北京大学
PEKING UNIVERSITY

区块链课程

孙惠平
sunhp@ss.pku.edu.cn

北京大学 软件与微电子学院
School of Software and Microelectronics, Peking University

北京大学
PEKING UNIVERSITY

PART 第二章

区块链基础

目 录

CONTENTS

第一节

密码基础

01 密码学

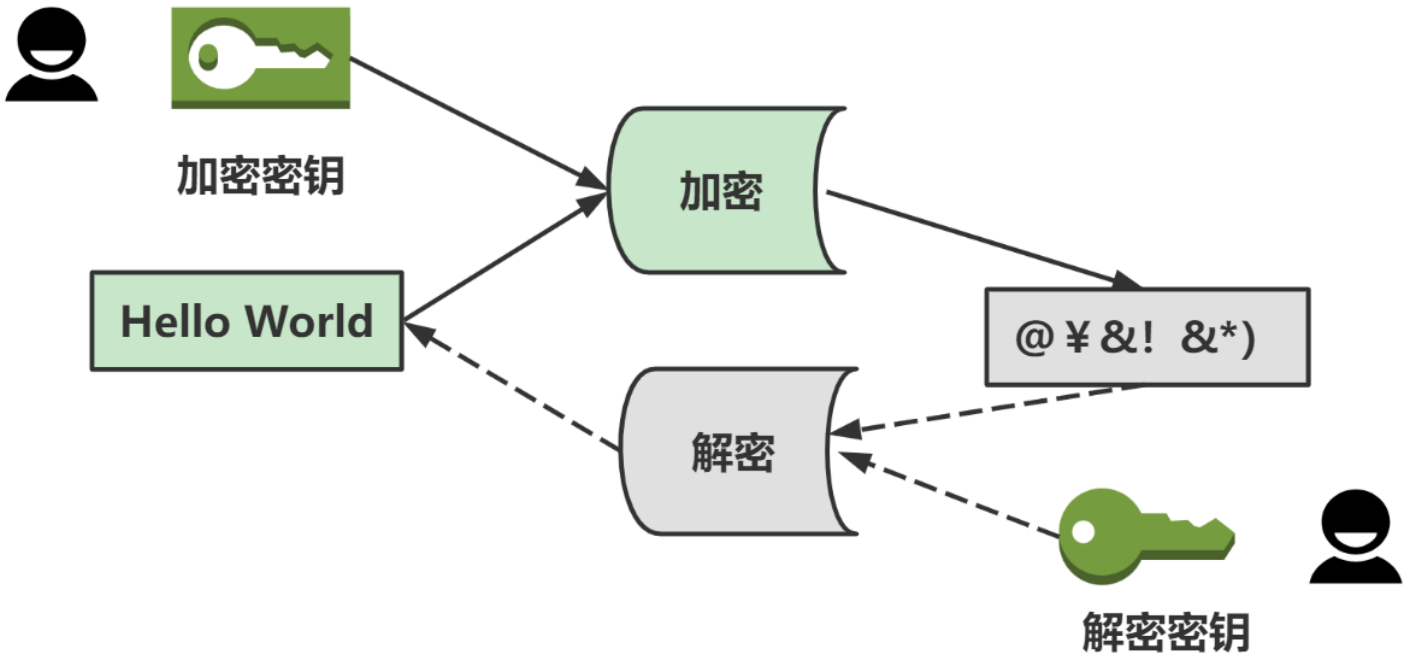
- 密码技术在区块链中的应用十分广泛。从区块链的**用户注册、用户交易、区块存储，密码技术**每时每刻都在保护着这个庞大系统的正常运转。

加密

数据加密就是对明文文件或数据按某种算法进行处理，使之称为密文。

解密

数据解密是在得到密文后，用相应的密钥将密文转化为明文。加密的逆过程称为解密。

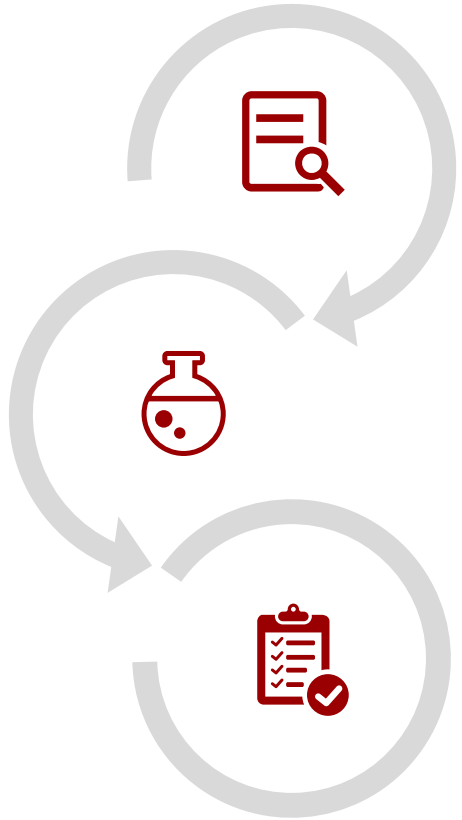


01 密码学

- **密码学要求：密码学作为信息安全的关键技术，需要制定安全的目标，主要包括：保密性（confidentiality）、完整性（integrity）、可用性（availability），三个性质简称CIA；**

保密性：指只有授权的用户可以获取信息。即信息能确保只有被合法用户访问，而不会被泄露给非授权的用户、实体或者过程。信息不仅能被授权的用户正常访问，并防止泄露。

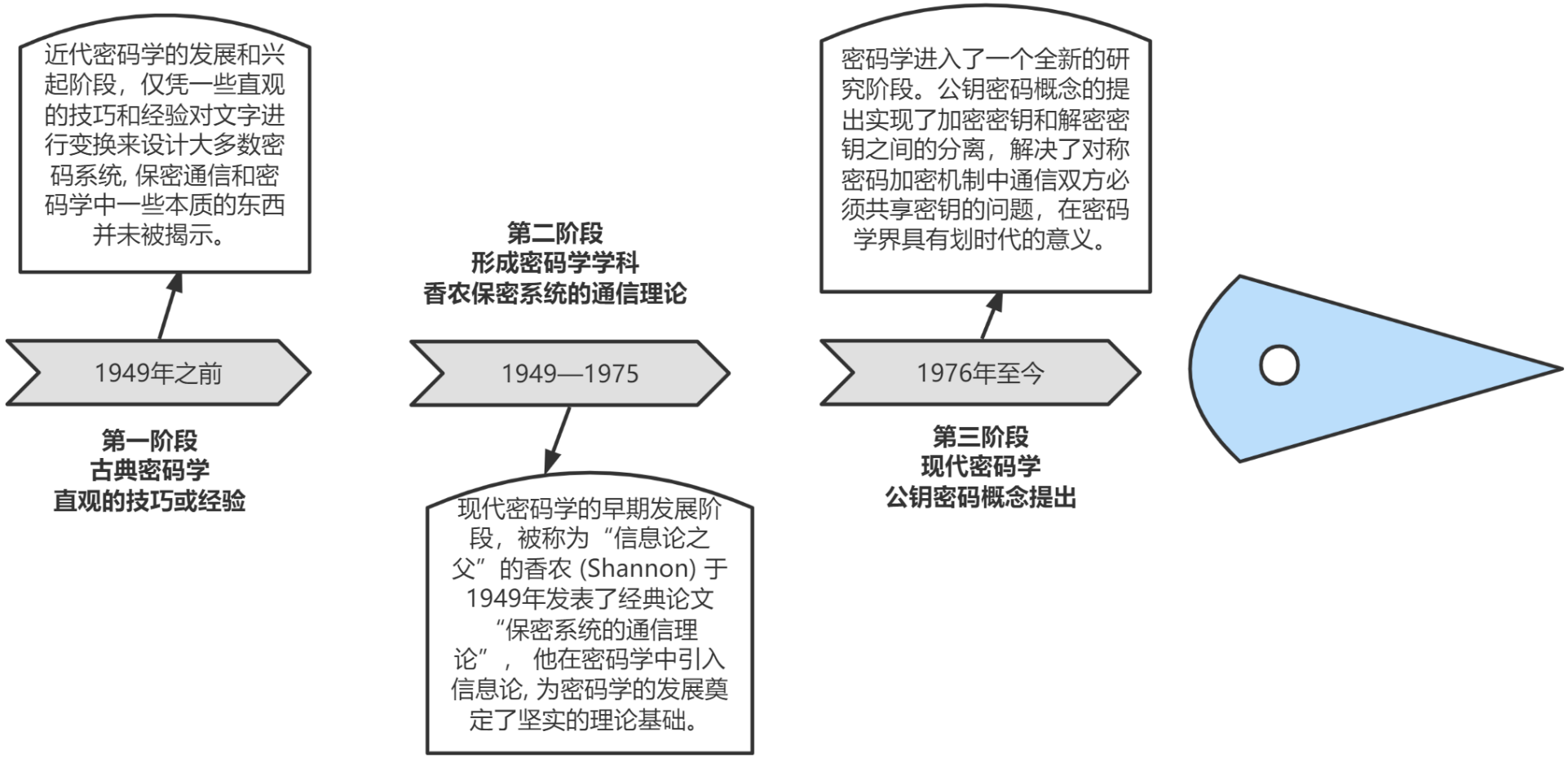
完整性：指信息在输入和传输的过程中，不被非法授权修改和破坏，保证数据的一致性。完整性是一种面向信息的安全性，确保信息可以正确生成、正确存储、正确传输。



可用性：指保证合法用户对信息和资源的使用不会被不正当地拒绝。在适当的时候可以由合法用户访问所有资源。可用性是信息系统面向用户的安全性能，向用户提供服务是信息系统最基本的功能

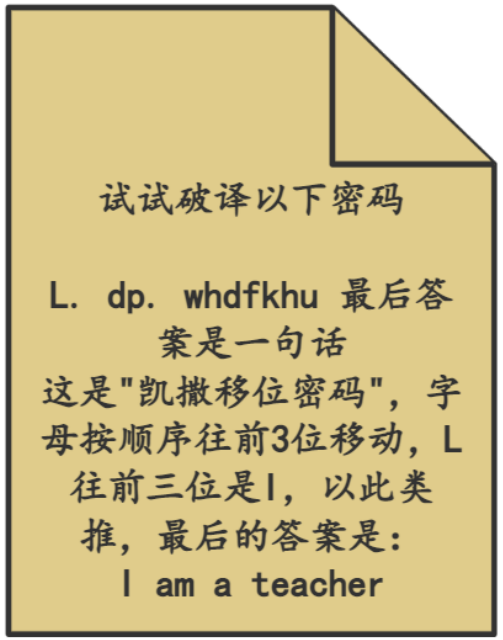
北京大学 PEKING UNIVERSITY

- **密码学历史：古典密码学、密码学学科、现代密码学**



02 古典密码学

- **密码学源头：密码学这个词源自希腊语，意思为隐藏和书写。是研究如何隐密地传递信息的学科。其目的是为两人在不安全的信道上进行通信而不被破译者获取通信的内容。**



国外起源

相传最早使用密码捆在木棒上方法是公元前 5 世纪的斯巴达人，送信人先绕棍子卷一张纸条，然后把要加密的信息写在上面，接着打开纸送给收信人。后来，罗马的军队用凯撒密码来进行通信。

国内起源

远在周代姜太公与周武王在军事作战时，使用“隐符”进行保密通信。所谓“隐符”，就是双方事先约定，不同长度的木板、竹节代表不同含义。

北京大学 PEKING UNIVERSITY

- **古典密码学算法可以分为代换密码和置换密码两种。**

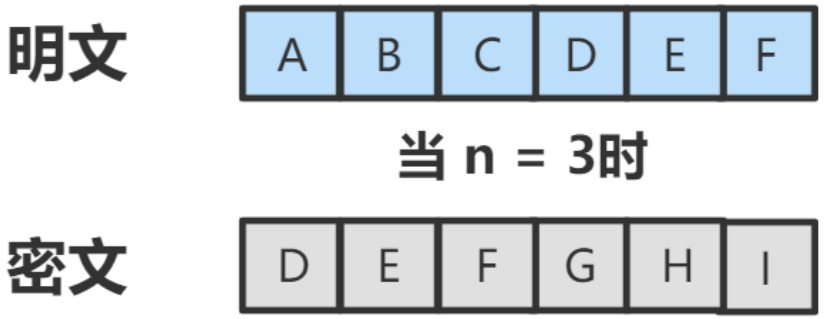
代换密码

代换技术是指使用一个一一映射代换表（只能一对一，不能一对多或者多对一）建立一个**替换规则**，将明文中的字符替换成相应其他字符。代换密码又分为单表代换密码和多表代换密码。单表代换即只按照映射关系对明文进行逐位代换，例如移位密码、凯撒密码、仿射密码

