执行交易，获得状态变更
- 存储模块：落盘存储交易、区块等数据



补充知识：交易打包共识框架



补充知识：交易打包（以PBFT为例）



1. 产生新的空块: 通过区块链(BlockChain) 获取当前最高块，并基于最高块产生新空块

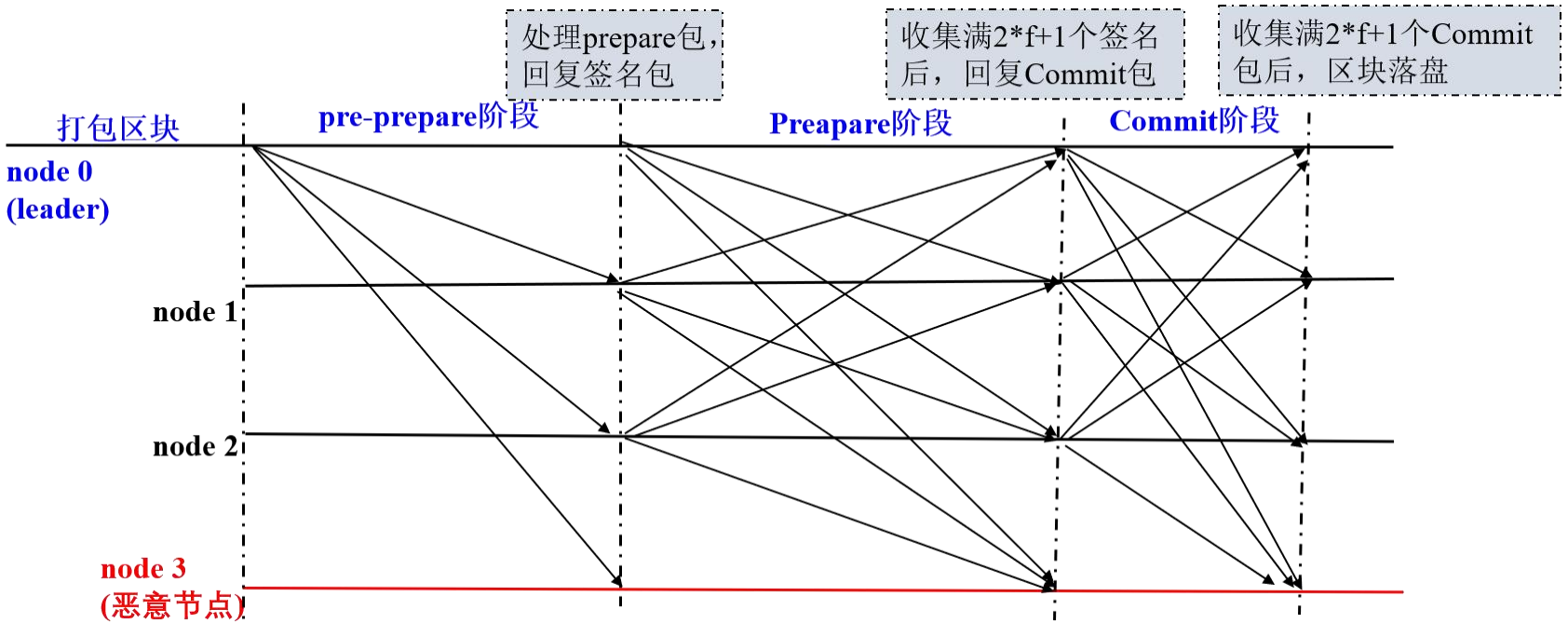
2. 从交易池打包交易: 产生新空块后，从交易池中获取交易，并将获取的交易插入到产生的新区块中；

3. 组装新区块: Sealer线程打包到交易后，将新区块的打包者(Sealer字段)置为自己索引，并根据打包的交易计算出所有交易的transactionRoot；

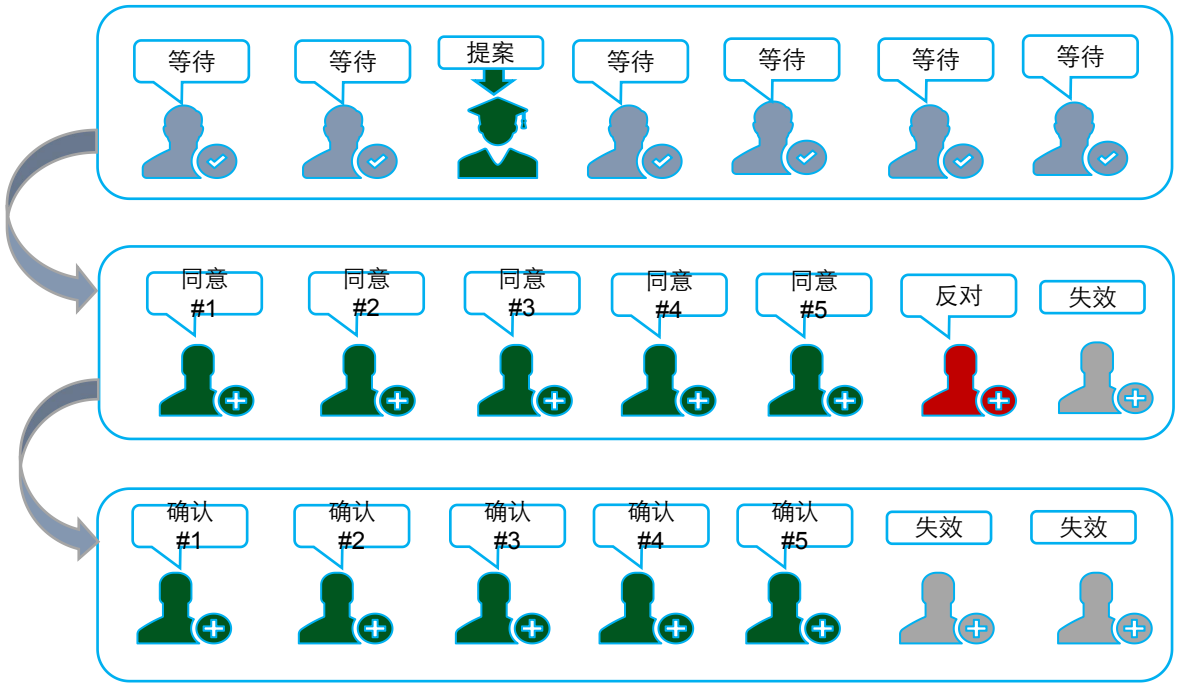
4. 产生Prepare包: 将组装的新区块编码到Prepare包内，通过PBFTEngine线程广播给组内所有共识节点，其他共识节点收到Prepare包后，开始进行三阶段共识

补充知识： PBFT共识算法

- 广播模型，三次广播，容错1/3节点故障



补充知识：一次PBFT协商过程：民主集中制



(提议：本小组周二上午开会)

确认记账者列表,每一轮次选出新的提案人,提案人排序打包，广播提案

(投票：周二上午开会,同意/反对/不表态)

所有记账者针对提案进行检验（检查交易，运行合约等），都通过的话发出同意投票，超过2/3进入下一轮

(确认：大家设定日程，并反馈参加确认消息)

所有记账者表示可以收妥提案，如果超过2/3人表示收妥，则提交存储，进入下一轮

- 实际的处理过程非常复杂，需要考虑：

公平高效的选出记账者列表 | 议长轮换和存活检测 | 超时进行轮次切换 | 共识时间 | 网络波动 | 广播流量 | 交易计算量 | 区块同步校验 | 过多节点不共识 | 网络规模太大 | 极端情况下的崩溃恢复

TRANSACTIONS
交易服务

交易有效性

❖如何确认一笔交易的有效性？

- 所有权确认（签名）
- 具有可动用的资金
- 其他交易不会用到同一笔资金

所有权-确认

Alice 要给 Bob 5个币，他可以用别人的5个币吗？

- 每一笔交易包含所有者的公钥（确定归属）
- 如果要花该交易收入的币，你必须提供相应的私钥（签名）
- 因此，如果你能拿到别人的私钥 ……

UTXO交易模型

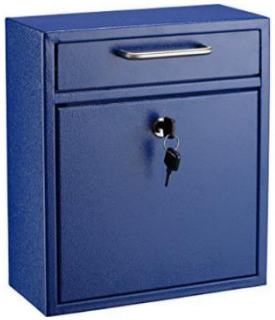
- ~~所有权确认~~
- 具有可动用的资金
- 其他交易不会用到同一笔资金

- 由于缺乏中心化的机构管理，与传统银行中使用账户结余不同，比特币使用了 Unspent Transaction Output (UTXO) 来确保
  - 同一笔资金只出现在一笔交易中

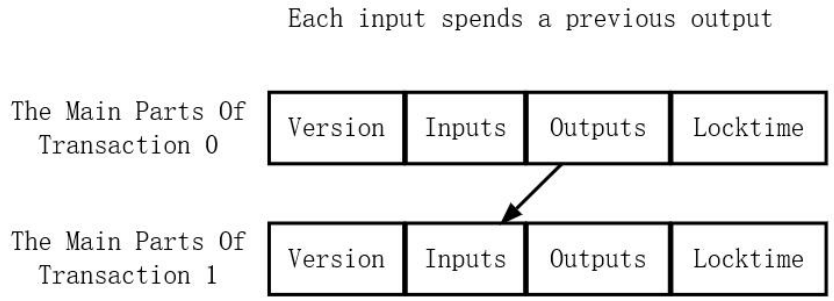
- Unspent Transaction Output (UTXO)： 未花费的交易输出

UTXO交易模型

❖**传统：**每笔收到的资金存储在保险箱里，每次交易多少就从里面取多少



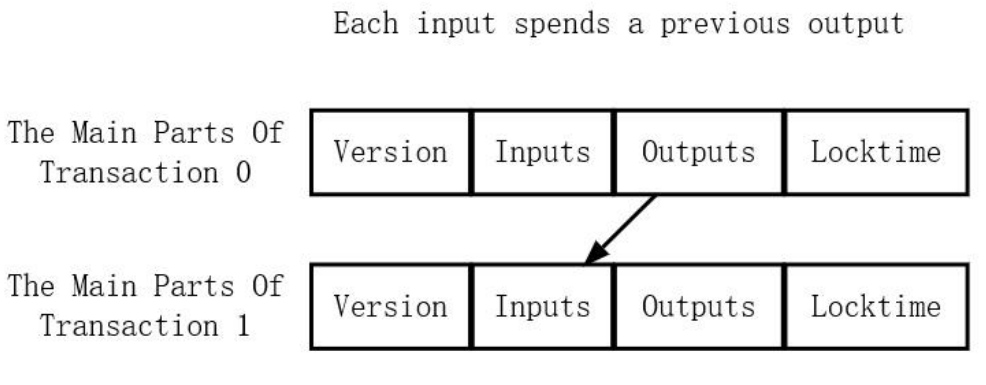
❖**UTXO：**每一笔收到的资金都存在一个储蓄罐里，每次交易都需要打碎一个或多个储蓄罐



