區塊鏈生態

區塊鏈的起源

記帳方式演進

- 記帳的目的?
- 記憶 → 畫圖
- 衍生信任問題

陸5大會計事務所之一「瑞華」爆醜聞 禍及35家企業IPO

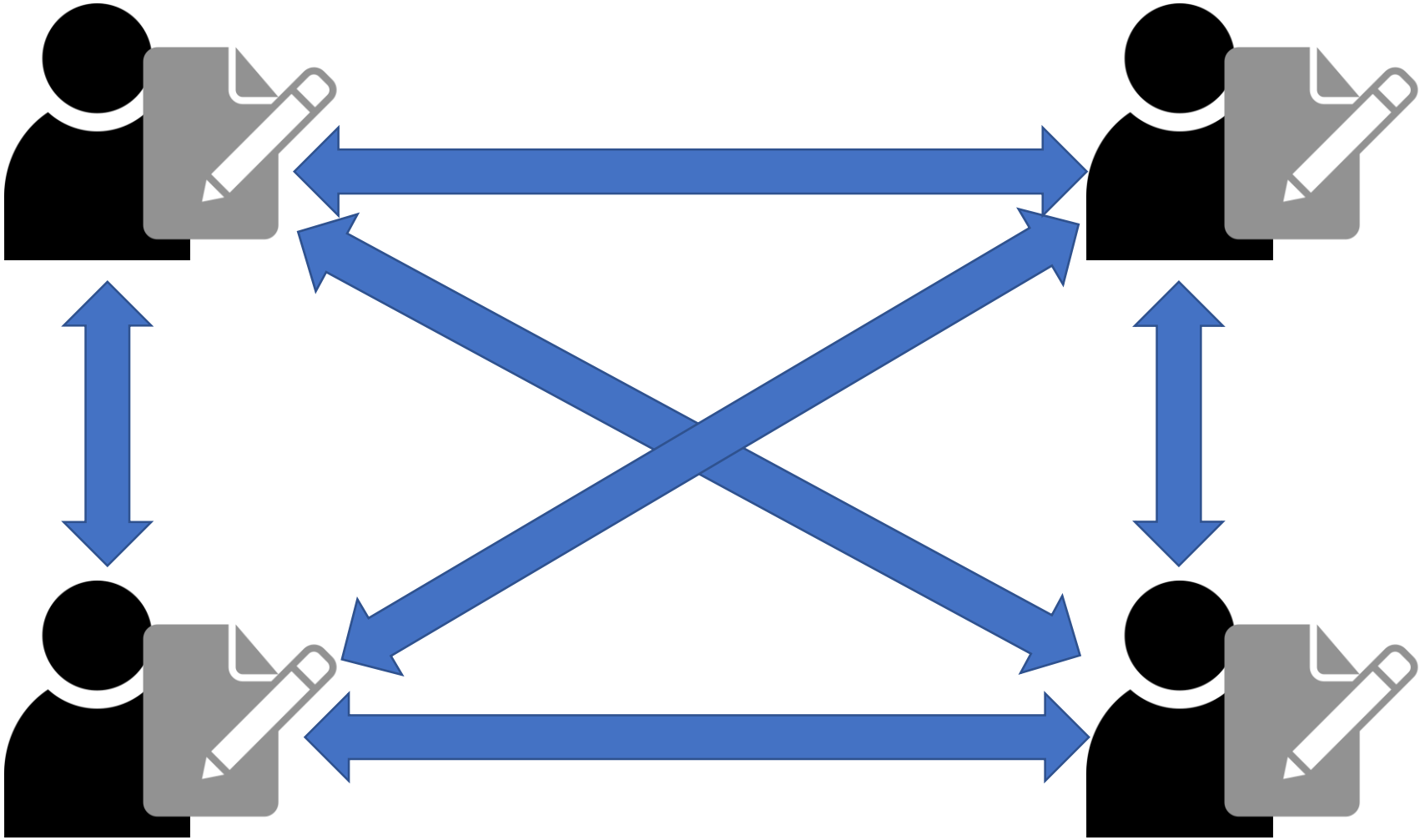