替换规则

$$E_n(x) = (x + n) mod 26$$

$$D_n(x) = (x - n) mod 26$$



- **古典密码学算法可以分为代换密码和置换密码两种。**

置换密码

置换密码算法的原理是不改变明文字符，只将字符在明文中的**排列顺序改变**，从而实现明文信息的加密。置换密码有时又称为换位密码。置换密码有列置换和周期置换。

03 对称密码学

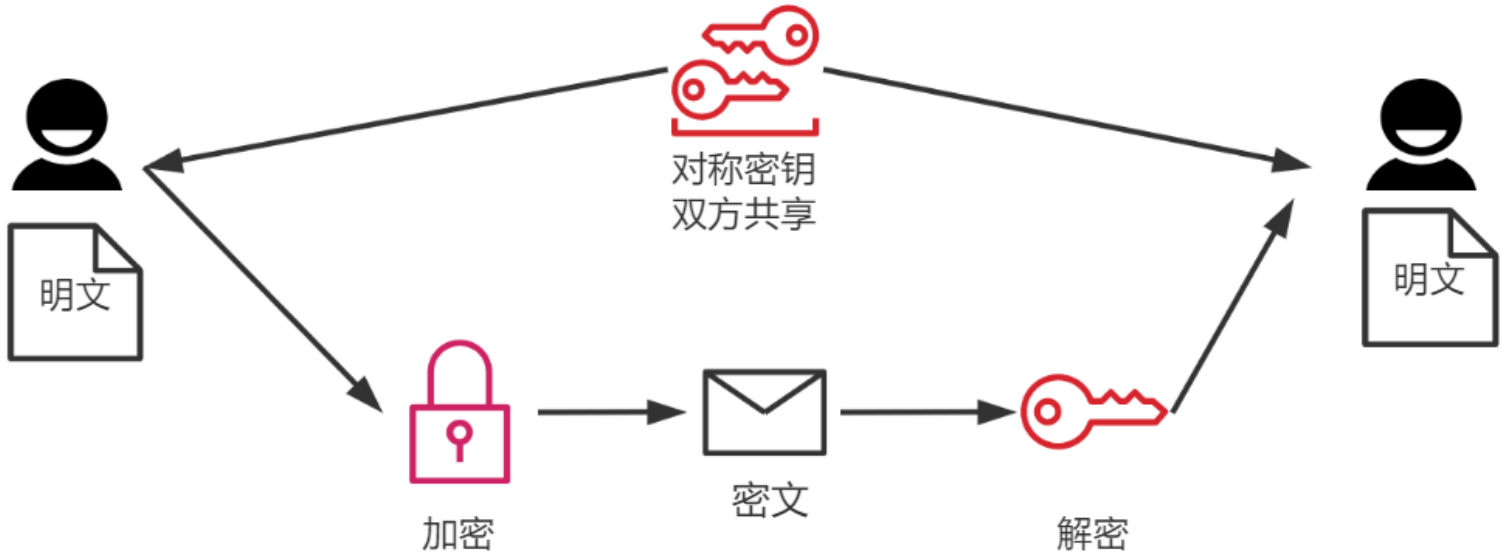
- **密码学中分为对称密码学和非对称密码学，对称密码体制指加密和解密使用相同密钥的加密算法。对称加密算法的特点是算法公开、计算量小、加密速度快、加密效率高。**

加密过程

- 发送方A选择一种加密算法，得到一个密钥。
- 发送方A对需要发送的数据加密。

解密过程

- 发送方将加密后的密文发送给接收方B。
- 发送方A将密钥交付给接收方B，B通过密钥解密。



北京大学 PEKING UNIVERSITY

- **对称密码有两种类型：流密码和分组密码。数据加密标准（DES，Data Encryption Standard）、高级加密标准（AES，Advanced Encryption Standard）是分组密码的典型示例。**

流密码

流密码也可以称为**序列密码**，就是明文和密钥的长度一致，一一进行异或运算，可以得出密文，实现简单，便于硬件实施，加解密处理速度快，但是扩展性较低。流密码基于内部状态生成密钥流的连续元素，如果状态独立于明文或密文消息而改变，称为同步流密码，若根据以前的密文数字更新状态则称为自同步流密码。

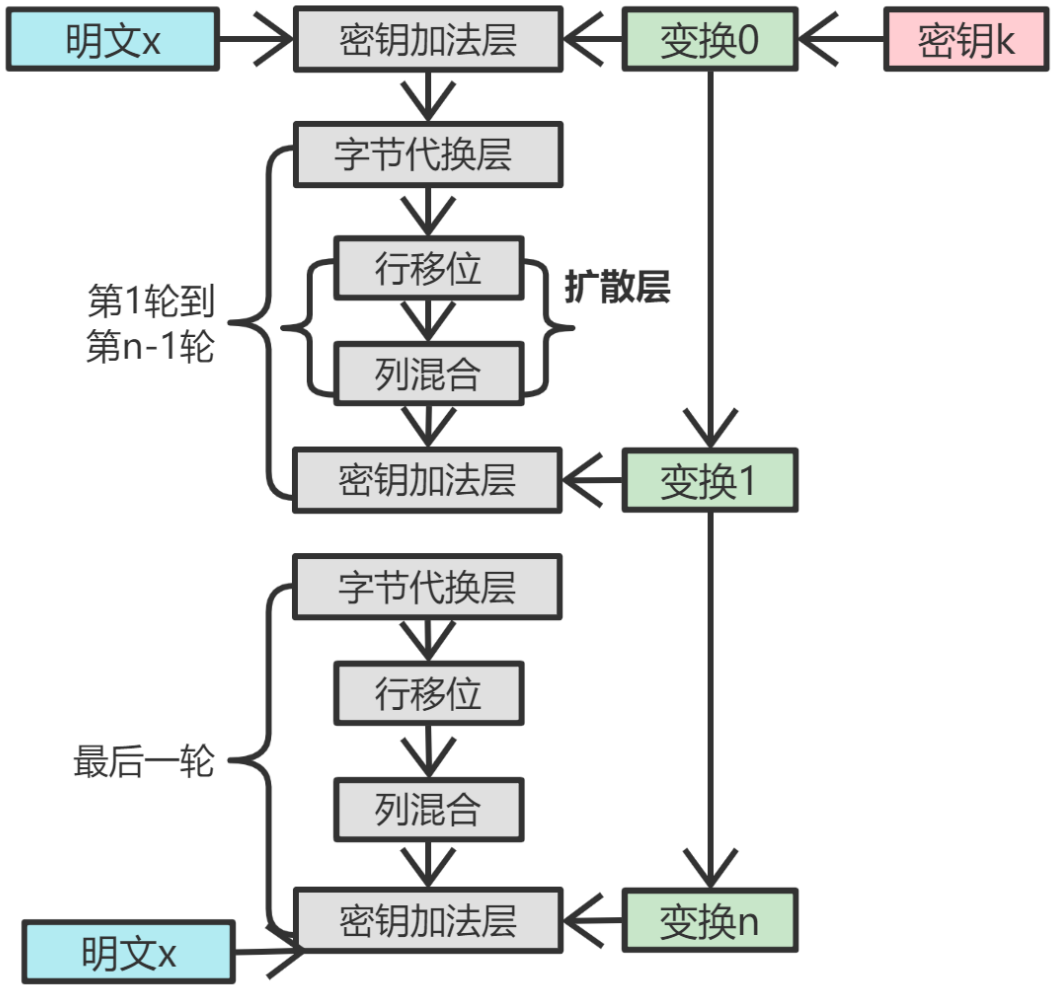
分组密码

分组密码是将明文消息编码后的数字序列，**划分成长度相等的组**，每组分别在密钥的控制下变换成等长的数组数字序列，当明文的长度超过分组密码的分组长度，就需要对分组密码进行迭代，以便加密，优点是明文信息有良好的扩展性，不需要密钥同步，有较强的适用性。

04 AES算法

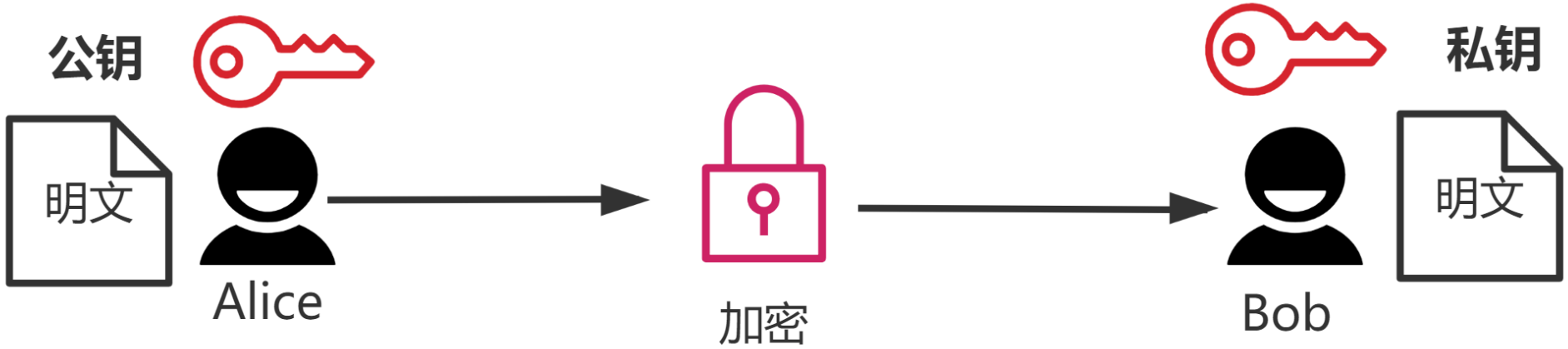
- **AES是高级加密标准**，它的出现主要是为了取代DES加密算法的，因为**DES算法的密钥长度是56位**，因此算法的理论安全强度是2的56次方，但随着计算机的飞速发展，计算计的处理能力越来越强，人们对安全性的需求也与日俱增。AES算法主要有**四种操作处理。**

1. **密钥加法层**
2. **字节代换层**
3. **行位移层**
4. **列混淆层**



05 非对称密码学

- **非对称密码学又称公钥密码学。对称加密方式存在的问题：每一对通信节点都需要事先共享密钥K，对网络中的一个节点来说，与若干个其他节点通信时，管理若干个密钥K并不是一件容易的事情。**



生成密钥

- Bob使用**密钥生成算法**生成**公钥/私钥对**，他公开公钥供需要与他通信的人使用。

加密过程

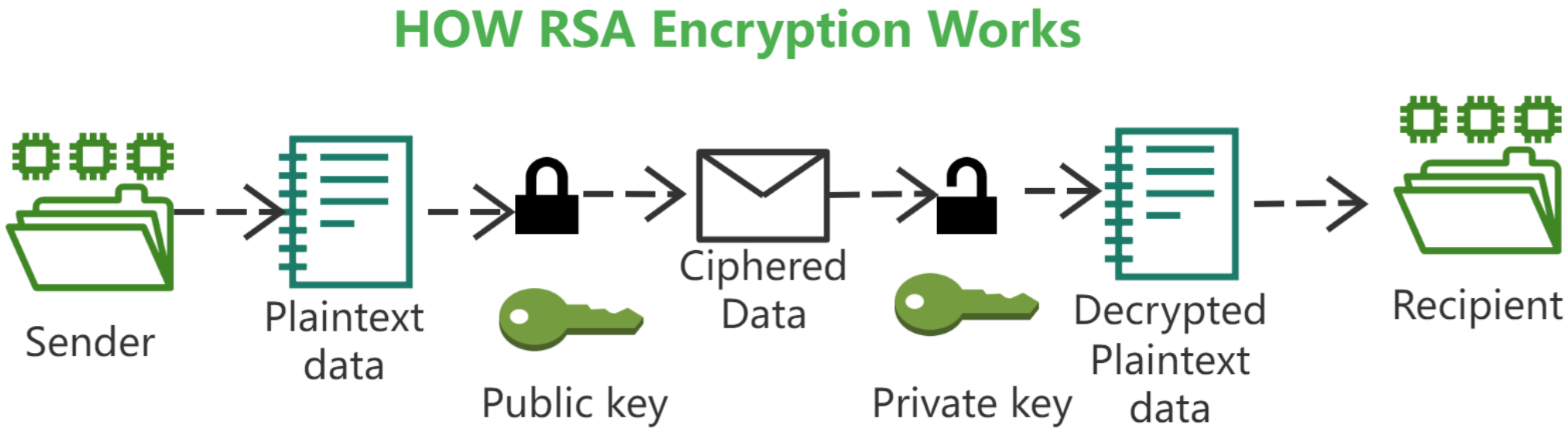
- Alice使用Bob**公钥利用加密算法**加密明文，生成密文，然后将密文发送到不安全信道中。

解密过程

- Bob**从不安全信道中接收到密文**，然后私钥利用**解密算法**解密出明文。

06 RSA算法

- RSA算法是一种**典型的公钥加密算法**。由三位著名的密码学家Ron Rivest、Adi Shamir和Leonard Adleman于1977年提出的，“RSA”这个名称就是他们三人姓氏开头字母拼在一起。



可靠性

- 源于寻找两个大质数比较简单、但是分解它们乘积的质因子是一件十分困难的事情。
- 说除了暴力破解没有更好的解决方案。攻击者要想破解RSA算法，需要分解大整数，该过程是一个几乎不能完成的过程，因此RSA算法是非常安全的。