❖**每笔新的交易都需要打破旧的 “UTXO”以生成新的“UTXO”**

注：比特币交易是可分的，最小的单位叫做 satoshi，等于0.0000001BTC

UTXO交易模型



❖UTXO交易模型基本形式：

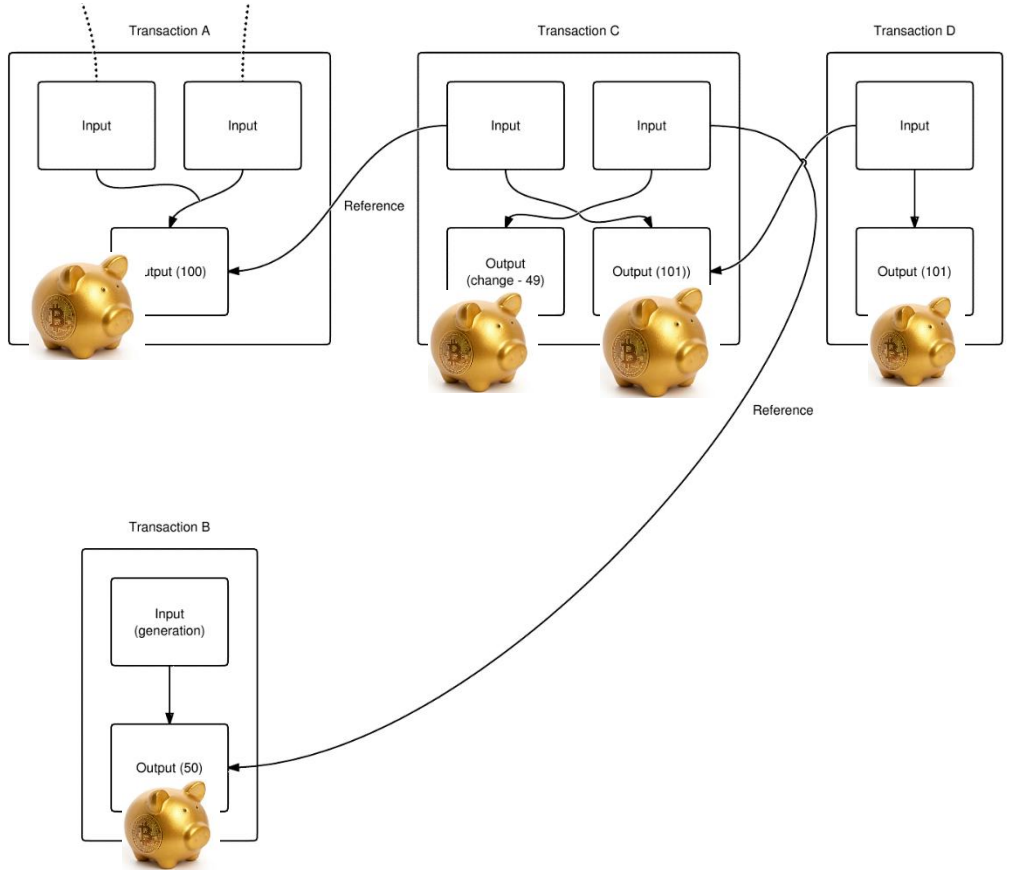
版本号+Input: 未花费交易UTXO + Output:支付地址及数量+锁定时间

❖安全性：

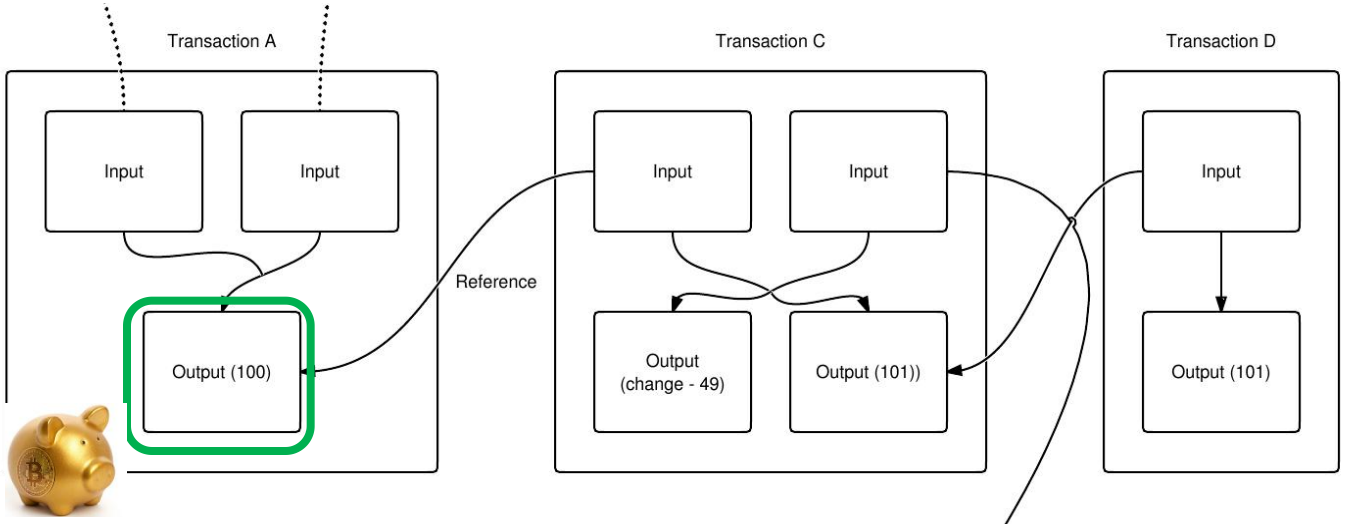
结合所有权确认，每个交易都包含着比特币拥有者的签名，所有交易记录可溯源，记录着每个比特币从“出生”开始的所有“主人”