分享 分享 留言 列印 存新聞 A- A+

2019-07-29 15:52 聯合報 記者林宸誼／即時報導 讚 36 分享

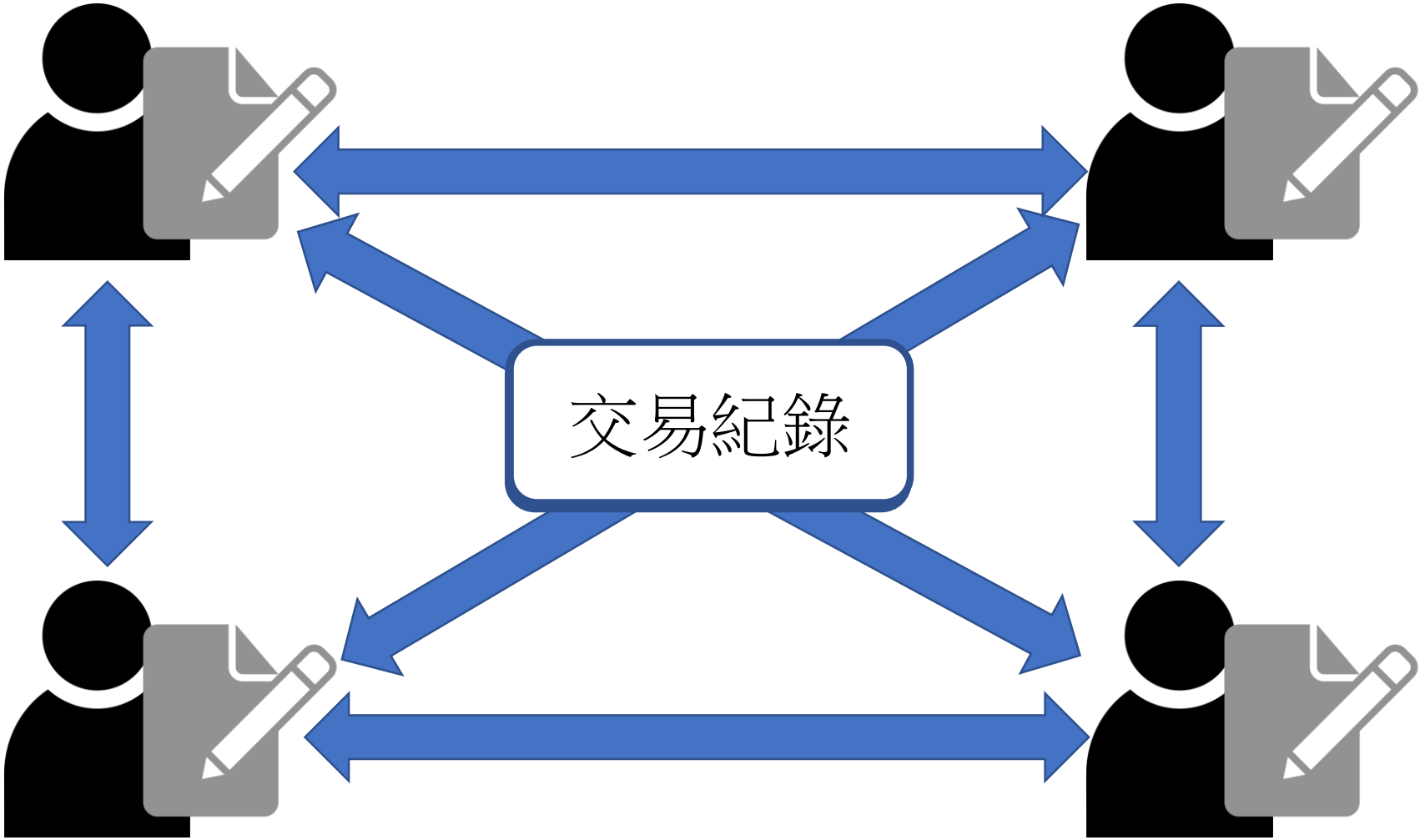