06 RSA算法

- **RSA算法步骤**

选择两个不同的大素数p和q；

计算乘积$n = pq$和$\Phi(n) = (p-1)(q-1)$；

选择大于1小于$\Phi(n)$的随机整数e，使得$\gcd(e, \Phi(n)) = 1$；注：gcd即最大公约数。

计算d使得$de = 1 mod\ \Phi(n)$；注：即$de\ mod\ \Phi(n) = 1$。

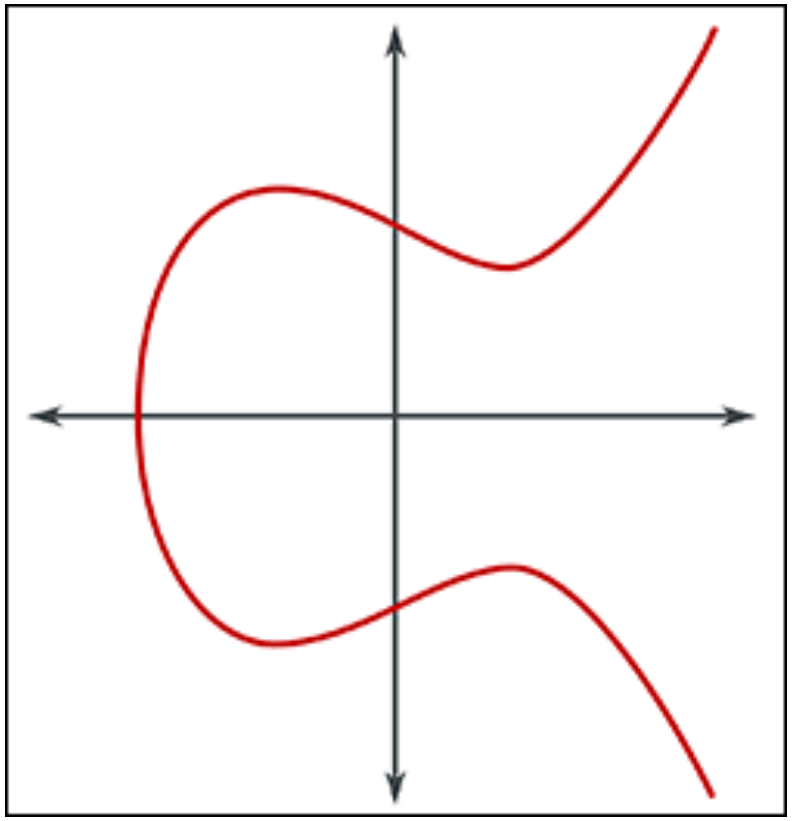
对每一个密钥$k = (n, p, q, d, e)$，定义加密变换为$Ek(x) = xe\ mod\ n$，解密变换为$Dk(x) = yd\ mod\ n$，这里$x, y \in Zn$；

p, q销毁，以$\{e, n\}$为公开密钥，$\{d, n\}$为私有密钥。

07 椭圆曲线算法

- **ECC加密算法又称椭圆曲线加密算法，是由密码学家Koblitz和Miller于1985年提出。椭圆曲线加密算法利用椭圆曲线上的有理点构成Abel加法群上椭圆离散对数的计算困难性而建立的。以加密模型为例，该算法的主要过程如下：**

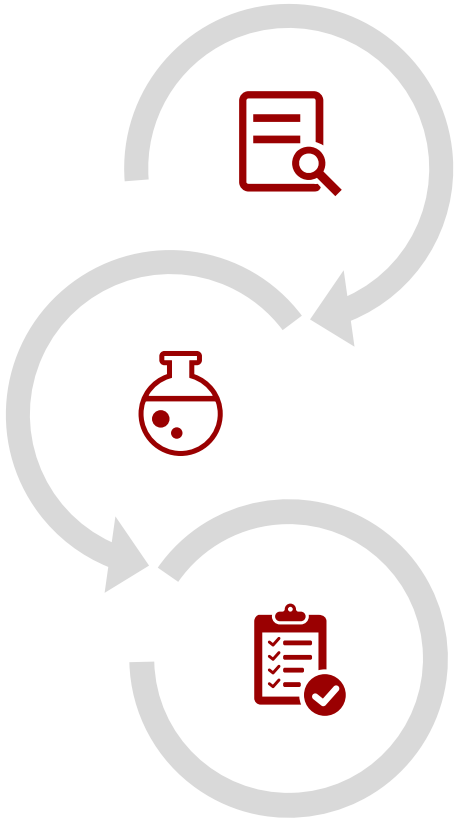
  - Bob首先选定一条椭圆曲线$\mathrm{E_p}(a,b)$，并取椭圆曲线上一点作为基点G。然后选择一个整数k作为私钥，并生成公钥$K=kG$。在$(E_{\mathrm{p}}(a,b),G,K)$为公钥、$k$为私钥。
  - Alice将明文M编码到椭圆$\mathrm{E_p}(a,b)$曲线上一点P，然后选择一个整数R。Alice使用Bob的公钥加密明文M，计算$\mathrm{C_1=P+RK}$、$\mathrm{C_2=RG}$。在上述过程中，（$\mathrm{C_1}$，$\mathrm{C_2}$）为密文，将密文发送到不安全信道中。
  - Bob使用自己的私钥整数k解密密文，$\mathrm{M=C_1-kC_2}$，即可复原出明文M。



08 数字签名算法

- **数字签名是基于公钥密码体制（非对称密钥密码体制）的。数字签名的基本特征如下：**

报文的完整性：接收者不能伪造对报文的签名或更改报文内容。

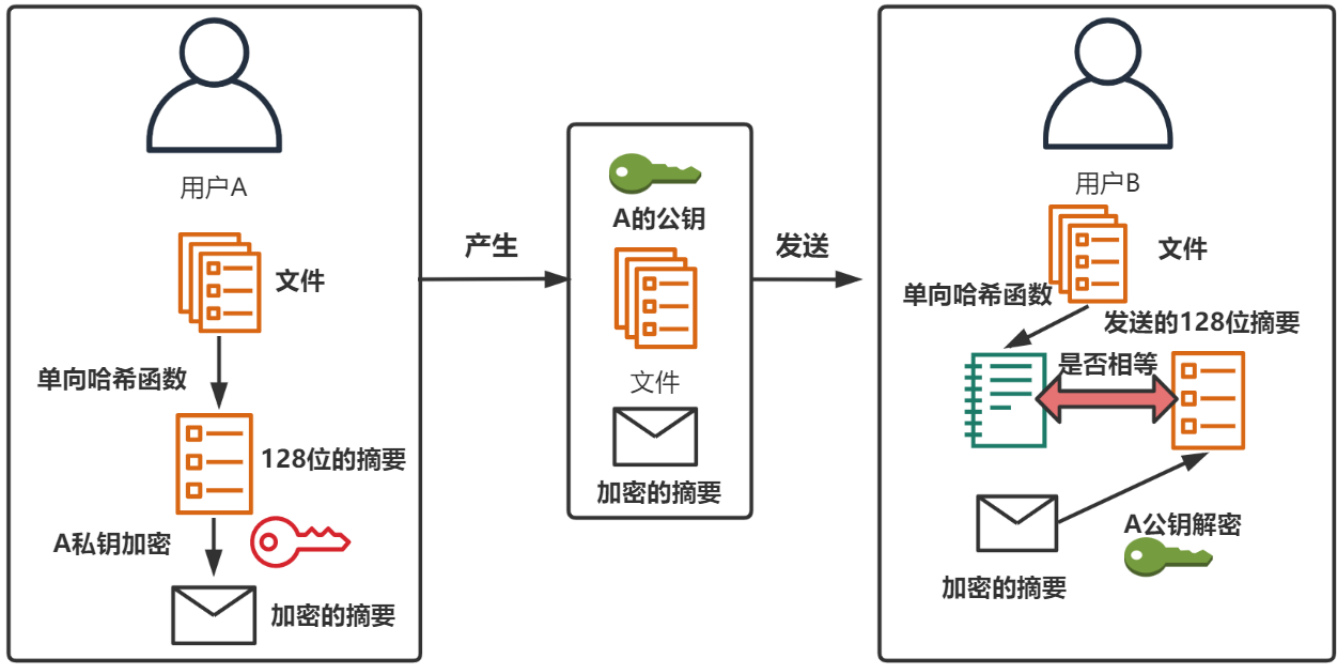


报文鉴别：接收者能够核实发送者对报文的签名；

不可否认：发送者事后不能抵赖对报文的签名。

08 数字签名算法

- 数字签名



- 首先，文件经过单向散列函数的处理得到一份占128位的摘要；
- 随后，用户A使用自己地私钥对这份128位地摘要进行加密，得到一份加密地摘要。
- 户A把文件、加密的摘要和公钥打包一起发给用户B。
- 用户B将收到的文件经过单向散列函数处理得出一份128位摘要
- 如果两份摘要相等，说明文件经过用户A签名之后，在传输的过程中没有被更改；若不相等，说明文件在传输过程中被更改了，或者说已经不是原来的文件了，此时用户A的签名失效。

第 二 节

结构基础

01 Hash函数

02 SHA256函数

03 Hash指针

04 链式结构

05 梅克尔树

06 区块结构

北京大学 PEKING UNIVERSITY

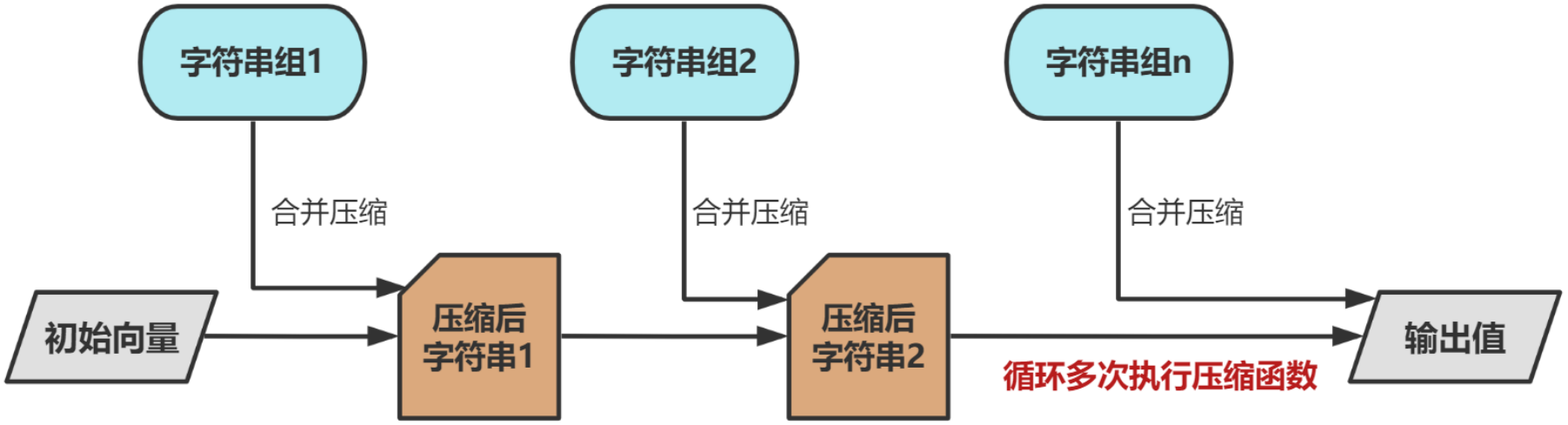
定义

Hash函数也称作哈希函数，它是一类可将任意长度字符串映射到定长字符串的函数，该过程可以看作是对一串信息的编码过程。

体系

Hash函数经过了多年的发展，已经形成较为成熟的体系。上世纪主流的Hash函数是MD家族的Hash函数，包括MD2、MD4、MD5等 。NSA在MD家族的Hash函数的基础上，发布了SHA家族的Hash函数，包括SHA1、SHA2系列、SHA3系列等，其中包括比特币中使用的Hash函数——SHA256。

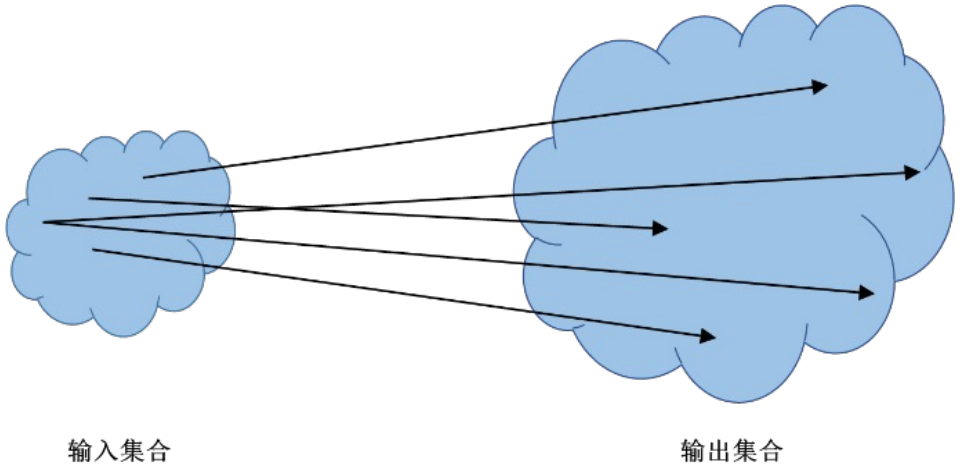
- 哈希函数的实现



Hash函数的实现基本上是**由若干个压缩函数组成** Hash函数实质上是**多次执**行特定的压缩函数，首先将任意长度的字符串拆分成符合压缩函数输入要求的长度分组，其中最后一个分组通常需要填充特定的字符。