UTXO交易模型

UTXO：从比特币自身角度上，每次交易都是在某块“**比特币地皮**”的“**所有权证**”上改改名字而已

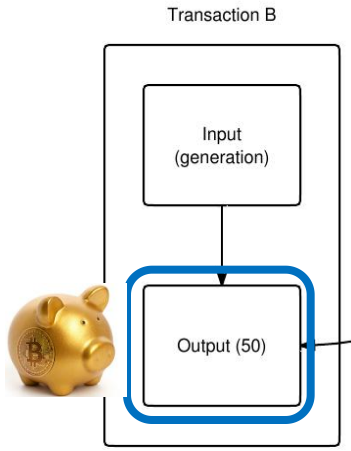


UTXO交易实例



小王

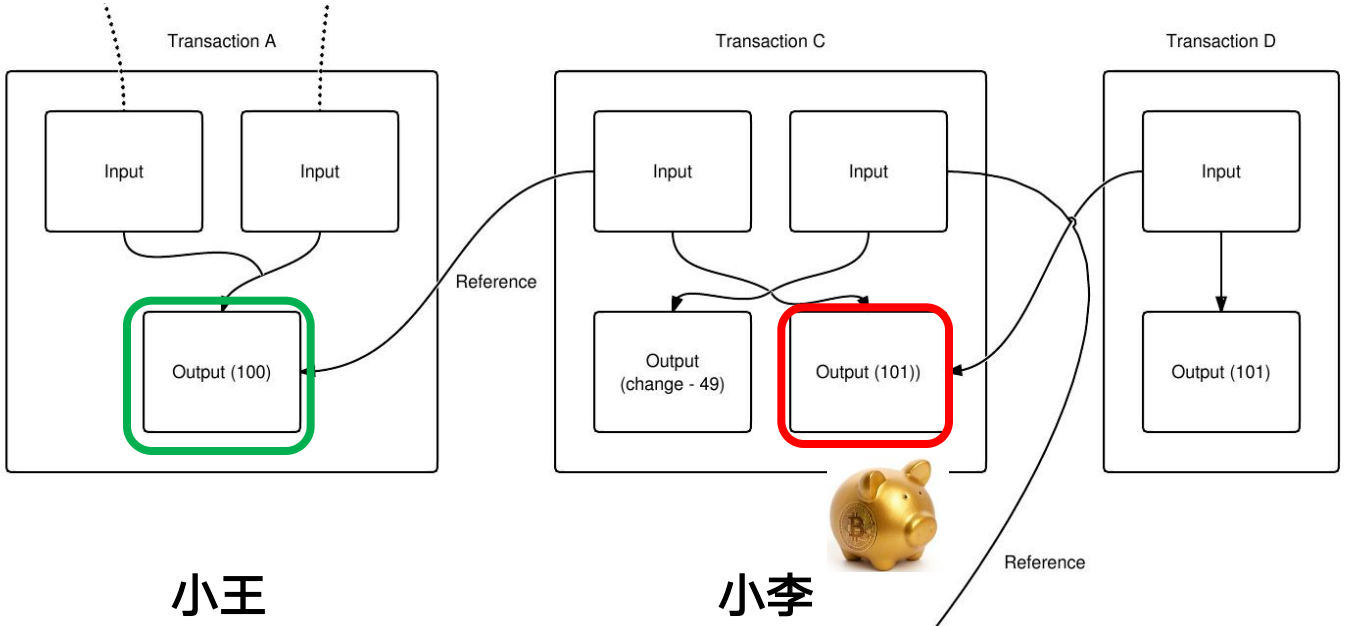
小李



- 小王要转给小李101个比特币，手头有2笔比特币：

□一笔100个比特币（老王给的）
□一笔50个比特币（挖矿）

UTXO交易实例

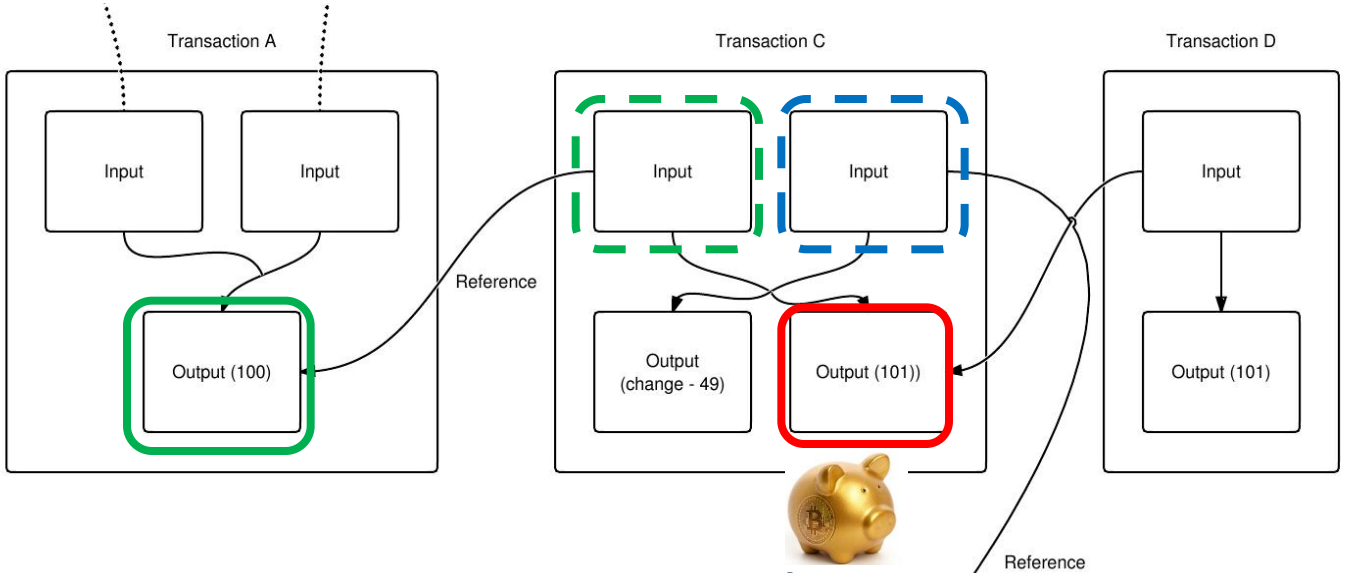


- 小王要转给小李101个比特币，手头有2笔比特币：

口一笔100个比特币（老王给的）
口一笔50个比特币（挖矿）

- 101=

UTXO交易实例

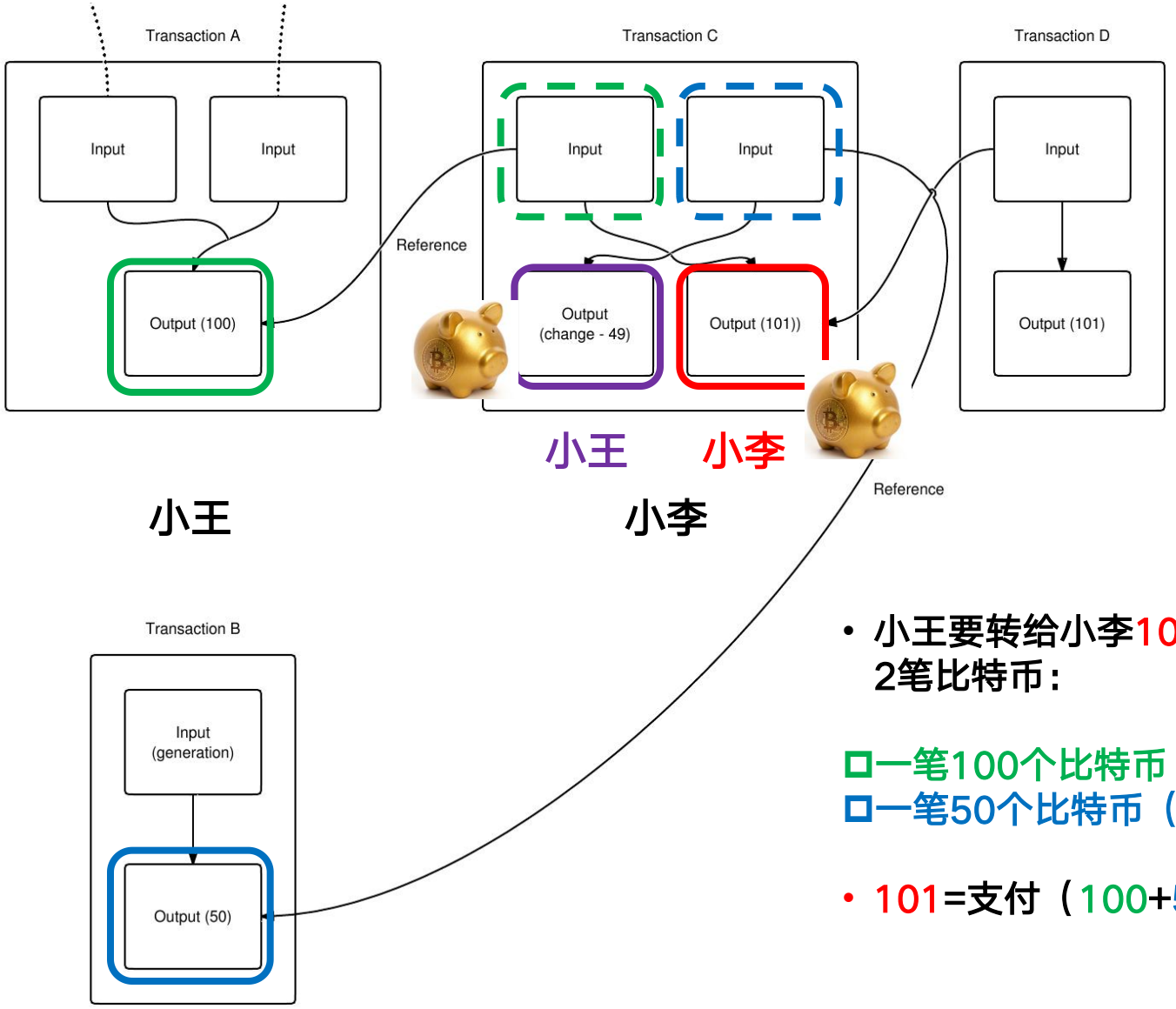


- 小王要转给小李101个比特币，手头有2笔比特币：

□一笔100个比特币（老王给的）
□一笔50个比特币（挖矿）

- 101=支付（100+50）

UTXO交易实例

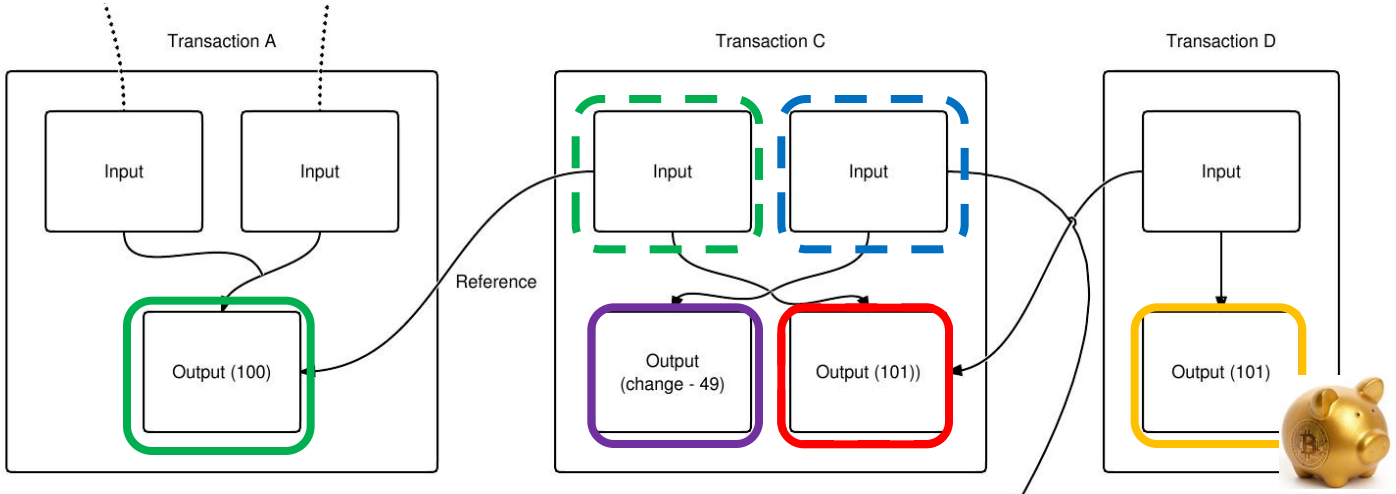


- 小王要转给小李101个比特币，手头有2笔比特币：

□一笔100个比特币（老王给的）
□一笔50个比特币（挖矿）

- 101=支付（100+50）- 找零（49）

UTXO交易实例



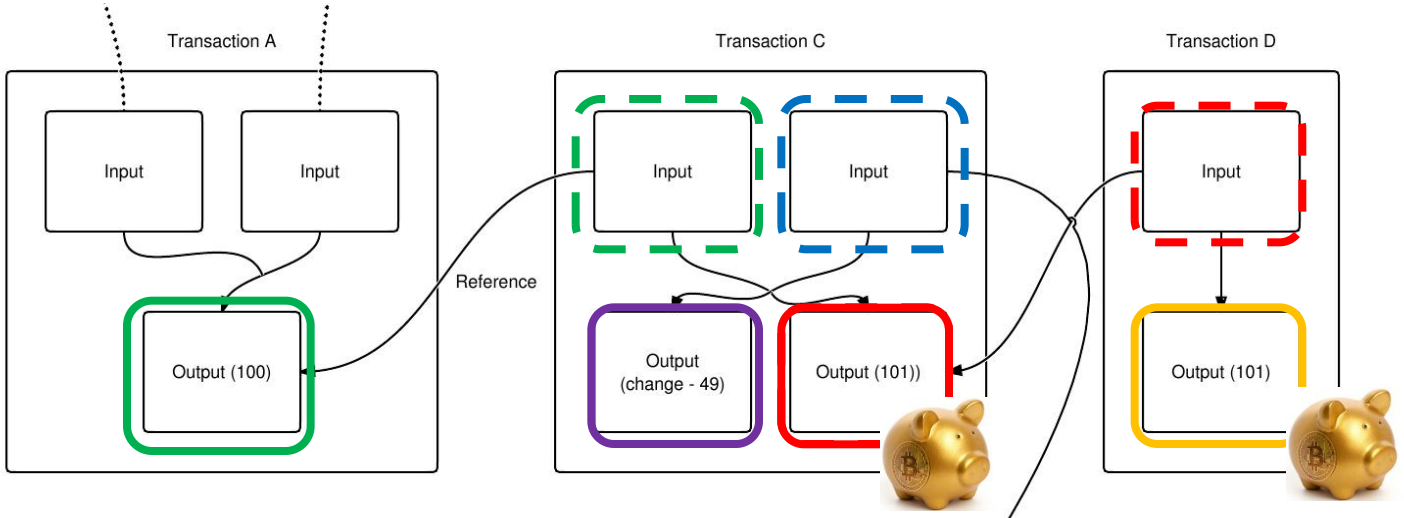
- 小王要转给小李101个比特币，手头有2笔比特币：

□一笔100个比特币（老王给的）
□一笔50个比特币（挖矿）

- 101=支付（100+50）- 找零（49）

- 小李就可以继续支付给小张101个比特币

UTXO交易实例



- 小王要转给小李101个比特币，手头有2笔比特币：

  □一笔100个比特币（老王给的）
  □一笔50个比特币（挖矿）

- 101=支付（100+50）- 找零（49）

- 小李就可以继续支付给小张101个比特币

UTXO模型信息

Transaction View information about a bitcoin transaction

75eb73a617e1aaef1a187d9e19bdaee7c38476e1399e8cab8f95e9e5e83bd4b7

1KoFB5SQq3kPZ8z6KFt97aTEQW1fjs75SS 148mARsoudWiUdJi9sDoUa9tRtfogMGNdz	→	1NEFQJ6rQNnRaUtQwaxu8dV14McHMzFfC5 13MbEETZszRA2tF5s6MGxVtK4P5vw7X8W8