分散式帳本

→ 複式記帳法

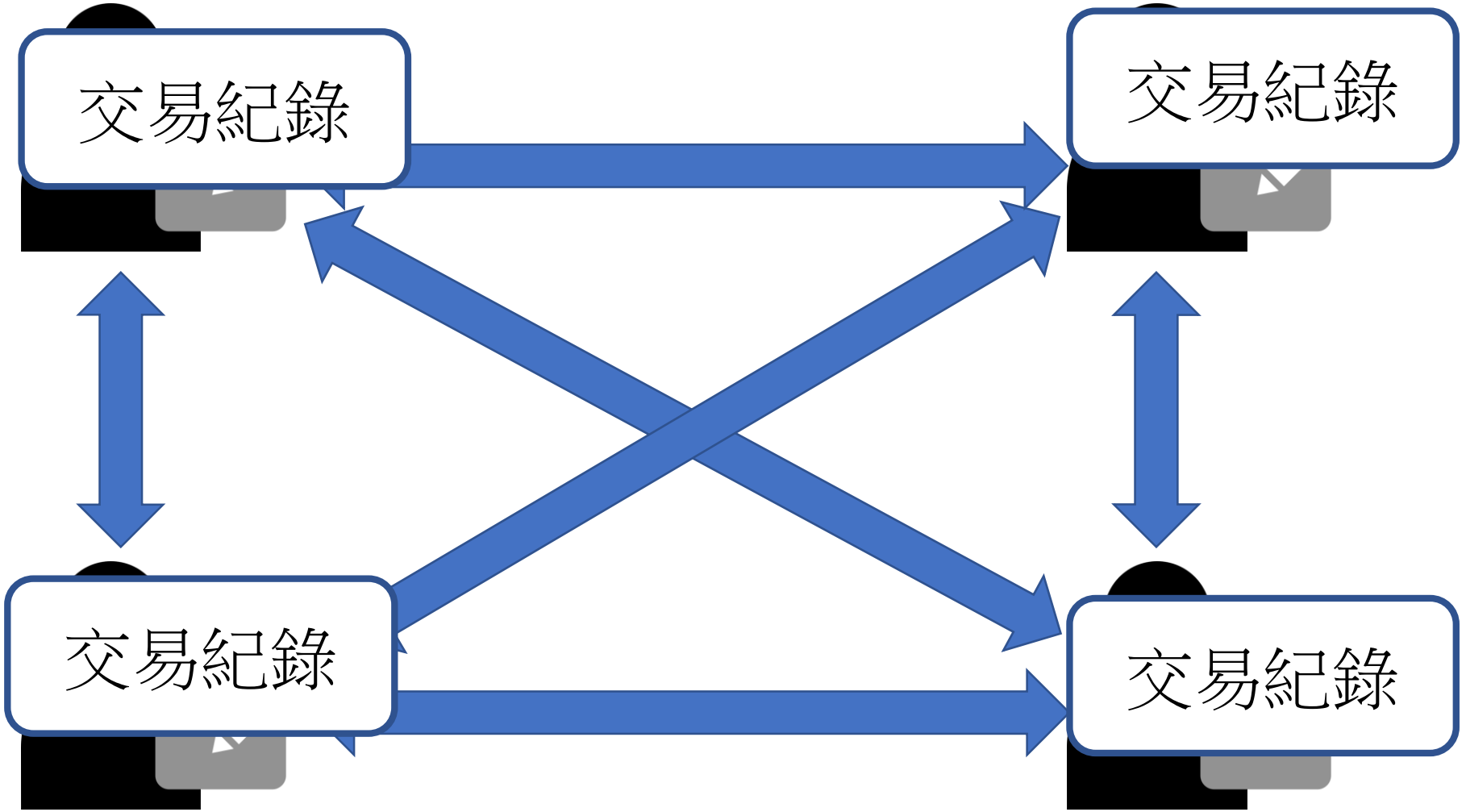
分散式帳本



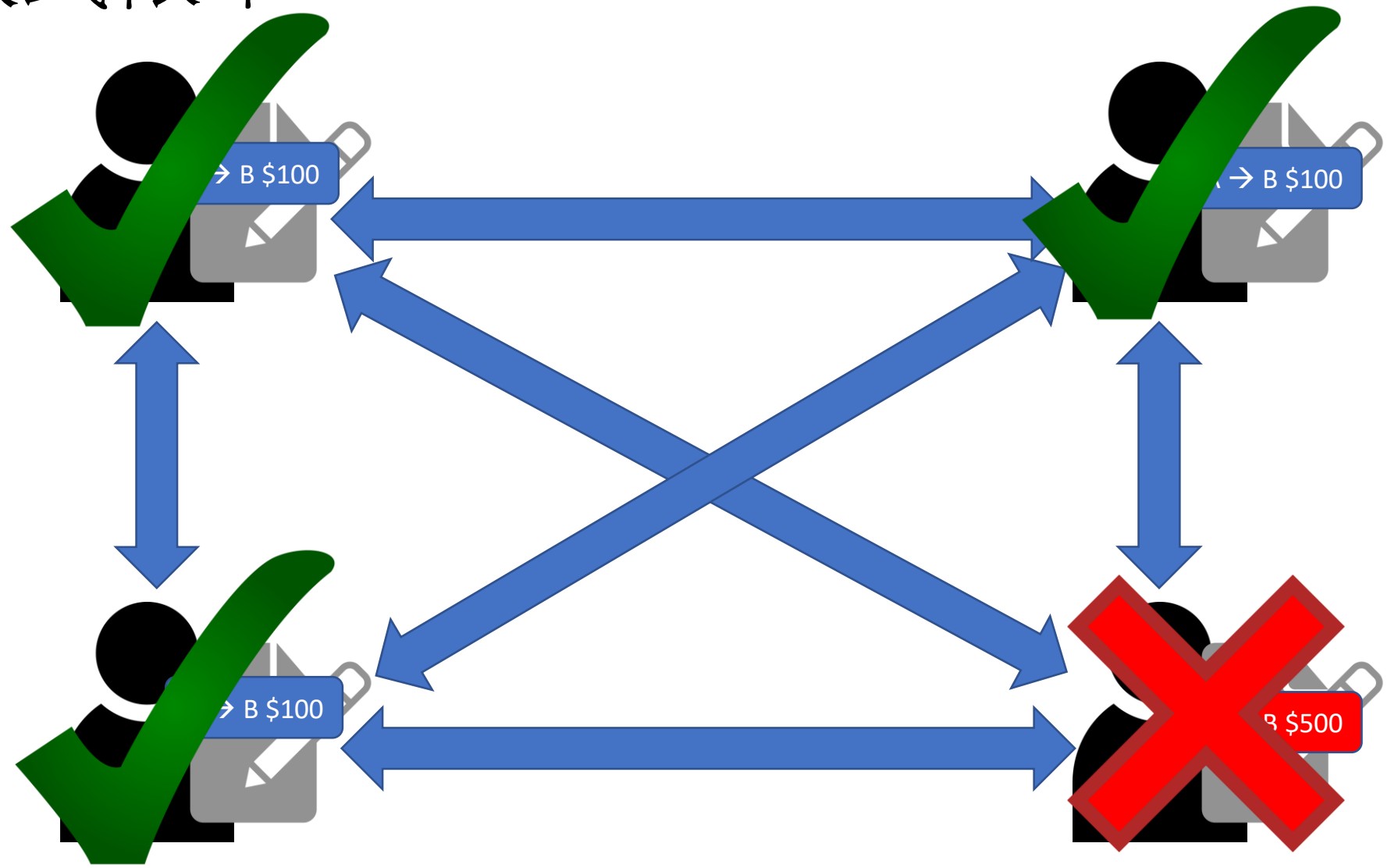
分散式帳本