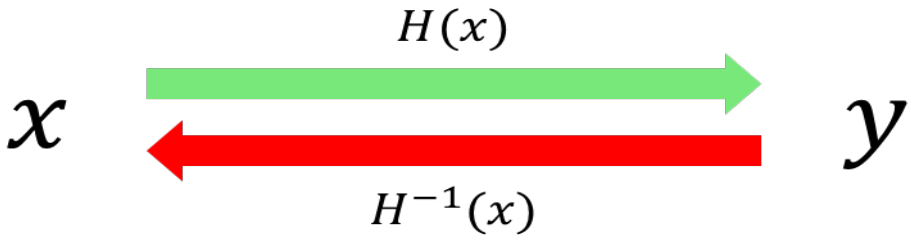
01 哈希函数

- 一个Hash函数通常具有以下三条性质，包括免碰撞性、隐匿性和谜题难解性。



谜题难解性：

给定因变量$y=H(x)$时，不能通过有效的算法计算自变量x。对这种能够在一定的时间复杂度下正向计算、但是很难逆向计算的函数称为单向函数 。



免碰撞性：

系统可以将一条信息的哈希值可以当作它的身份标识，该身份标识就像身份证号一样，在整个编码系统中是唯一的 。

隐匿性：

Hash函数的谜题难解性是在隐匿性的基础上延伸出来的。当给定因变量$y=H(x)$时，若其对应的自变量x存在部分随机序列，则不能有效地求解x。

02 SHA256函数

- **首先是一个哈希函数。SHA256算法是SHA-2族算法的其中一个算法，被应用于比特币挖矿和merkle树中。SHA-2（Secure Hash Algorithm 2），是由美国国家安全局研发的一种密码散列函数标准，属于SHA算法之一，是SHA-1的后继者。**

算法标准

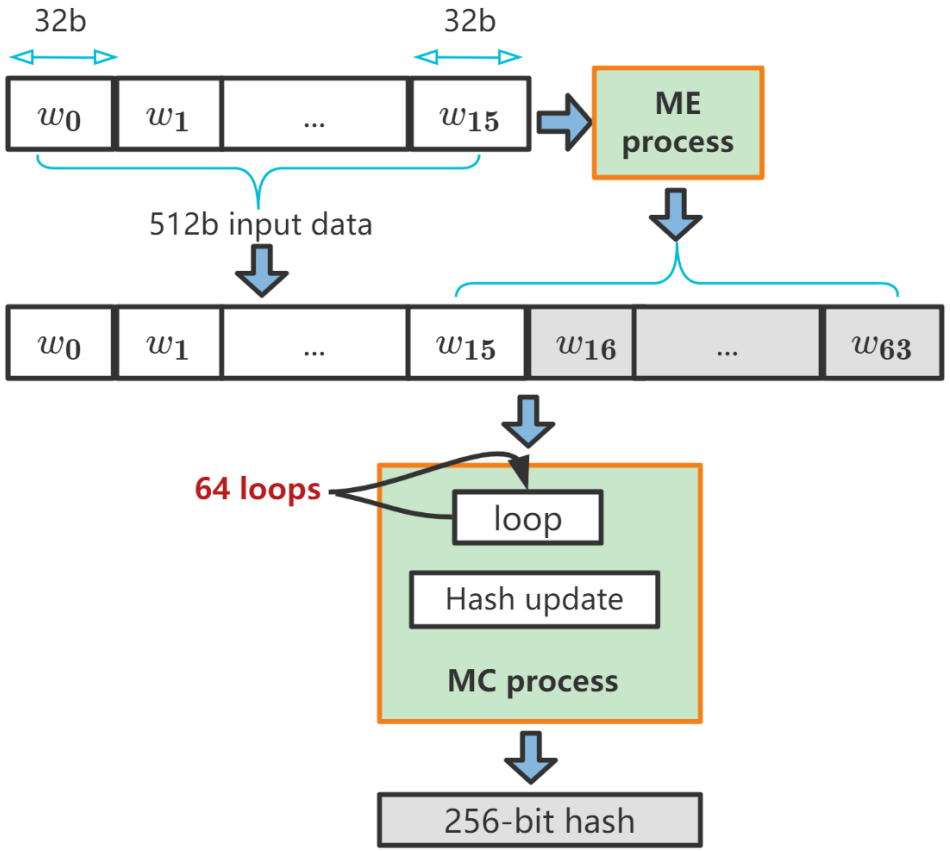
SHA-224、SHA-256、SHA-384、SHA-512、SHA-512/224、SHA-512/256。这些变体除了生成摘要的长度 、循环运行的次数等一些微小差异外，算法的基本结构是一致的。

算法输入输出

- **输入**：任意长度的消息；
- **输出**：SHA256都会产生一个256bit长的哈希值，称作消息摘要。相当于是个长度为32个字节的数组，通常用一个长度为64的十六进制字符串来表示

02 SHA256函数

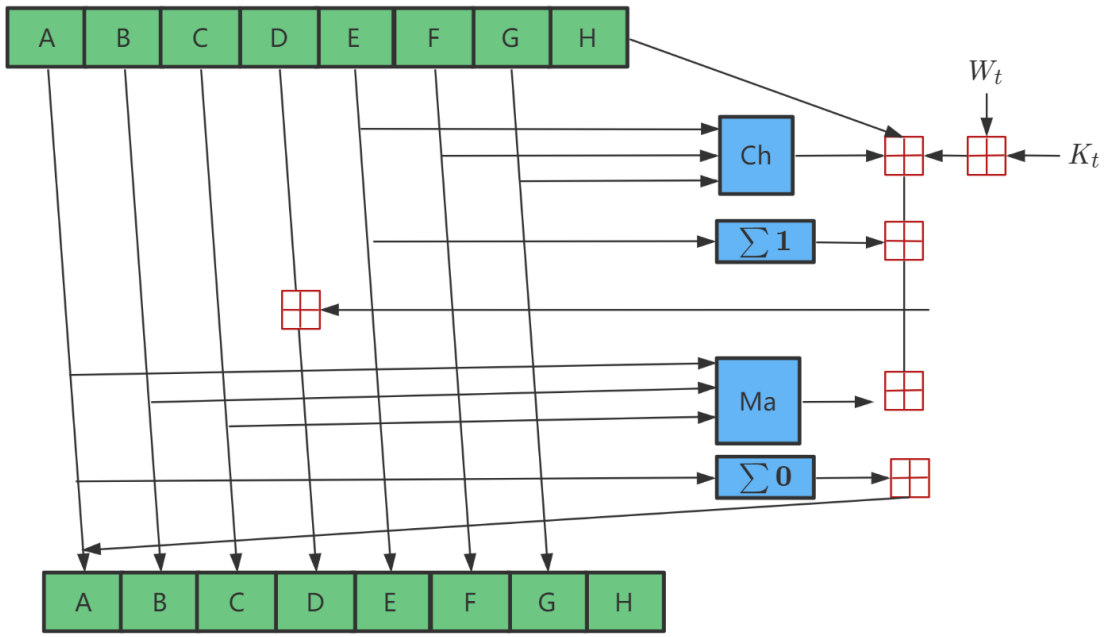
- SHA256算法步骤



- **常量初始化：**SHA256算法中用到了8个哈希初值以及64个哈希常量；
- **信息预处理：**预处理就是在想要Hash的消息后面补充需要的信息。在报文末尾进行填充，使报文长度在对512取模以后的余数是448；
- **逻辑运算：**SHA256散列函数中涉及的操作全部是逻辑的位运算。包括逻辑运算“与”、“补”、“异或”、循环右移n个bit，右移n个bit。

02 SHA256函数

- SHA256算法核心步骤：计算消息摘要



消息分解

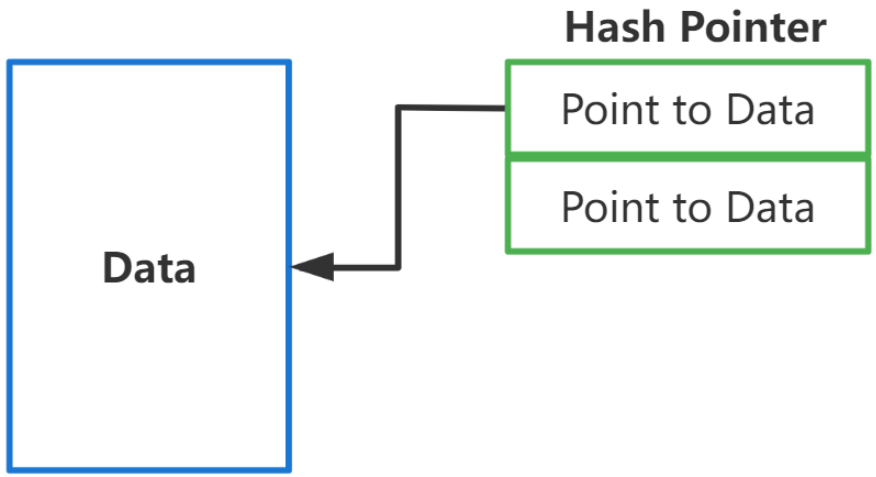
- 首先：将**消息分解成512-bit大小的块**，假设消息M可以被分解为n个块，于是整个算法需要做的就是完成n次迭代，n次迭代的结果就是最终的哈希值，即256bit的数字摘要。

每次迭代

- **Step1：构造64个字**，对于每一块，将块分解为16个32-bit的big-endian的字，记为$w[0],\ldots,w[15]$也就是说，前16个字直接由消息的第i个块分解得到，
- **Step2：进行64次循环**，将ABCDEFGH这8个字（word）在按照一定的规则进行更新，即进行64次加密循环即可完成一次迭代。深蓝色方块是事先定义好的非线性逻辑函数。

03 Hash指针

- **区块链为了提供更好的数据完整性保证，使用哈希指针（Hash Pointer）的技术。**



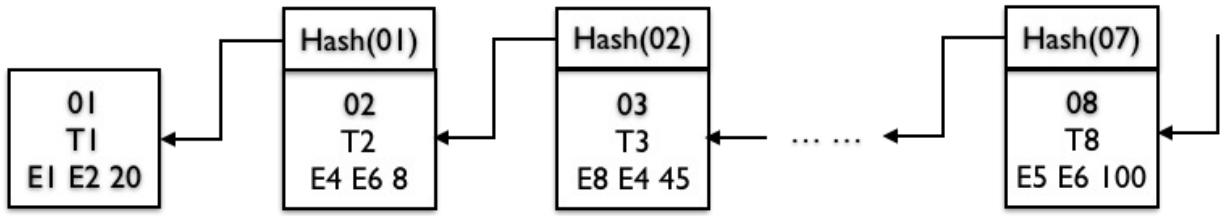
- Hash指针将**指针和哈希函数**相结合，哈希指针不仅包括数据存储的指针（即地址），还包括所指数据的哈希值。知道一个数据的指针可以取回该指针所指的数据，知道一个数据的Hash指针**不仅可以取回该指针所指的数据，还可以判断该数据是否已经修改。**

04 链式结构

- **在上述交易数据的平板结构上，可以通过增加一列Hash指针来构建余个链式的交易数据存储，假设交易是按照序号不断发生的。**

改造过程：将T1交易数据的Hash指针存在T2交易中，将T2交易数据和携带的T1的Hash指针生成一个新的摘要指针存在T2交易中，按照这样的处理方式，新发生一个交易，就将上一个交易的数据生成摘要指针存在新交易中。

链式结构



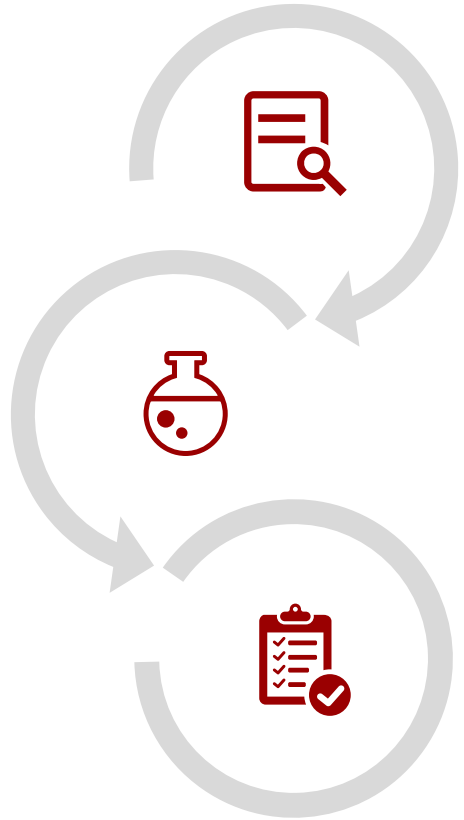
平板结构

序号	交易	内容	交易数额	Hash指针	Hash指针计算方法
01	T1	E1 E2	20		
02	T2	E4 E6	8	Hash(01)	= Hash(01,T1, E1 E2, 20)
03	T3	E8 E4	45	Hash(02)	= Hash(02,T2, E4 E6, 8, Hash(01))
04	T4	E6 E7	32	Hash(03)	= Hash(03,T3, E8 E4, 45, Hash(02))
05	T5	E5 E4	14	Hash(04)	= Hash(04,T4, E6 E7, 32, Hash(03))
06	T6	E7 E8	21	Hash(05)	= Hash(05,T5, E5 E4, 14, Hash(04))
07	T7	E2 E5	30	Hash(06)	= Hash(06,T6, E7 E8, 21, Hash(05))
08	T8	E5 E6	100	Hash(07)	= Hash(07,T7, E2 E5, 30, Hash(06))
…	…	…	…	…	… …