0.065773 BTC
0.00010771 BTC
0.06588071 BTC

Summary	
Size	436 (bytes)
Weight	1744
Received Time	2015-03-24 00:08:11
Included In Blocks	348915 (2015-03-24 00:12:26 + 4 minutes)
Confirmations	176377 Confirmations
Visualize	View Tree Chart

Inputs and Outputs	
Total Input	0.06598071 BTC
Total Output	0.06588071 BTC
Fees	0.0001 BTC
Fee per byte	22.936 sat/B
Fee per weight unit	5.734 sat/WU
Estimated BTC Transacted	0.065773 BTC
Scripts	Show scripts & coinbase

来源：blockchain.info

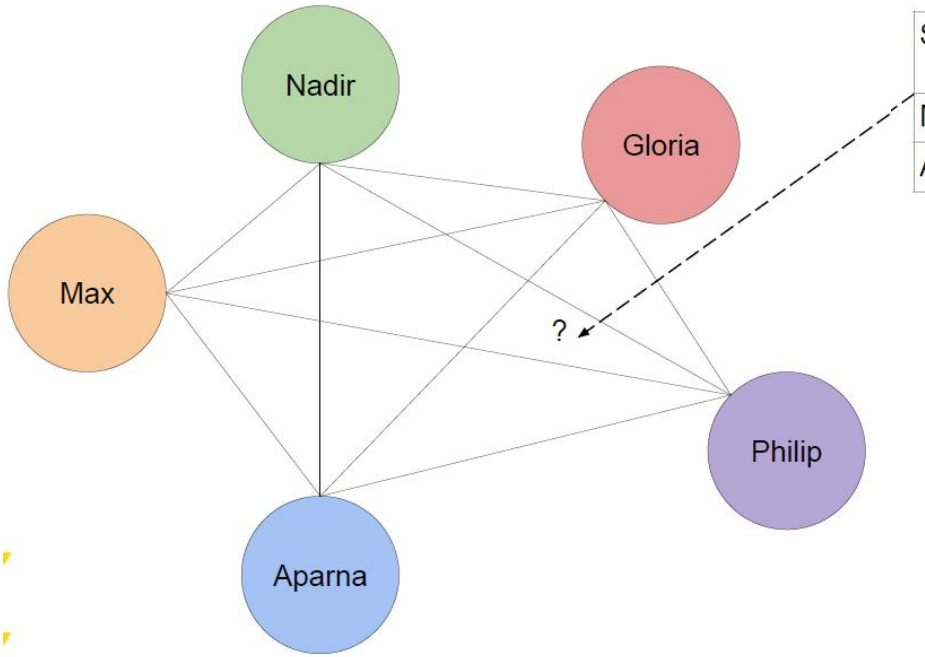
比特币共识机制

- ~~比特币身份确认~~ ~~身份确认~~
- ~~UTXO交易模型~~ ~~交易服务~~
- 链式交易信息记录 **记录管理**
- 工作量证明(POW) **信任规则**

RECORD
记录管理

记录：分布式数据库

- 在身份确认，交易有效后，如何存储交易记录?
- 如何保障交易账本的可溯性？

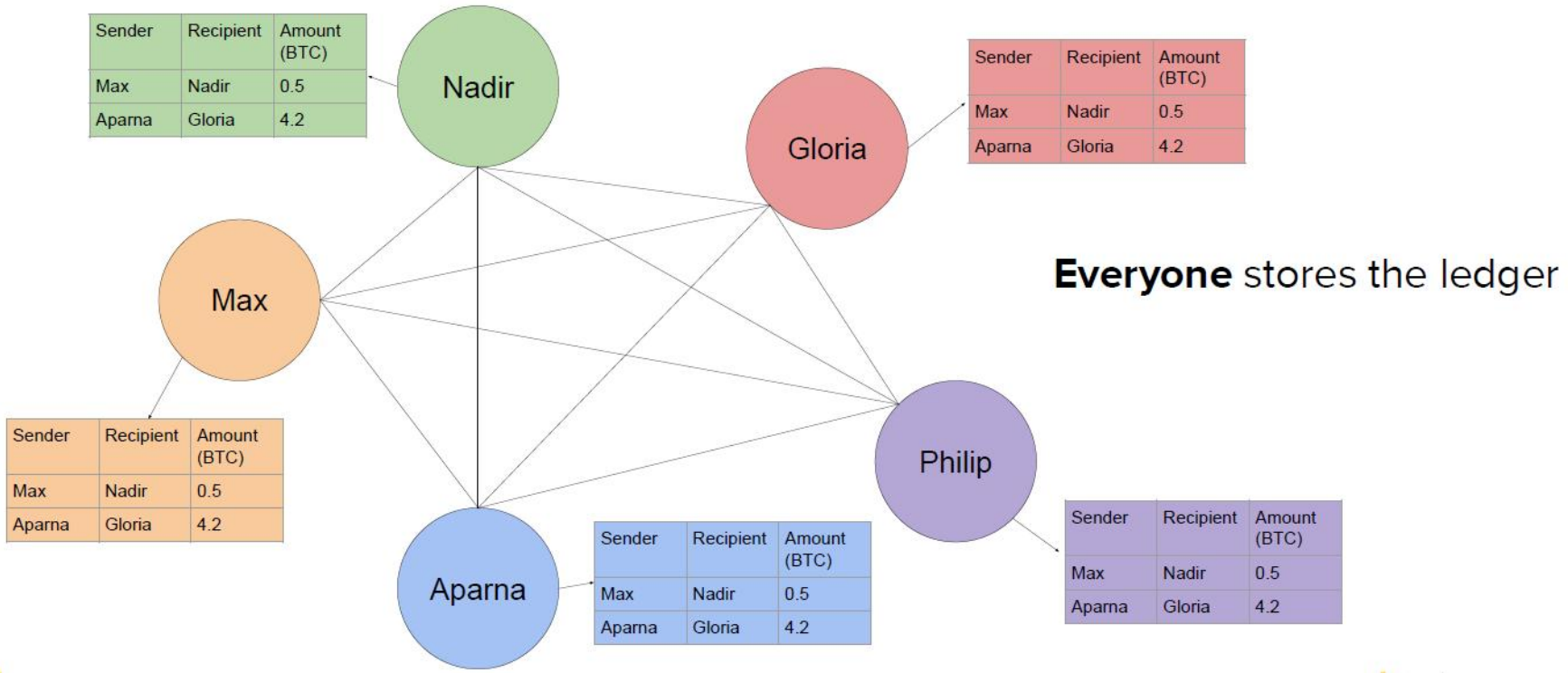


Sender	Recipient	Amount (BTC)
Max	Nadir	0.5
Aparna	Gloria	4.2

- —分布式数据库

记录：人人即银行

- 每个比特币网络上的节点都存储一个完整的账本



记录格式：区块链

- 每一段时间生成一个区块，区块里记录着这个时间段内的交易记录，区块与区块间按时间安全紧密地串联在一起，就成了区块链

Sender	Recipient	Amount (BTC)
Max	Nadir	0.5
Aparna	Gloria	4.2
Philip	Gloria	23
Max	Philip	3.2
Nadir	Aparna	0.3
Gloria	Philip	17

Sender	Recipient	Amount (BTC)
Max	Nadir	0.5
Aparna	Gloria	4.2

Sender	Recipient	Amount (BTC)
Philip	Gloria	23
Max	Philip	3.2

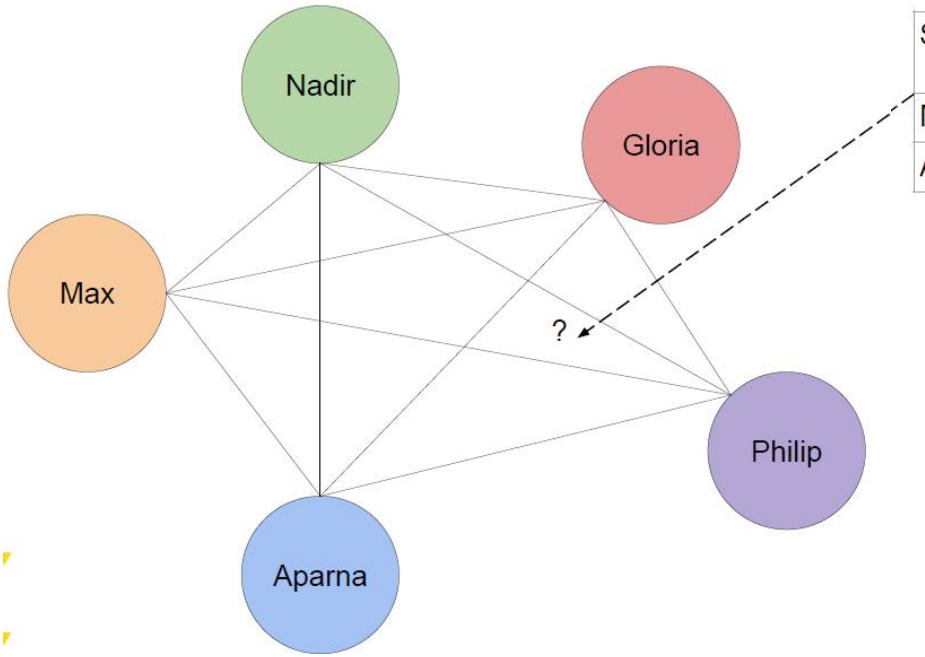
Sender	Recipient	Amount (BTC)
Nadir	Aparna	0.3
Gloria	Philip	17

- **区块形成及串连的规则就是共识机制**

RECORD
记录管理

记录：分布式数据库

- 在身份确认，交易有效后，如何存储交易记录?
- 如何保障交易账本的可溯性？

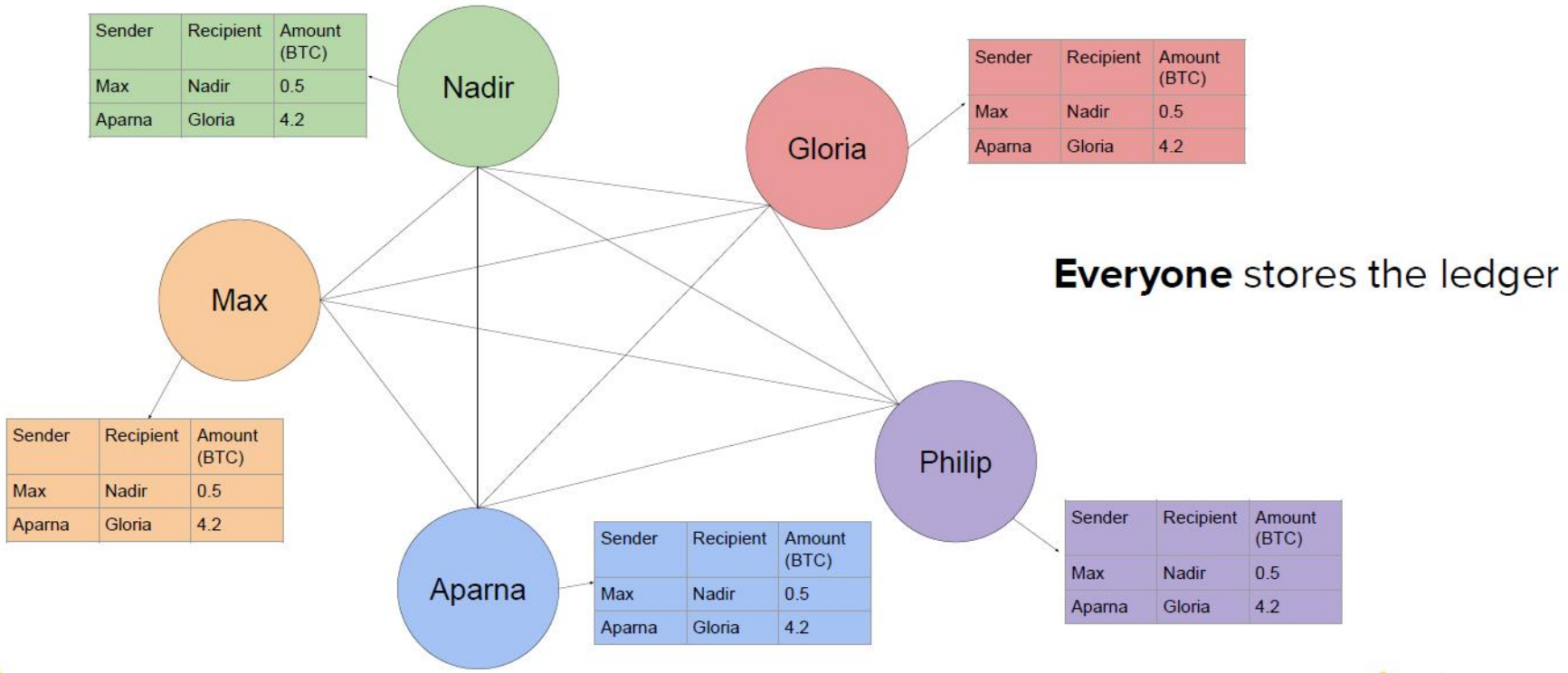


Sender	Recipient	Amount (BTC)
Max	Nadir	0.5
Aparna	Gloria	4.2

- —分布式数据库

记录：人人即银行

- 每个比特币网络上的节点都存储一个完整的账本



记录格式：区块链

- 每一段时间生成一个区块，区块里记录着这个时间段内的交易记录，区块与区块间按时间安全紧密地串联在一起，就成了区块链

Sender	Recipient	Amount (BTC)
Max	Nadir	0.5
Aparna	Gloria	4.2
Philip	Gloria	23
Max	Philip	3.2
Nadir	Aparna	0.3
Gloria	Philip	17

Sender	Recipient	Amount (BTC)
Max	Nadir	0.5
Aparna	Gloria	4.2

Sender	Recipient	Amount (BTC)
Philip	Gloria	23
Max	Philip	3.2

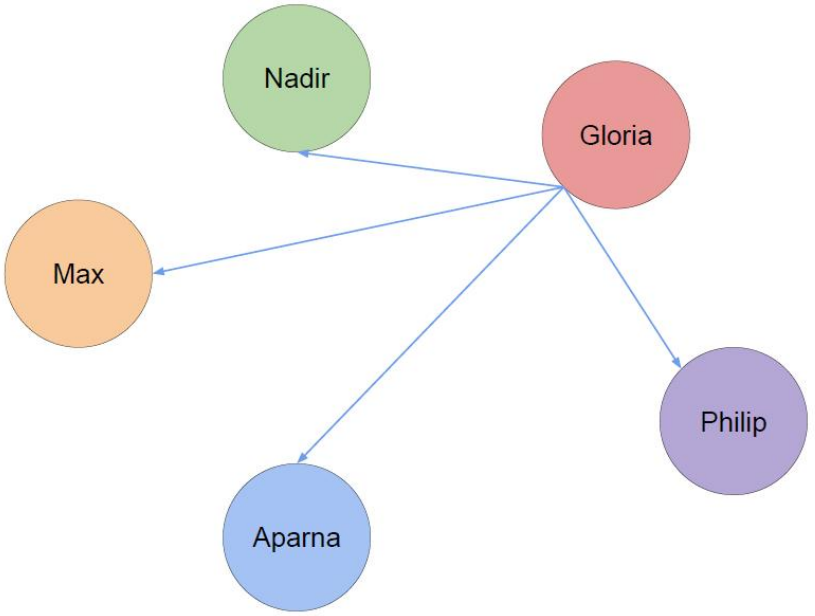
Sender	Recipient	Amount (BTC)
Nadir	Aparna	0.3
Gloria	Philip	17

- **区块形成及串连的规则就是共识机制**

AGREEMENT（CONSENSUS）
共识机制

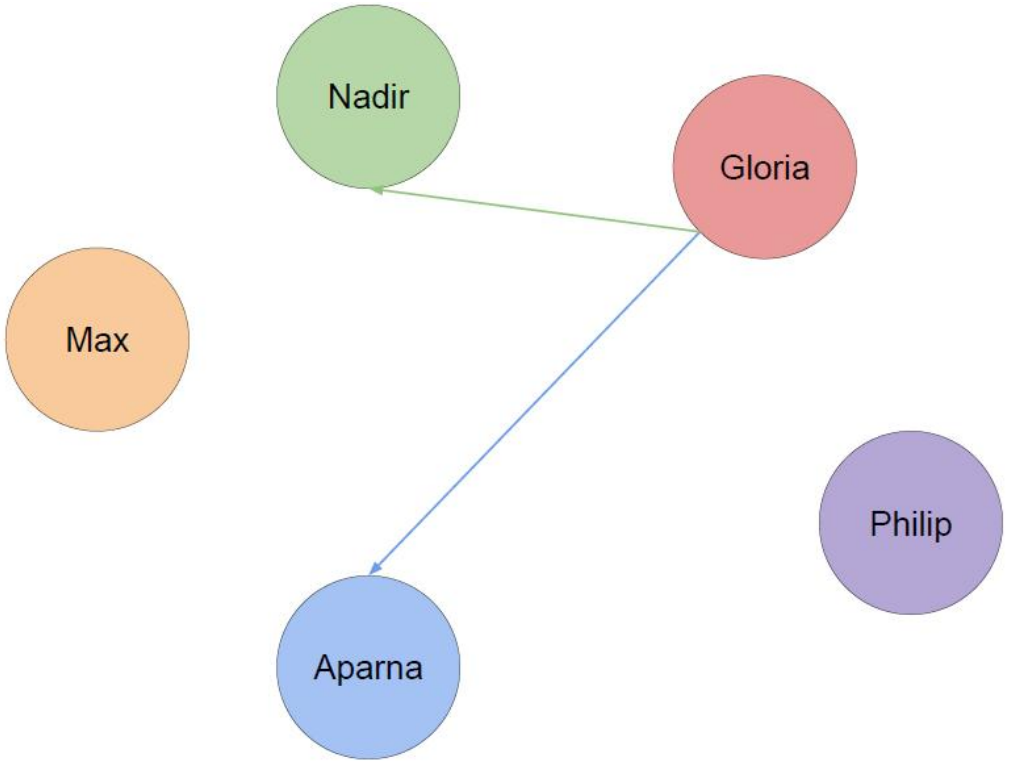
共识机制

- 在一般的公有区块链上，如何达成共识？
  - 如果每个区块链上的用户只要接收到有效记录后，都记录到自己账本，会发生什么



共识机制中常见的攻击

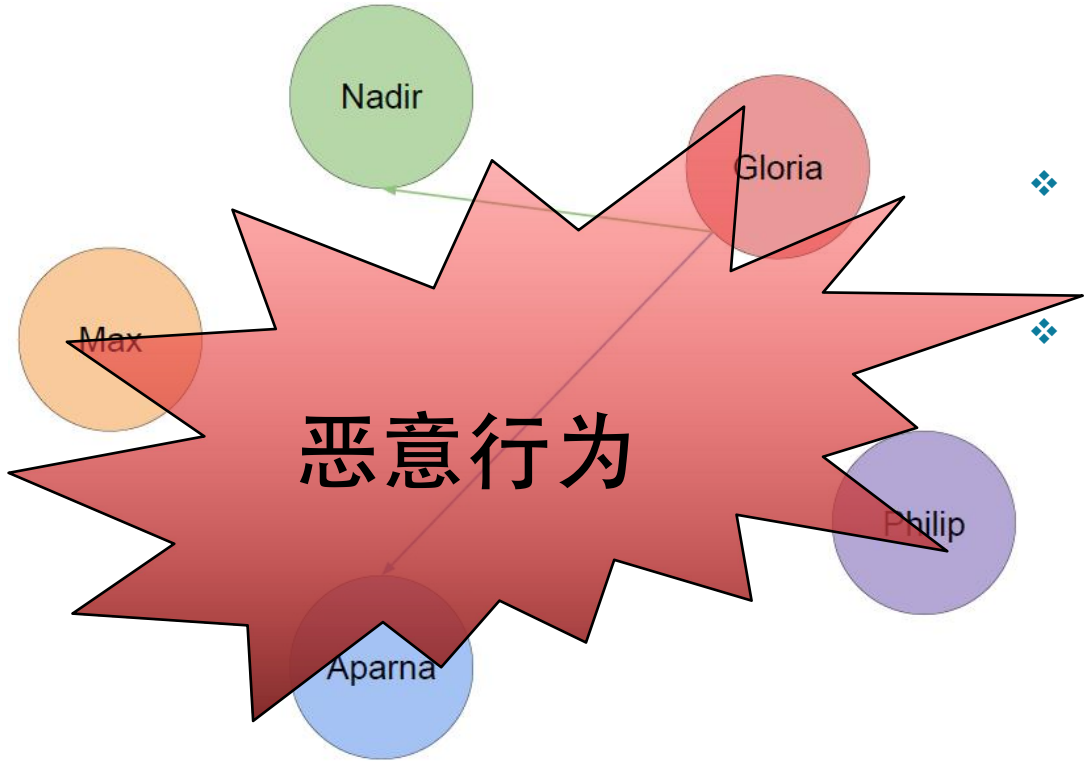
❖Double Spend Attack 双花攻击



- Gloria答应给Aparna 10个比特币，同时答应给Nadir 10个比特币.....但是她总共只有10个比特币
- Gloria的行为就是双花攻击
- **将导致每个节点记录的账本信息不一致**