04 链式结构

- **链式结构的修改难度是O（n），n表示链的长度和要修改的交易位置之差：如果n很大需要重构一遍所有涉及的交易，需要耗费非常大的存储和计算资源。**

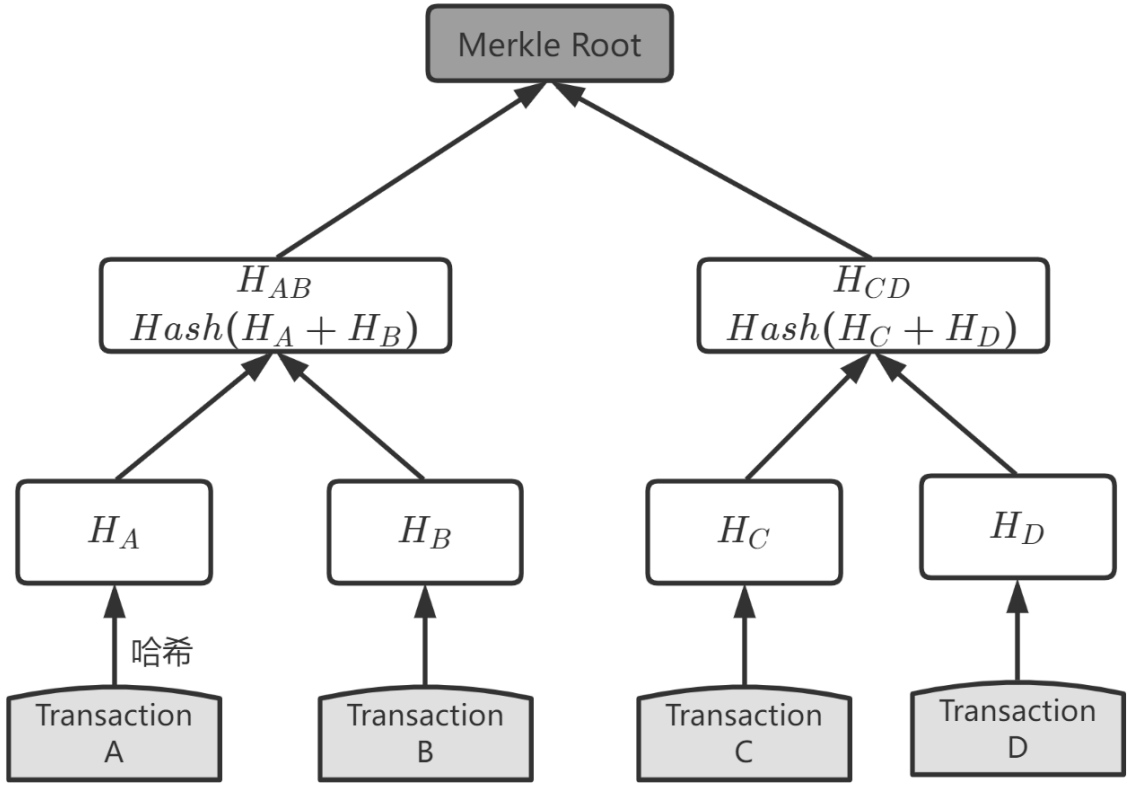
链式结构可以提供**较好的数据真实性和完整性保证**，平板结构下数据的真实性和完整性需要其上的应用来保证，数据库本身并不提供，区块链借助于链式结构可以在**系统层面提供数据真实性和完整性保证。**



链式结构使得交易数据的**存储难于修改**，即区块链的只增和不可篡改，不是真的不能修改，交易数据是可以修改的，但是修改后不被发现比较困难，需要**修改者付出较大的代价。**

链式结构的只增和不可篡改即是区块链的优点，也是区块链的缺点，数据一旦上链要想修改也非常困难，如果需要对数据**频繁的写**的应用可能使用区块链就会**面临一些挑战。**

05 梅克尔树



- 区块链实际系统中为了**节省资源**，大多数采用多个**交易打包成一个区块**，然后将区块作为一个计算单元来梅克尔。
- 梅克尔树是**使用Hash指针替代了二叉树的普通指针**，最初的设计目的是为了**提供数据的存证**，叶子结点是具体的交易、梅克尔树不仅可以**检查数据的存在性**，如果是排序的梅克尔树，还可以检查数据是否不存在于该树中。

05 梅克尔树

为了证明区块中存在某个特定的交易，一个节点只需要计算log2(N)个32字节的哈希值，形成一条从特定交易到树根的认证路径或者Merkle路径即可。

在单个区块中有成百上千的交易是非常普遍的，这些交易都会采用同样的方法归纳起来，产生一个仅仅32字节的数据作为梅克尔树根。

随着交易数量的急剧增加，这样的计算量就显得异常重要，因为相对于交易数量的增长，以基底为2的交易数量的对数的增长会缓慢许多。

06 区块结构

- 区块结构如下表所示

大小	字段	描述
4字节	区块大小	用字节表示的该字段之后的区块大小
80字节	区块头	组成区块头的几个字段
1-9（可变整数）	交易计数器	交易的数量
可变的	交易	记录在区块里的交易信息

区块结构：区块大小、区块头、交易计数器、交易

06 区块结构

北京大学 PEKING UNIVERSITY

- 区块头结构如下表所示

大小	字段	描述
4字节	版本	版本号，用于跟踪软件/协议的更新
32字节	父区块哈希值	引用区块链中父区块的哈希值
32字节	Merkle根	该区块中交易的Merkle树根的哈希值
4字节	时间戳	该区块产生的近似时间
4字节	难度目标	该区块工作量证明算法的难度目标
4字节	Nonce	用于工作量证明算法的计数器

06 区块结构

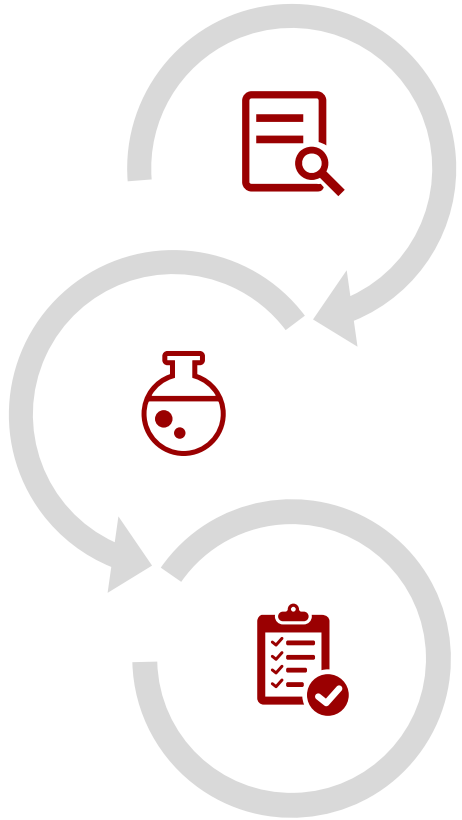
- 区块头由三组区块元数据组成。

父区块哈希

首先是一组引用父区块哈希值的数据，这组元数据用于将该区块与区块链中前一区块相连接。

元数据

第二组元数据，即难度、时间戳和nonce，与挖矿竞争相关。

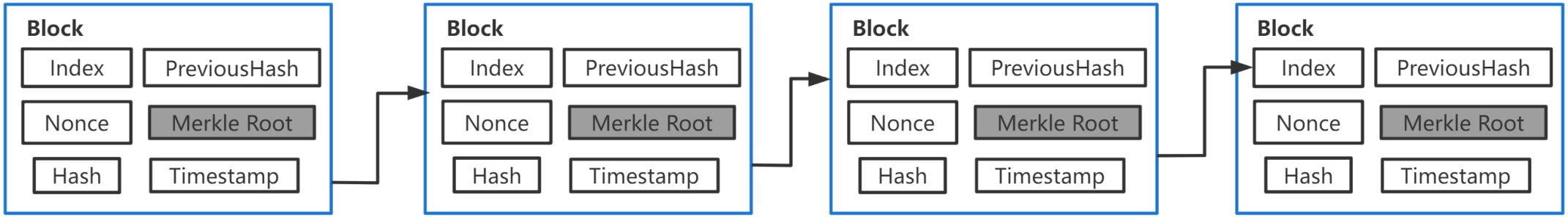


Merkle树根

第三组元数据是Merkle树根。主要用来简化交易的结构。

06 区块结构

- 比特币的完整节点保存了区块链从创世区块起的一个本地副本。随着新的区块的产生，该区块链的本地副本会不断地更新用于**扩展这个链条**。当一个节点从网络接收传入的区块时，它会验证这些区块，然后链接到现有的区块链上。为建立一个连接，一个节点将检查传入的区块头并寻找该区块的“**父区块哈希值**”。



第 三 节
网络结构

01 对等网络

02 代表应用

03 网络结构

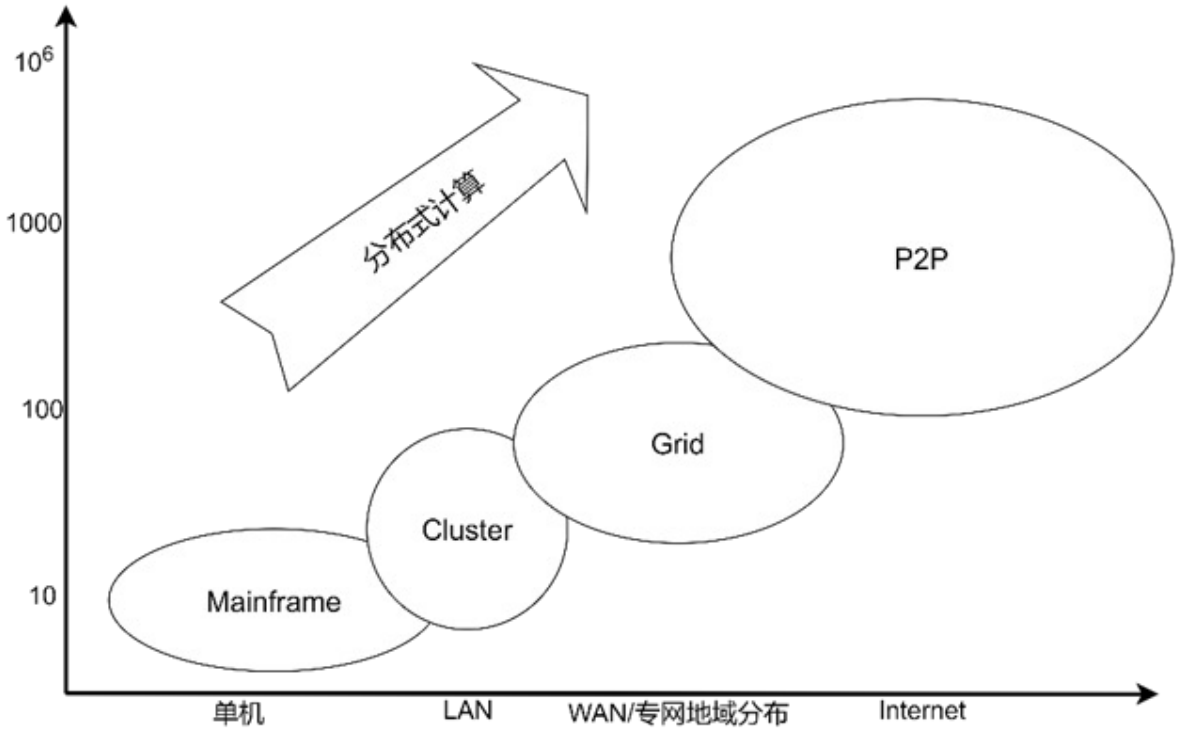
04 路由算法

01 对等网络

网络定义

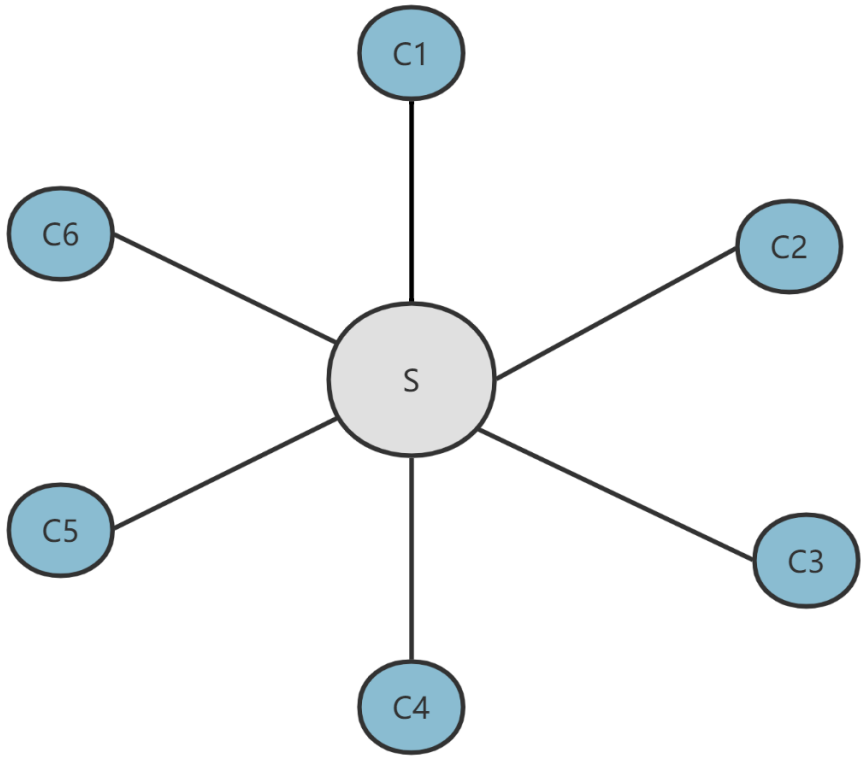
对等网络(pee-to-peer network,简称P2P网络)是**分布式系统与计算机网络**相结合的产物,是采用对等模式工作的计算机网络。