共识机制常见的攻击

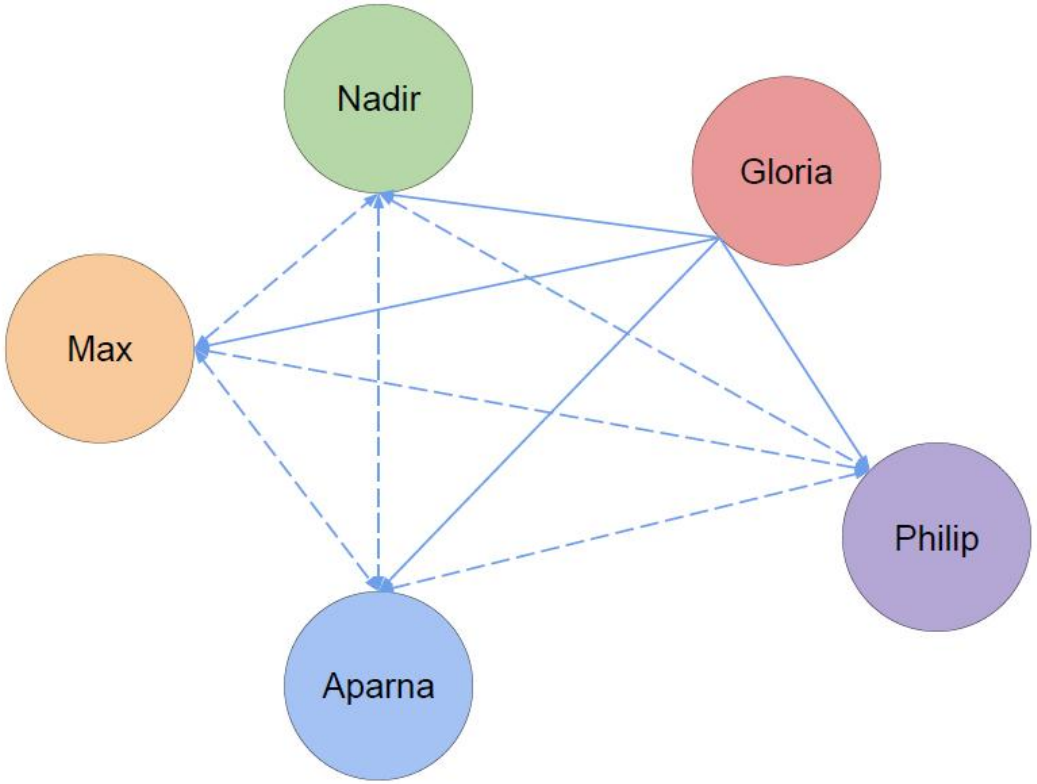
❖Double Spend Attack 双花攻击



- Gloria答应给Aparna 10个比特币，同时答应给Nadir 10个比特币.....但是她总共只有10个比特币
- Gloria的行为就是双花攻击
- **如何保证这种 独立的 记录过程不存在这种恶意行为？**

共识机制常见的攻击

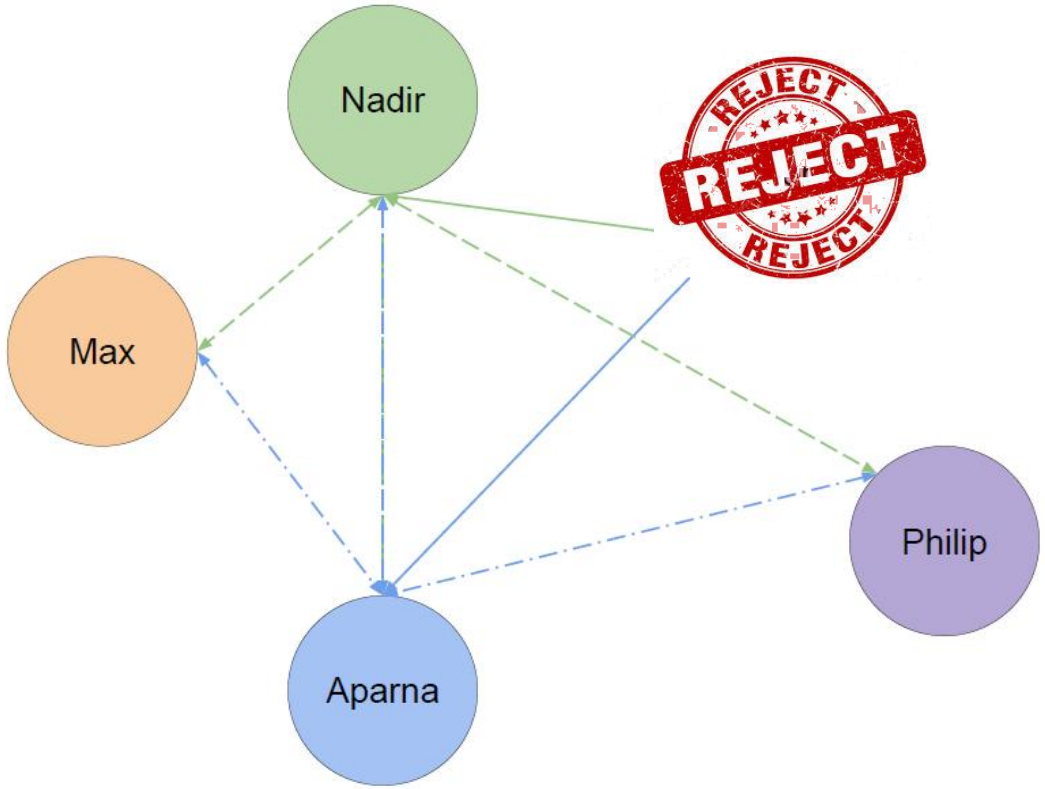
❖同等验证



- ❖ **取代每个用户孤立的决定**
  - 提交者向其他用户提交一条交易信息
  - 其他用户进行投票
  - 当获得一定数目投票后，大家同意将交易信息进行保存
- ❖ **保证所有节点存储相同的交易账单**

共识机制常见的攻击

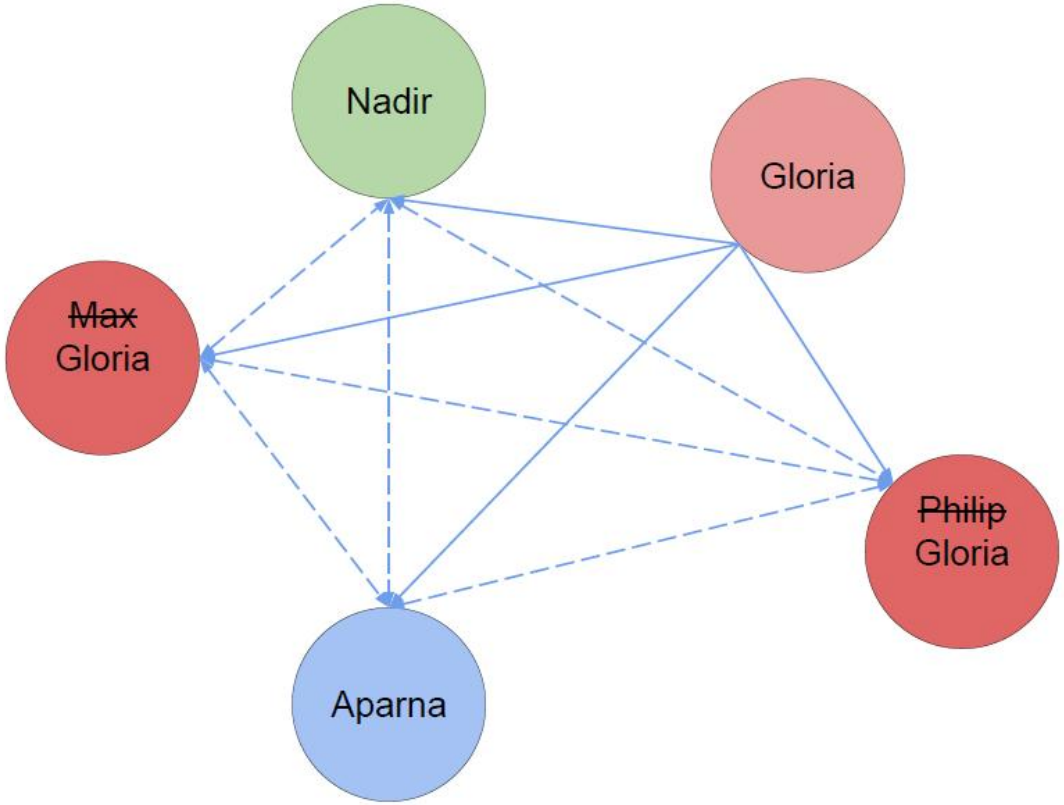
❖拒绝双花



- 现在，当Gloria准备进行双花时，会被其他用户拒绝接受该笔交易。
- 节点在观测到多笔交易使用同一笔比特币时，会对Gloria的提案投出反对票。

共识机制常见的攻击

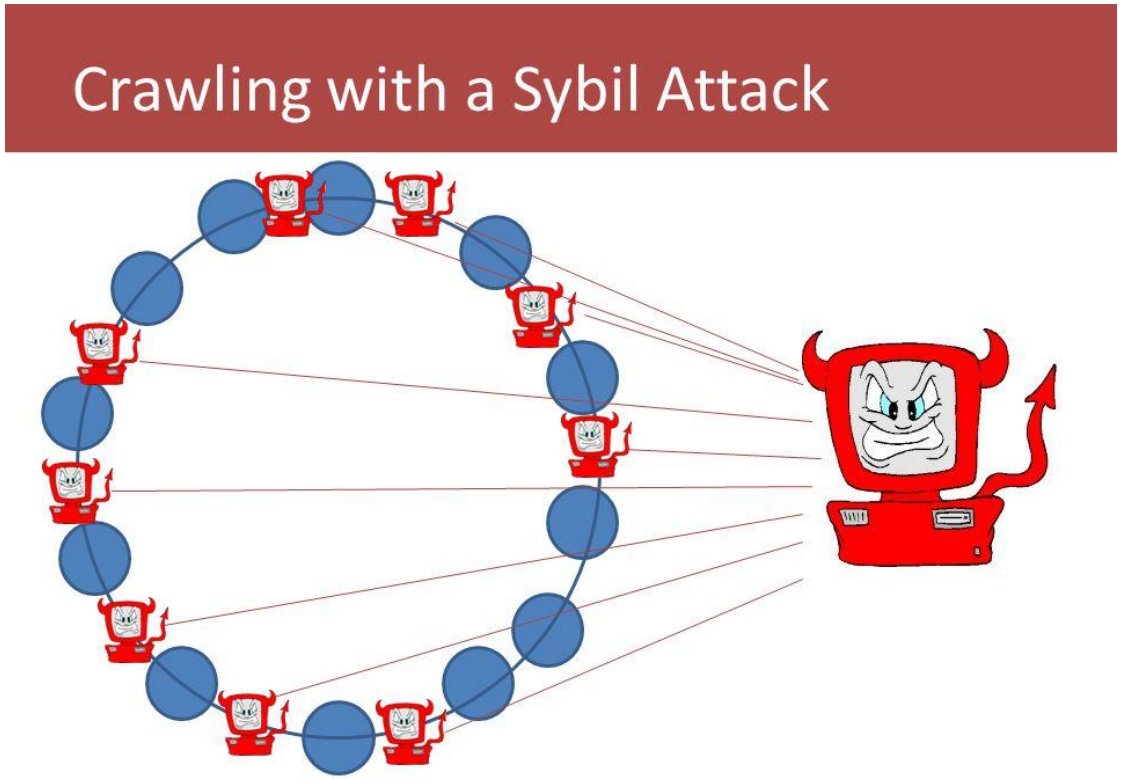
❖Sybil Attack 多重身份攻击/女巫攻击



- ❖ 比特币作为无中心登记的匿名服务
- 创建多重身份代价极低
- 多重身份意味着多重的投票权利

共识机制常见的攻击

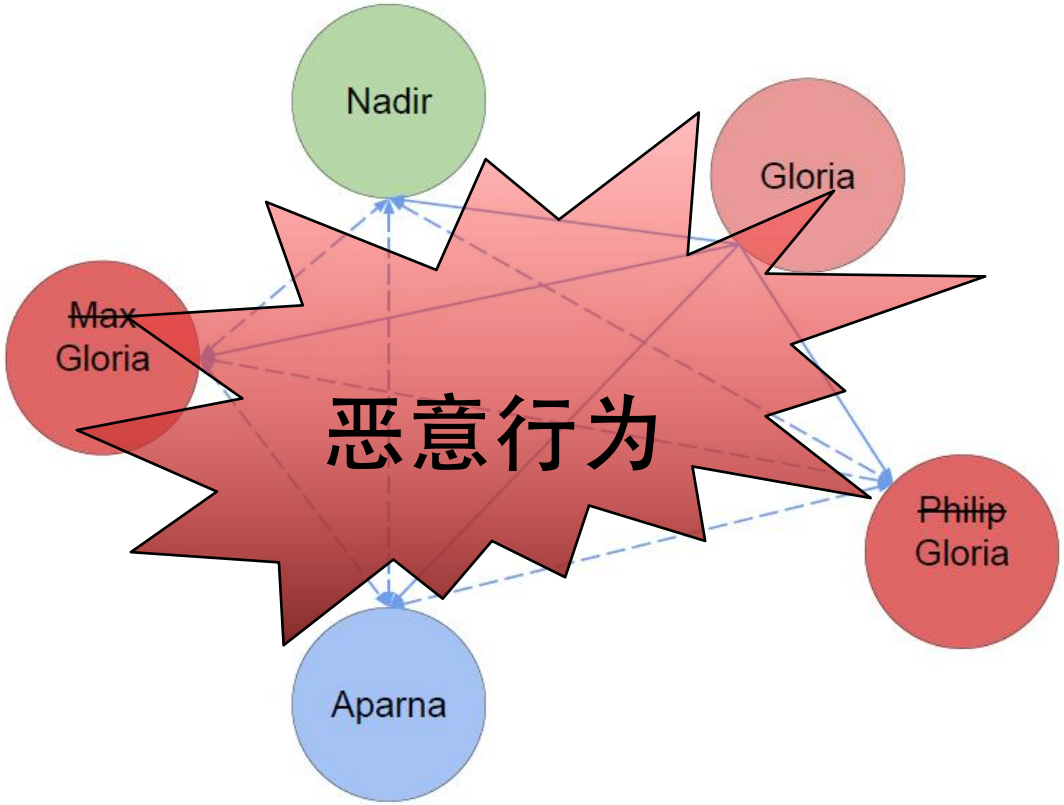
❖Sybil Attack 多重身份攻击/女巫攻击



- 比特币作为无中心登记的匿名服务
- 创建多重身份代价极低
- 多重身份意味着多重的投票权利

共识机制常见的攻击

❖Sybil Attack 多重身份攻击

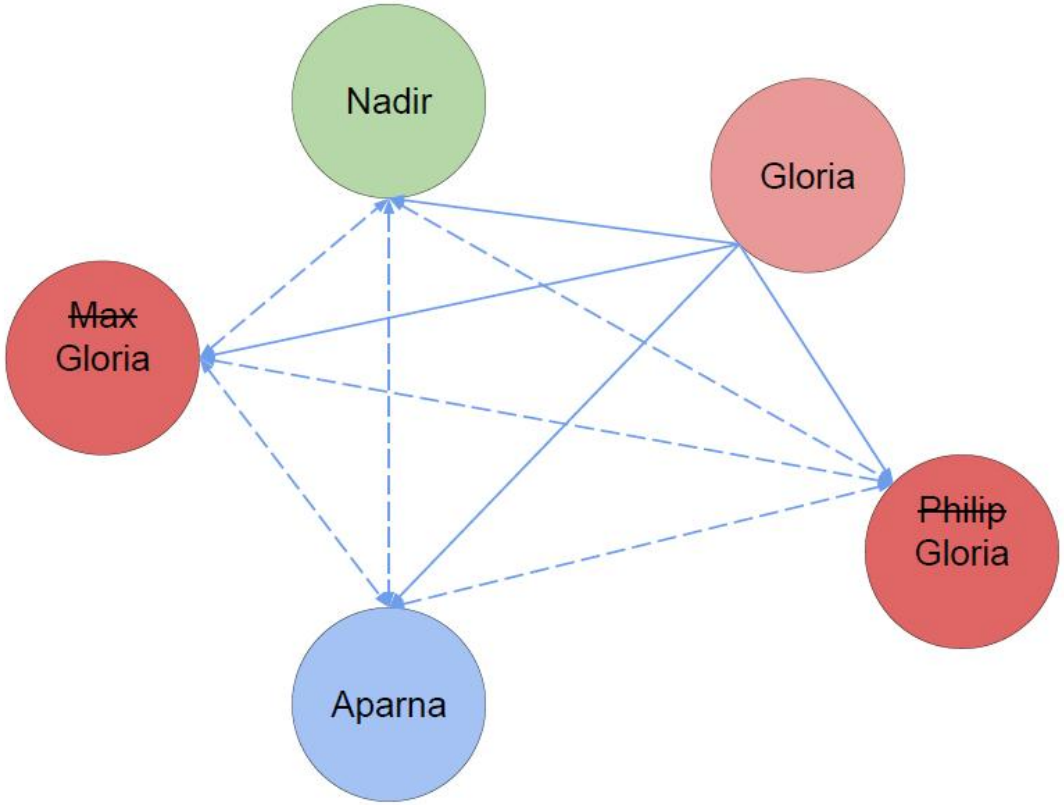


- ❖ 比特币作为无中心登记的匿名服务
  - 创建多重身份代价极低
  - 多重身份意味着多重的投票权利

- ❖ Gloria可以实行多重身份攻击，从而允许她的双花行为

共识机制常见的攻击

❖Sybil Attack 多重身份攻击



- ❖ 原因：投票几乎没有成本！
- ❖ 取代用身份投票的机制，我们采用资源成本进行投票
- ❖ 提高作恶的代价！

工作量证明

POW应用

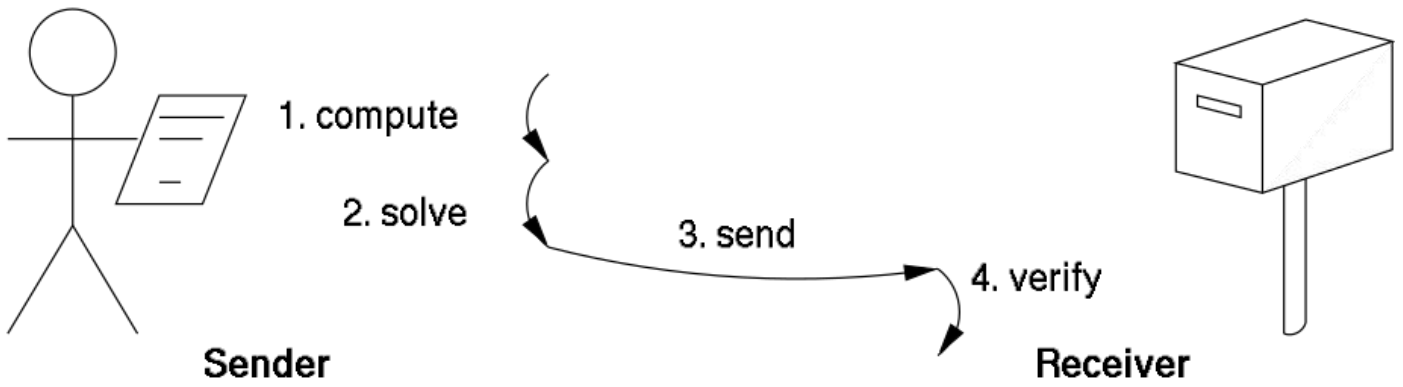
- **工作量证明（POW），**此一概念最早由 Cynthia Dwork 和 Moni Naor 于 1992 年的学术论文提出，而工作量证明一词则是在 1999 年由 Markus Jakobsson 与 Ari Juels 所发表

- POW 最早被用于阻止拒绝服务攻击（DDOS）、反垃圾邮件等一些服务滥用的经济对策

POW应用

- **第一个POW 应用**是1996年 Adam Back 开发的 "Hashcash" 应用，它采用工作量证明共识机制来过滤垃圾邮件，微软也将其应用在 Hotmail, Exchange, Outlook 等电邮服务上。