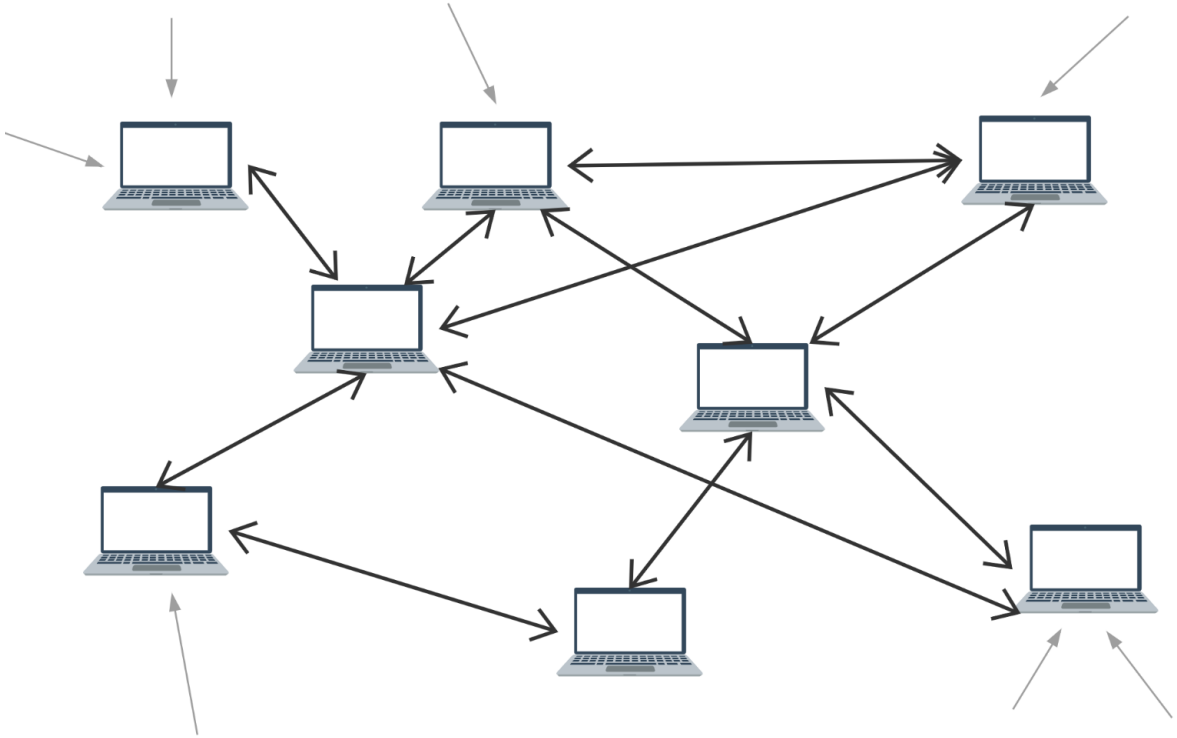
右图描绘了随着分布式系统规模的扩展,分布式计算的模式相应发生的改变。在对等网络中,每个网络节点**在行为上是自由的,在功能上是平等的,在连接上是互联的,**所有节点分布式地自组织成一个整体网络,因此,它能够极大程度地提高网络效率,充分利用网络带宽,开发每个网络节点的潜力。



01 对等网络

- **本质思想：**对等模式的本质思想在于打破传统的客户/服务器模式,让一切网络成员 享有自由、平等、互联的功能,不再有客户、服务器之分,任何两个网络节点之间都能**共享文件、传递消息**。





01 对等网络

- **P2P是由若干互联协作的计算机构成的系统。**

P2P系统特征

- 系统依存于边缘化（非中央是服务器）设备的主动协作，每个成员直接从其他成员而不是从服务器的参与中受益；
- 系统中成员同时扮演服务器与客户端的角色；
- 系统应用的用户能够意识到彼此的存在而构成一个虚拟或实际的群体。

P2P系统含义

- **P2P实现技术**：实现P2P应用系统时所用到的技术，包括相关协议（如Gnutell、FastTrack等）。
- **P2P通信模式**：与传统的C/S模式不同，每个通信放都具有相同的逻辑能力，并且每个通信放都有能力发起一个通信过程。；
- **P2P网络**：由P2P节点、附属管理设备（如索引服务器等）及其相关应用等组成的可实现P2P功能的网络。

02 代表应用

- **P2P的代表应用：napster**

音乐文件共享应用

1999年，著名的音乐和文件共享应用程序Napster出现了。Napster是点对点网络运动演变为BitTorrent的开始，参与用户建立了一个虚拟网络，完全独立于物理网络，无需遵守任何管理机构或限制。

由于点对点机制可用于访问任何类型的分布式资源，因此它们在去中心化应用程序中起着核心作用。

02 代表应用

- **P2P的代表应用：BitTorrent、Skype**

BitTorrent简称BT，是一个借助分布式服务器提供**共享文件索引**的混合式P2P网络，**文件分片下载**，并限定了用户在下载的同时必须上传。

Skype是一个**优秀的P2P网络语音传输工具**，既提供高清晰的**语音对话**，还可以用来拨打国内国际电话。同时，Skype也提供**网络聊天、文件传输**等功能。

02 代表应用

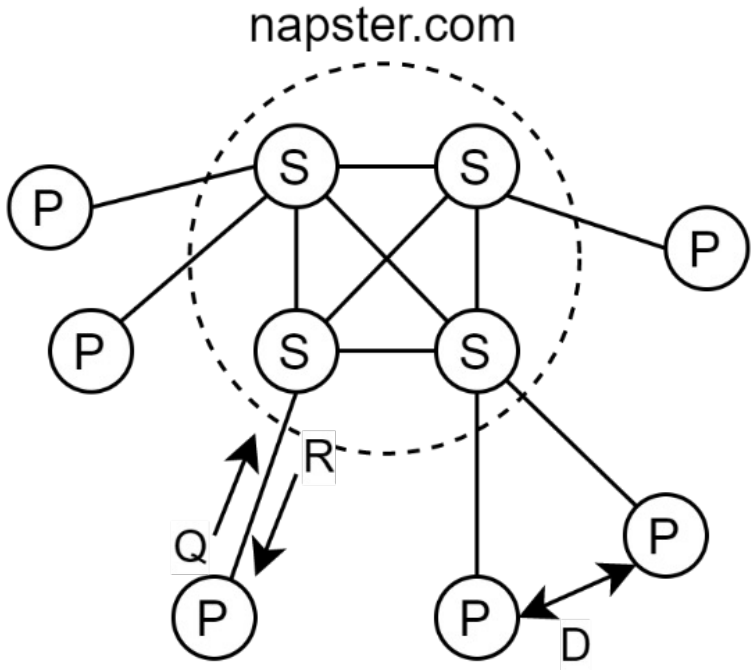
- **P2P的代表应用：KaZaA、eDonkey**

KaZaA最大的特点是将**网络节点组织成两类——超节点和普通节点**，以此来开发节点的异构性。

eDonkey（电驴）将网络节点以类似KaZaA的方式组织成两层——**服务器层和客户层**，但eDonkey将**文件分块**从而提供多源下载机制。

除了以上提到的工具外，还有一些国内知名P2P软件，如网络电视软件PPLive、实时通信软件QQ、多源下载软件迅雷等。

- 对等网络结构主要分为混合型对等网络、无结构对等网络、有结构对等网络



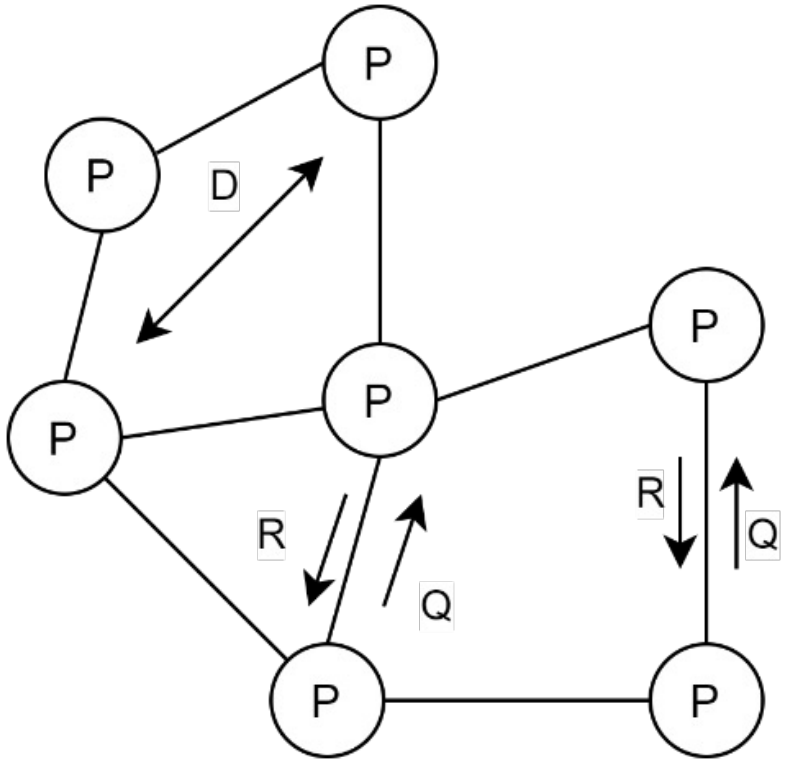
混合型对等网络含义

又称第一代P2P网络，都采用混合式体系结构，即**星形拓扑结构**，服务器仍然是整个网络的核心。图中所表示的就是Napster的工作原理图。Napster网站是一个服务器集群，所有服务器消息互联并**对外提供统一的访问接口**，从用户角度看他们**访问的都是同一台服务器。**

混合型对等网络特点

- 采用星形拓扑结构，服务器仍然是整个网络的核心
- 需要先查询文件索引，之后用户建立与文件提供者的连接
- 采用服务器集群的机制，提高查询的容错率
- 出于简单、高效的考虑，混合式P2P网络基本上不提供匿名性
- 后续的混合式P2P网络都或多或少的提供了增强机制

- 无结构对等网络



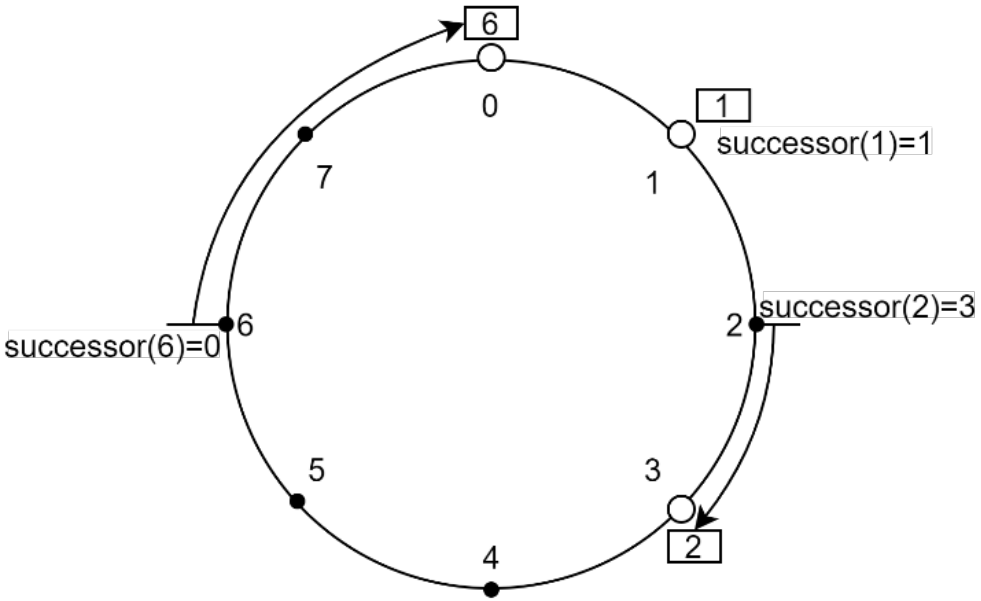
无结构对等网络含义

在Napster出现后不久，**Gnutella**出现，它也是第二代P2P网络最好的代表。在Gnutella网络中只存在一种节点——**对等实体**（Peer），不再存在其服务器。**每个Peer既是客户端又是服务器**，既能够向其他Peer**发送查询请求并获得查询结果**，还负责**监控网络局部的通信状态。**

无结构对等网络特点

- 无结构指的是P2P网络没有固定、严格的拓扑结构，而是一个随机生成、松散组织的普通图
- 消息路由带有很大的随机性，对于随机节点失效具有高容错性。
- 简易的自适应行为、可扩展性较低、具有高安全性和匿名性

- 有结构对等网络



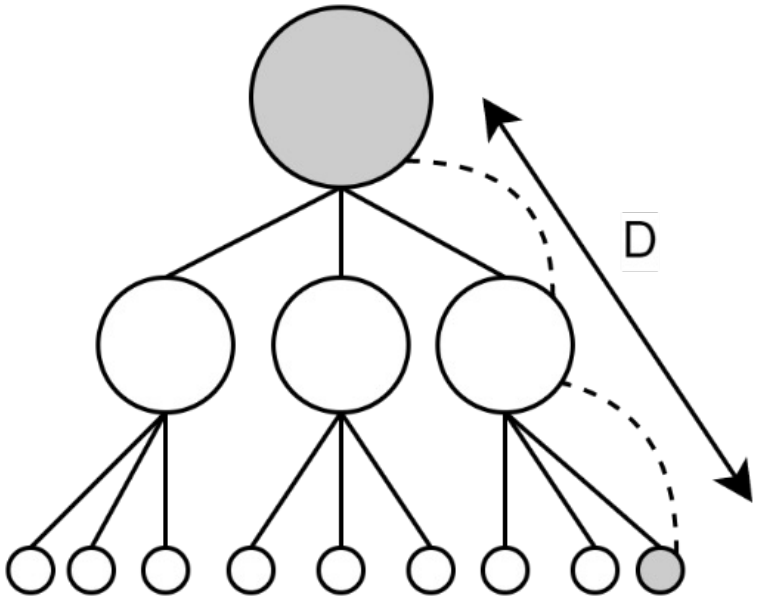
有结构对等网络含义

有结构P2P网络领域中最为经典的模型就是**Chord**，Chord**将数据对象索引分配到网络节点**，所有节点按照nodeID从小到大顺时针排列在一个环上，数据对象k（即objectID）被分配到环上顺时针方向紧随k的第一个节点，称该节点为k的后继，记作successor(k)。每个Chord节点也有其后继。

有结构对等网络特点

- 有一个**严格的网络拓扑结构**，主要拓扑结构包括带弦环、多维空间、超立方体、跳表等结构。
- 使用**分布式散列表**，将节点数据对象映射到覆盖网络中。
- **分布式、局部贪心**的路由算法。
- 复杂的**自组织自适应算法**。
- 局部性的目的是**提高覆盖网络和物理网络的一致性**。

- 洪泛法



1 基本原理

一种简单的路由算法，将收到的数据包，**往所有的可能连结路径上递送**，**直到数据包到达为止**，它是**无结构P2P网络**最朴素、最直接的路由方法。

2 洪泛过程

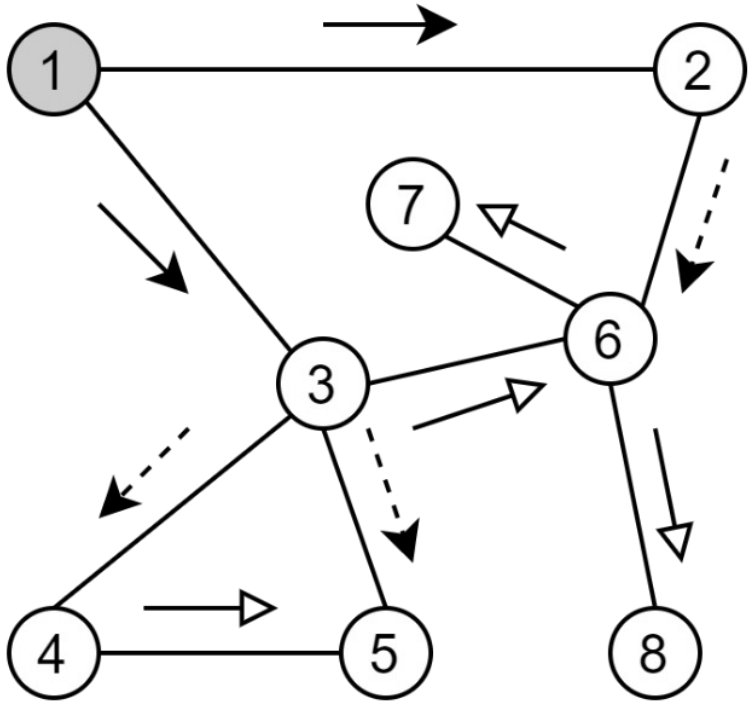
如左图3层的洪泛过程。节点希望请求数据时，它会向其所有邻居（图中为3个）**发送请求数据包，**所有连线上都存在一个请求的副本（图中共有12个副本）。直到找到对应的目标数据，**目标数据会沿着路径返回给请求节点。**

3 优缺点

优点是能够保证，当一个数据包可以被送达时，该数据包一定会被送达

缺点是传输的数据量极大，不适合高负载的环境，出现闭环需要解决无限循环的数据包。

- Gossip



1 基本原理

Gossip protocol也叫Epidemic Protocol（流行病协议）基于社交网络中著名的**六度分隔理论**，由**种子节点**发起，**随机的选择**周围几个节点散播消息，收到消息的节点也会**重复**该过程，**直至所有的节点都收到了消息。**

2 传播过程

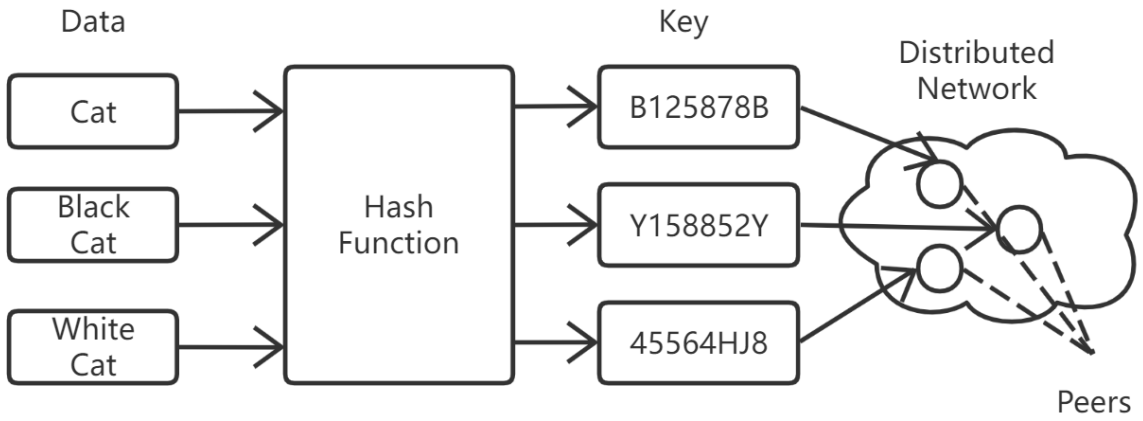
第一轮传播，节点1向节点2、3发送数据包；**第二轮传播**，节点2、3分别向自身的邻居发送数据包；**第三轮传播**，节点3、4、6自身的邻居发送数据包，其中节点6在第二轮已接收该数据包，在本轮中接收到来自节点3的数据包后直接丢弃。**最终所有节点都达到最新状态。**

3 优缺点

优点是网络是可拓展的，具有容错特性，强一致性收敛、完全的去中心化
缺点是消息的延迟和网络中存在消息冗余。

北京大学 PEKING UNIVERSITY

- 分布式哈希表（Distributed Hash Table，DHT）



➢ 分布式哈希表（Distributed Hash Table，DHT）是**分布式计算系统**中的一类，设计之初是为了**解决在P2P网络中节点过多**的情况下，一次查询非常慢的缺陷，它用来将一个**关键值**（key）的集合分散到所有在分布式系统中的节点，并且可以有效地将消息转送到唯一一个拥有查询者提供的**关键值的节点**（Peers）。

分布式哈希表特性

- **离散性**：构成系统的节点并没有任何中央式的协调机制。
- **伸缩性**：即使有成千上万个节点，系统仍然应该十分有效率。
- **容错性**：即使节点不断地加入、离开或是停止工作，系统仍然必须达到一定的可靠度。

第 四 节

激励基础

01 博弈论

02 囚徒困境

03 纳什均衡

04 帕累托最优

05 区块链应用

北京大学 PEKING UNIVERSITY

- **博弈论起源**

约翰·冯·诺伊曼

博弈论是针对互动情况下人们策略行为的数学建模研究。

在**社会科学**的**所有领域**以及**逻辑**、**系统科学**和**计算机科学**中都有所应用。

现代博弈论始于**两人零和博弈**中的**混合策略均衡**思想及其由约翰·冯·诺伊曼进行的证明。

众多学者自上世纪50年代起广泛地发展了博弈论，如今博弈论已被广泛认为是许多领域的重要研究工具。

而现实中，人们在日常生活的各个方面也会需要进行博弈。

北京大学 PEKING UNIVERSITY

- 博弈要素

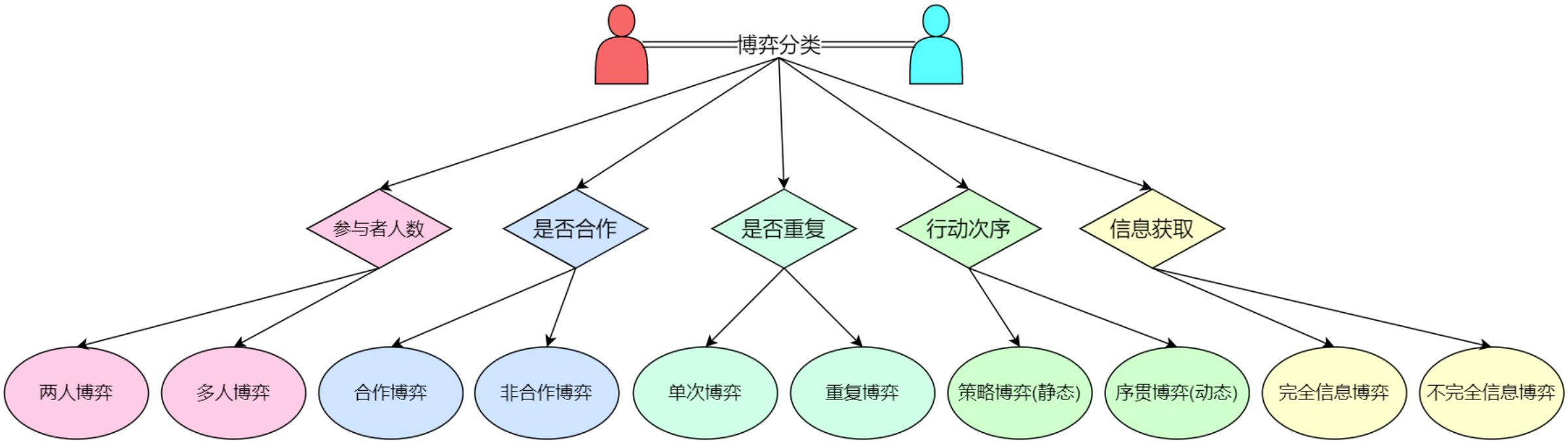
石头剪刀布-收益矩阵

A/B	石头	剪刀	布
石头	(0, 0)	(1, -1)	(-1, 1)
剪刀	(-1, 1)	(0, 0)	(1, -1)
布	(1, -1)	(-1, 1)	(0, 0)

- **博弈者**：博弈者是博弈中的决策者。在区块链中，博弈者可以是矿工、矿池或区块链用户等等。
- **策略**：博弈者的策略是一系列的行动、选择或决定，博弈者可以执行这些策略来达到他们期望的结果。
- **收益**：收益，即回报、利息或收入，用来反映博弈者的预期结果。一般来说，博弈者的收益不仅取决于博弈者自己采取的策略，还取决于其他博弈者采取的策略。