- 具体做法是要求所有收到的邮件都使用强 PoW 附件（邮票），比如Receiver向所有想发送电子邮件的 Sender 都分发一个"标准质询"。



- 此系统使得垃圾邮件发送者在大量发送邮件时在经济成本上不可行。

POW in Life

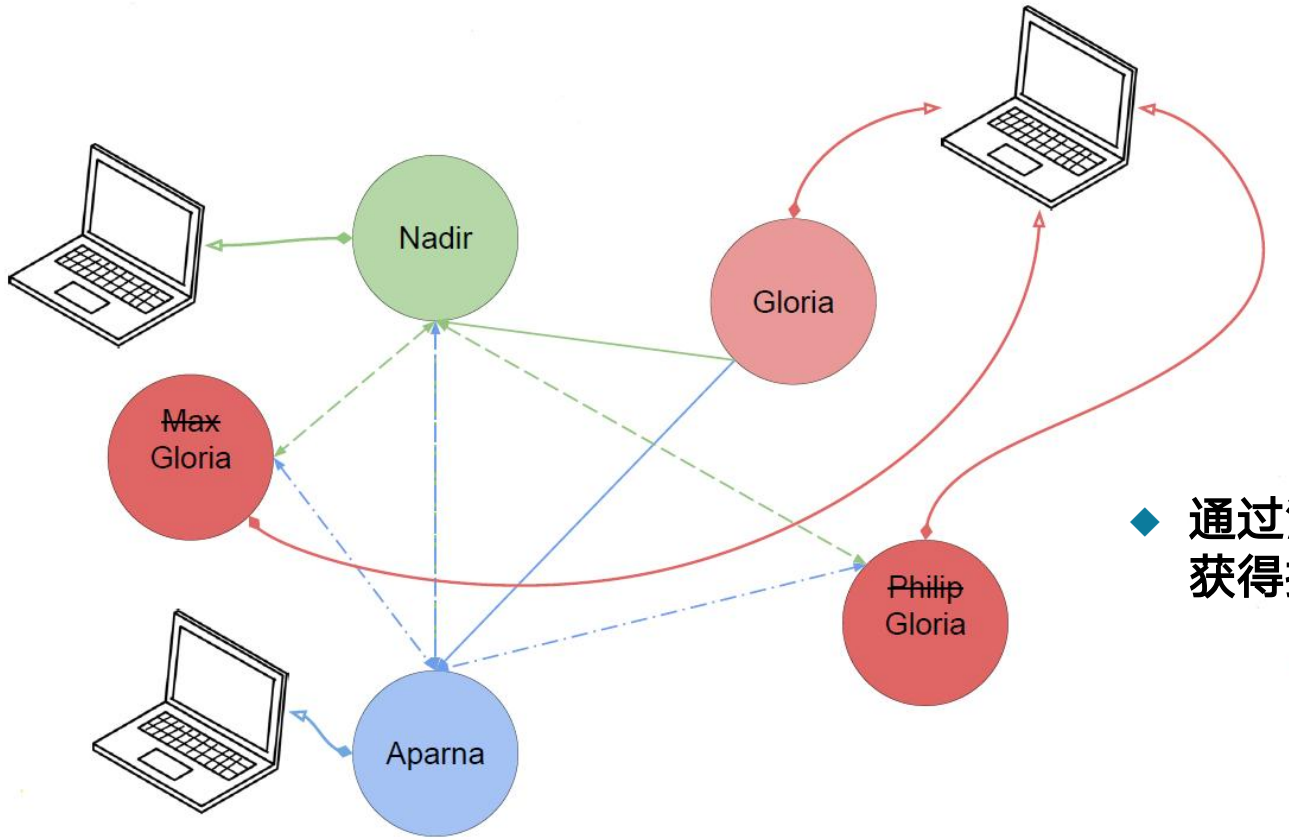
- 你想到一家公司去工作，这家公司会让你先实习一段时间，公司会考量你的实习质量来决定是否录用你，这段实习的时间就是你的工作量证明。

- 某男女谈恋爱，某女为了考验其的忠心，便让某男去给她买套房，某男便用光老爸老妈所有的钱买了套房，某女可以轻易的检验某男是否购房，这个行为也可以看做是工作量证明。

✓ **不容易完成**
✓ **容易验证**

工作量证明

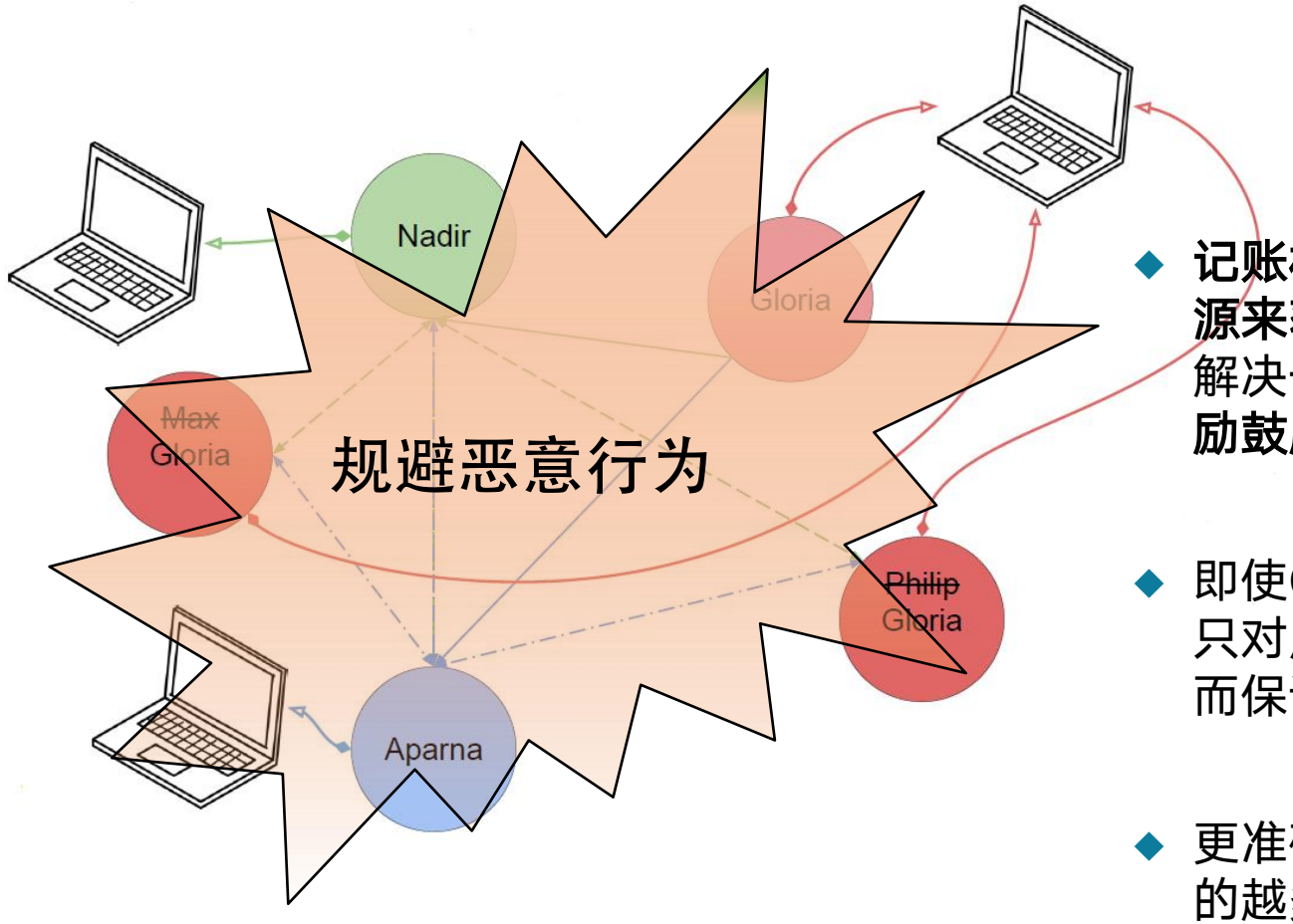
- Proof-of-Work 工作量证明



- **通过消耗资源解决一个问题获得投票资格（即记账权）**

工作量证明

- Proof-of-Work 工作量证明



- **记账权必须通过花费计算资源来获得**，比如说通过蛮力解决一个问题；**通过记账奖励鼓励投入资源**
- 即使Gloria有多个身份，也只对应到单个计算资源，从而保证记账的公正性
- 更准确来说像买彩票。投入的越多（花费越多)，中奖几率越大（成功记账）

“工作量”争夺记账权

■整个争夺记账的过程就是挖矿的过程，也就是比特币发行的过程

■ “挖矿”争夺记账权奖励

- 记账有利润：比特币奖励 + 交易手续费
- 很多人争夺记账权
- 通过付出计算量解决一个难题，谁先解决谁获得记账权
- 坏人作恶的成本变高

POW 挖矿

- **挖矿：是参与维护比特币网络的节点，通过协助生成新区块来获取一定量新增的比特币的过程**

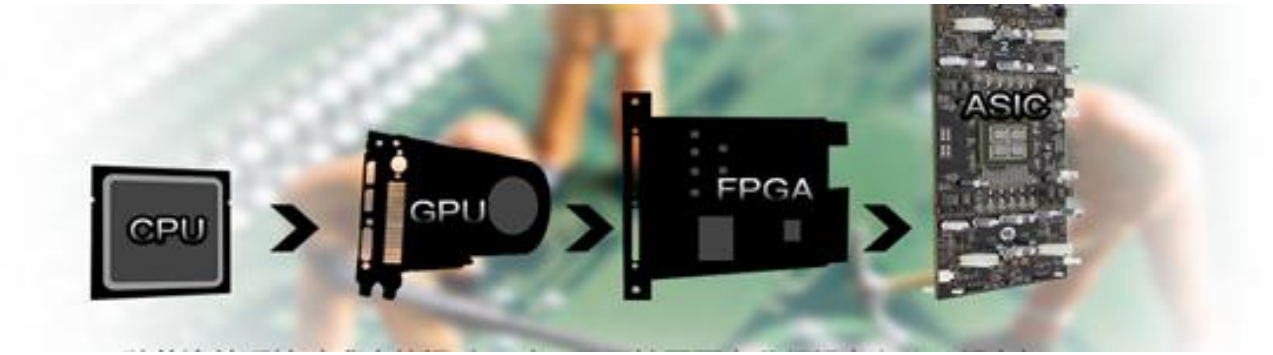
- 每 10 分钟左右生成一个不超过 1 MB 大小的区块（记录了这 10 分钟内发生的验证过的交易内容），串联到区块链尾部，每个区块的成功提交者可以得到系统 6.25 个比特币的奖励（该奖励作为区块内的第一个交易，一定区块数后才能使用），以及用户附加到交易上的支付服务费用

- **注**：每个区块的奖励最初是 50 个比特币，每隔 21 万个区块自动减半，即 4 年时间，最终在2140年比特币总量稳定在 2100 万个。因此，比特币是一种通缩的货币

POW 算力

由于 Hash 难题在目前计算模型下需要大量的计算，这就保证在一段时间内，系统中只能出现少数合法提案。反过来，能够提出合法提案，也证明提案者确实已经付出了一定的工作量。这也保障了，如果有人尝试恶意破坏，需要付出大量的经济成本

❖普通的 CPU（2009 年）、到后来的 GPU（2010 年） 和 FPGA（2011 年末）、到后来的ASIC 矿机（2013 年年初，目前单片算力已达每秒数百亿次 Hash 计算）、再到现在众多矿机联合组成矿池（知名矿池包括 F2Pool、BitFury、BTCC 等）



❖截止1/5/2018, 全网的算力已超过每秒2.6*10^18 次 Hash 计算，超过世界500强超级计算机算力总和的100倍!

总结：无中心网络需要何种工作量证明？

■**难题设计必须满足如下条件：**

- **不容易完成**（表明需要工作量）
- **容易验证**（其他节点可以快速确认确实付出了工作量）
- **工作过程公平**（任何节点没有完成工作的捷径）
- **具有随机性**（能力越强，只能保证率先完成概率越大）

第3课内容的 总结

回答了灵魂之问的第1问: 区块链是什么

- 1.1 区块链背景与现状
- 1.2 区块链基本概念
- 1.3 区块链技术原理