北京大学 PEKING UNIVERSITY

- 博弈分类



02 囚徒困境

- 博弈描述

坦白 or 抵赖？

假设两个嫌犯被抓捕，不能相互沟通，警方缺乏证据进行定罪，分别审问两人。如果两个人**都选择抵赖**，则由于证据不足，每个人都坐牢一年；若**一人坦白**，而**另一人抵赖**，则坦白者可以获释，抵赖者入狱十年；若**都选择坦白**，则因证据确凿，两人都判刑八年。

囚徒困境-收益矩阵

囚徒 A/囚徒 B	坦白	抵赖
坦白	(-8, -8)	(0, -10)
抵赖	(-10, 0)	(-1, -1)

4.3 纳什均衡

- **占优策略**

囚徒困境-收益矩阵

囚徒A/囚徒B	坦白	抵赖
坦白	(-8, -8)	(0, -10)
抵赖	(-10, 0)	(-1, -1)

弱占优策略含义

对一个参与人（A）来说，若存在一个策略，无论另一个参与人（B）选择何种行为策略，该策略可以获得的收益都**不低于**其他策略，则该策略就称为是A的**弱占优**策略。

严格占优策略含义

对一个参与人（A）来说，若存在一个策略，无论另一个参与人（B）选择何种行为策略，该策略都是**最佳选择**（收益**高于**其他选择），则该策略就称为是A的严格占优策略。

囚徒困境中的严格占优策略

在囚徒困境中，无论对方如何选择，自身选择坦白的收益都大于自身选择抵赖的收益，因此每个人的严格占优策略都是**坦白。**

(坦白，坦白)为囚徒困境的占有策略均衡。

占有策略均衡属于**纳什均衡。**

4.3 纳什均衡

- 纯策略纳什均衡

纳什均衡概念

指的是博弈中的一个策略组合(每个参与人选择一个相应的策略)，在其他参与人都**坚守**这个策略组合中的策略不变的情况下，**没有**参与人可以通过**改变自己的策略**而得到一个**更高的支付**。

- 纳什均衡可以看作一种信念上的均衡
- 在均衡状态下，参与人**没有动机**去更换策略
- 互为**最优应对策略**，谁也不可能通过单方面改变策略而得到额外收益，尽管如果两人都改变策略可能会更好

囚徒困境-收益矩阵

囚徒 A/囚徒 B	坦白	抵赖
坦白	纳什均衡 (-8, -8)	(0, -10)
抵赖	(-10, 0)	(-1, -1)

纯策略纳什均衡

- 纯策略：在给定信息下，只能选择一种**特定**策略。
- 纯策略纳什均衡指博弈者以**概率0或1**决定使用某个策略时达到的纳什均衡

4.3 纳什均衡

• 混合策略纳什均衡

石头剪刀布-收益矩阵

A/B	石头	剪刀	布
石头	(0, 0)	(1, -1)	(-1, 1)
剪刀	(-1, 1)	(0, 0)	(1, -1)
布	(1, -1)	(-1, 1)	(0, 0)

- 在石头剪刀布的博弈游戏中，**没有纯策略**可以达成纳什均衡。
- 对于玩家A，在玩家B的策略分别为石头、剪刀、布的情况下，玩家A的最佳应对分别为布、石头、剪刀，**并没有单一的最优反应策略。**

混合策略纳什均衡含义

- 引入随机性；
- **混合策略**是给**每个纯策略**分配一个**概率**，一个玩家的策略集就是一个“样本空间”；
- 混合纳什均衡指博弈者以**概率**p决定使用某个策略时达到的纳什均衡。

$$\frac{1}{9}(0)+\frac{1}{9}(-1)+\frac{1}{9}(1)+\frac{1}{9}(1)+\frac{1}{9}(0)+\frac{1}{9}(-1)+\frac{1}{9}(-1)+\frac{1}{9}(1)+\frac{1}{9}(0)=0$$

- 在石头剪刀布的博弈游戏中，当参与者选择石头、剪刀、布的概率相等，各为1/3时，A每轮游戏的期望收益最高。
- 因此该概率分布策略为博弈的混合策略纳什均衡。

4.4 帕累托最优

北京大学 PEKING UNIVERSITY

- 帕累托最优与社会最优

帕累托最优

- 帕累托最优是指资源分配的一种理想状态。给定固有的一群人和可分配的资源，如果从一种分配状态到另一种状态的变化中，在**没有使任何人情况变坏的前提下**，使得**至少一个人变得更好**，这就是**帕累托改善**。**帕累托最优**的状态就是不可能再有更多的帕雷托改善的状态；换句话说，**不可能在不使任何其他人受损的情况下再改善某些人的情况。**

囚徒困境-收益矩阵

囚徒 A/囚徒 B	坦白	抵赖
坦白	(-8, -8)	(0, -10)
抵赖	(-10, 0)	(-1, -1)

社会最优

- 一组策略选择是社会最优（或社会福利最大化），若它使参与者的回报之和（**总收益**）**最大；**
- 社会最优也**一定**会是帕累托最优；
- 社会最优和纳什均衡**有可能**一致。

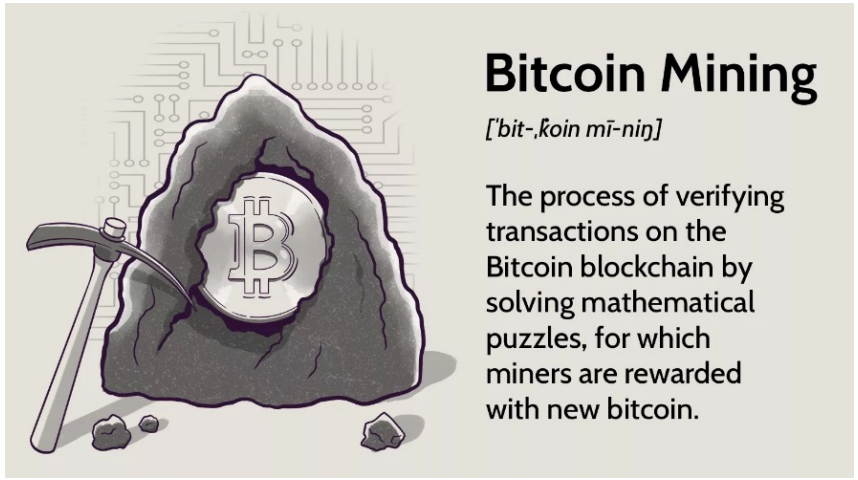
囚徒困境-收益矩阵

囚徒 A/囚徒 B	坦白	抵赖
坦白	(-8, -8)	(0, -10)
抵赖	(-10, 0)	(-1, -1)

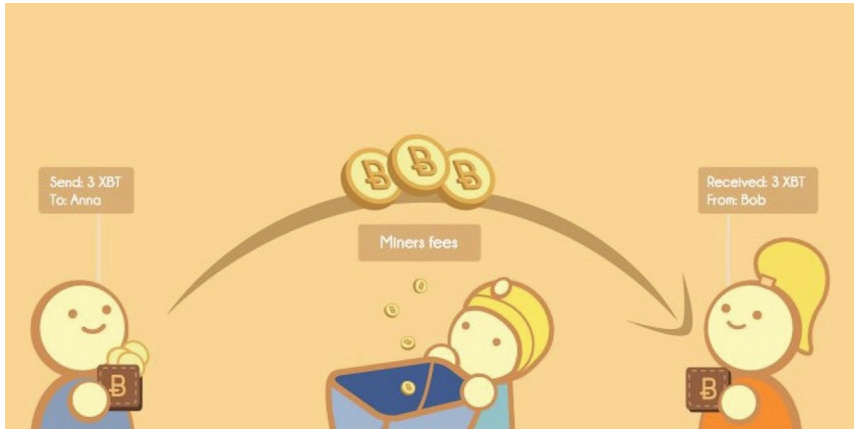
4.5 区块链应用

- 区块链中已有激励

1 挖矿激励



2 交易费用激励

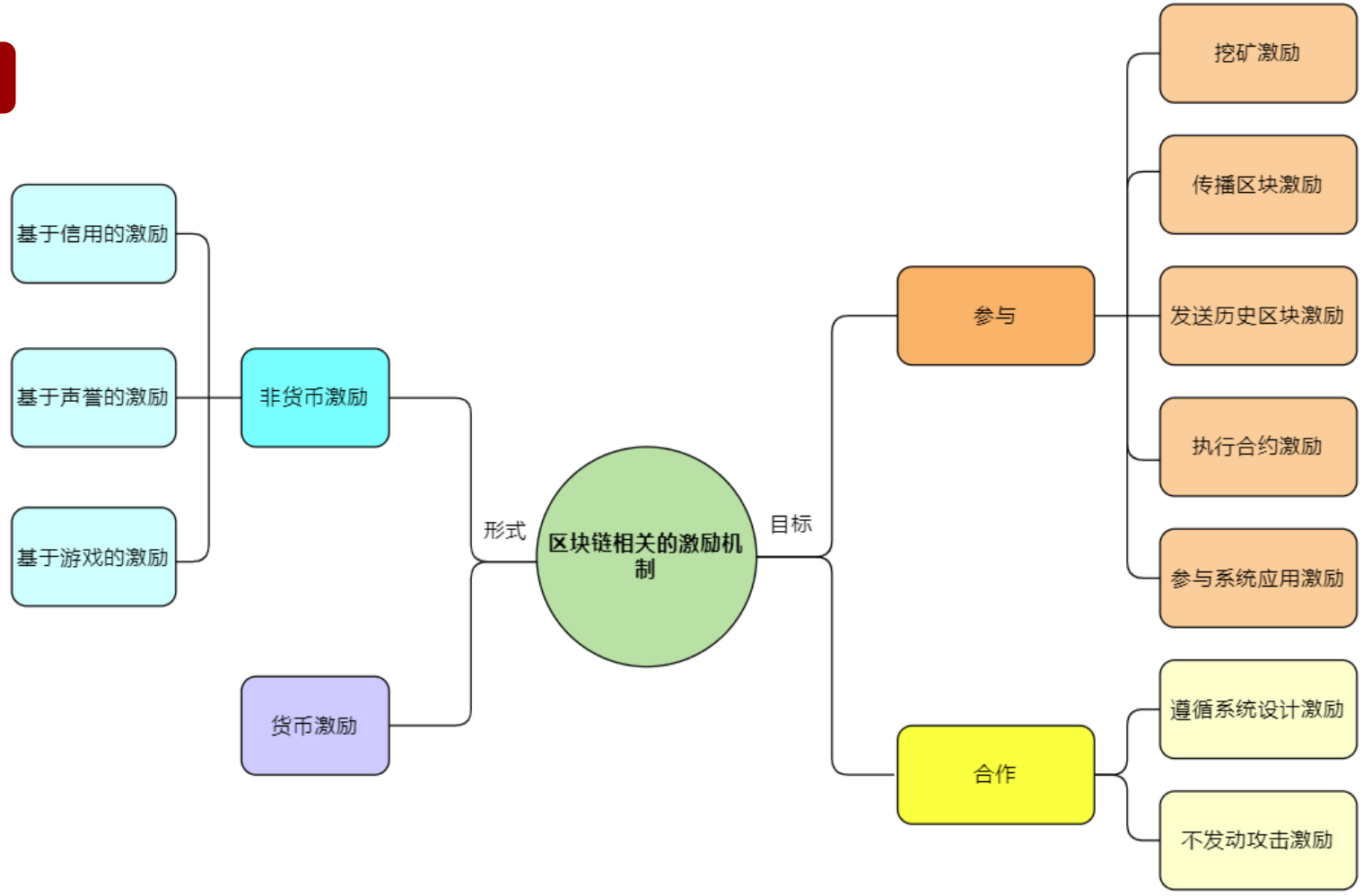


- 比特币挖矿奖励大约**每4年**减少一半，即每挖出**210000个新区块**，系统奖励就会**减半**，从2009年1月开始的每区块**50BTC**奖励不断递减直到2140年**21000000枚**比特币全部发行完。而矿工在打包的每笔交易中单独获得的**交易费**会逐渐成为矿工的大比重收入。
- 交易费的定义标准是每笔交易中**输入总额与输出总额之间的差值**。目前交易费在矿工的总收入中占比**小于0.5%**，但随着挖矿奖励的日趋减少，交易费在矿工收入的**占比将会逐渐增加**，尤其到**2140年之后所有的矿工奖励将完全由交易费决定**。

4.5 区块链应用

- 激励机制研究设计

激励机制分类



本章小结

本章主要围绕区块链基础展开。

- 第一节**密码基础**主要介绍了密码学、古典密码学、对称密码学、AES、非对称密码学、RSA算法、椭圆曲线算法、数字签名算法等基础知识；
- 第二节**结构基础**主要从Hash函数、SHA256函数、Hash指针、链式结构、梅克尔数、区块结构等方面介绍了区块链的基本结构
- 第三节**网络基础**介绍了对等网络及其代表应用、网络结构、路由算法。
- 第四节**激励基础**从博弈论、囚徒困境、纳什均衡、帕累托最优、区块链应用展开介绍。

北京大学
PEKING UNIVERSITY

感 谢 观 看