分散式帳本



分散式帳本



分散式帳本

- 不存在中央權威 (去中心化)
- 所有人的帳本都長一樣
- 不可竄改

去中心化

- 在區塊鏈網絡中每人都有一本
- 公正紀錄每個人擁有的幣數量

不可篡改

- 交易一旦經驗證後上鏈，就不可對交易本身進行竄改
- 愈久以前的交易，被篡改的可能愈低

區塊鏈的特性

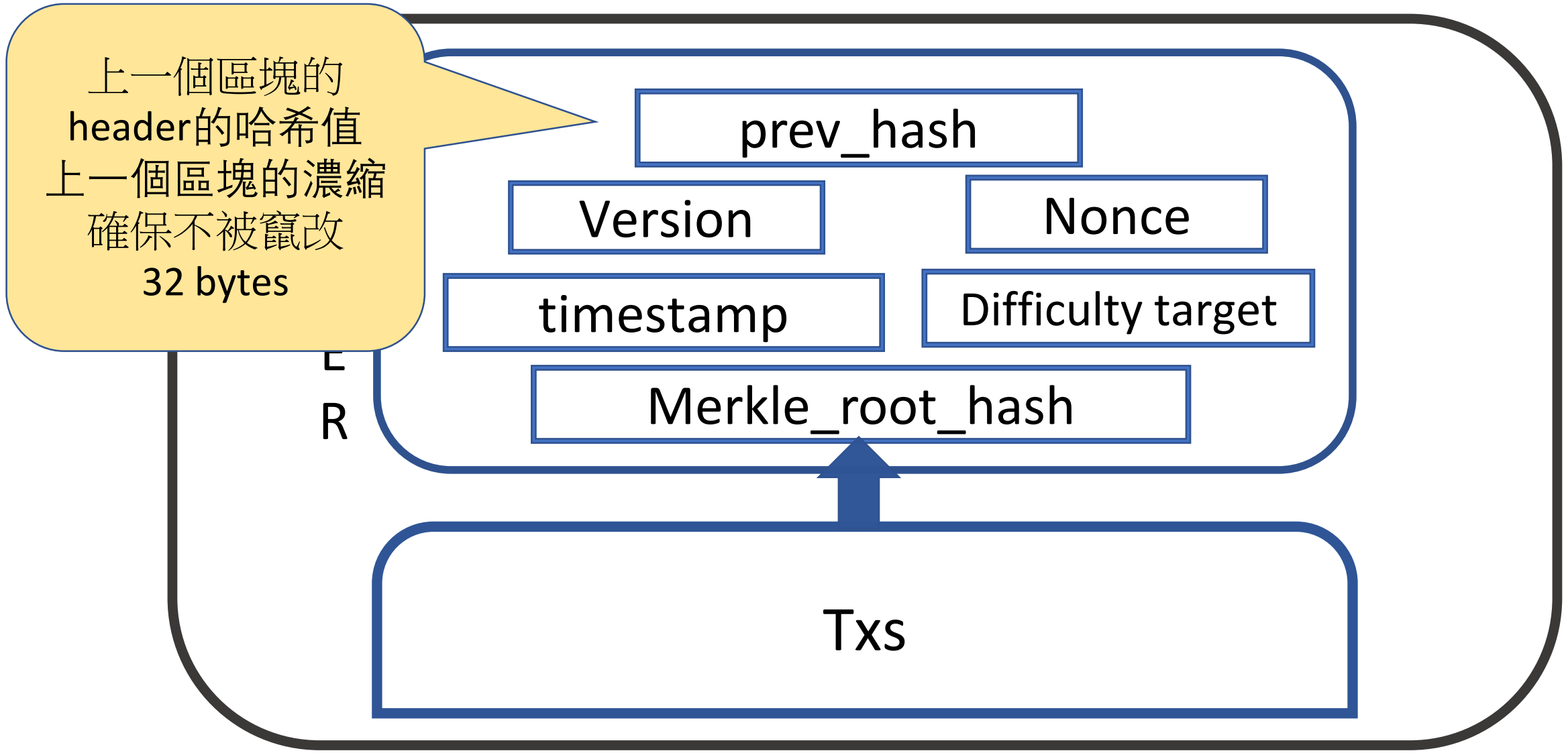
- 去中心化
- 不可篡改

區塊鏈的運作原理

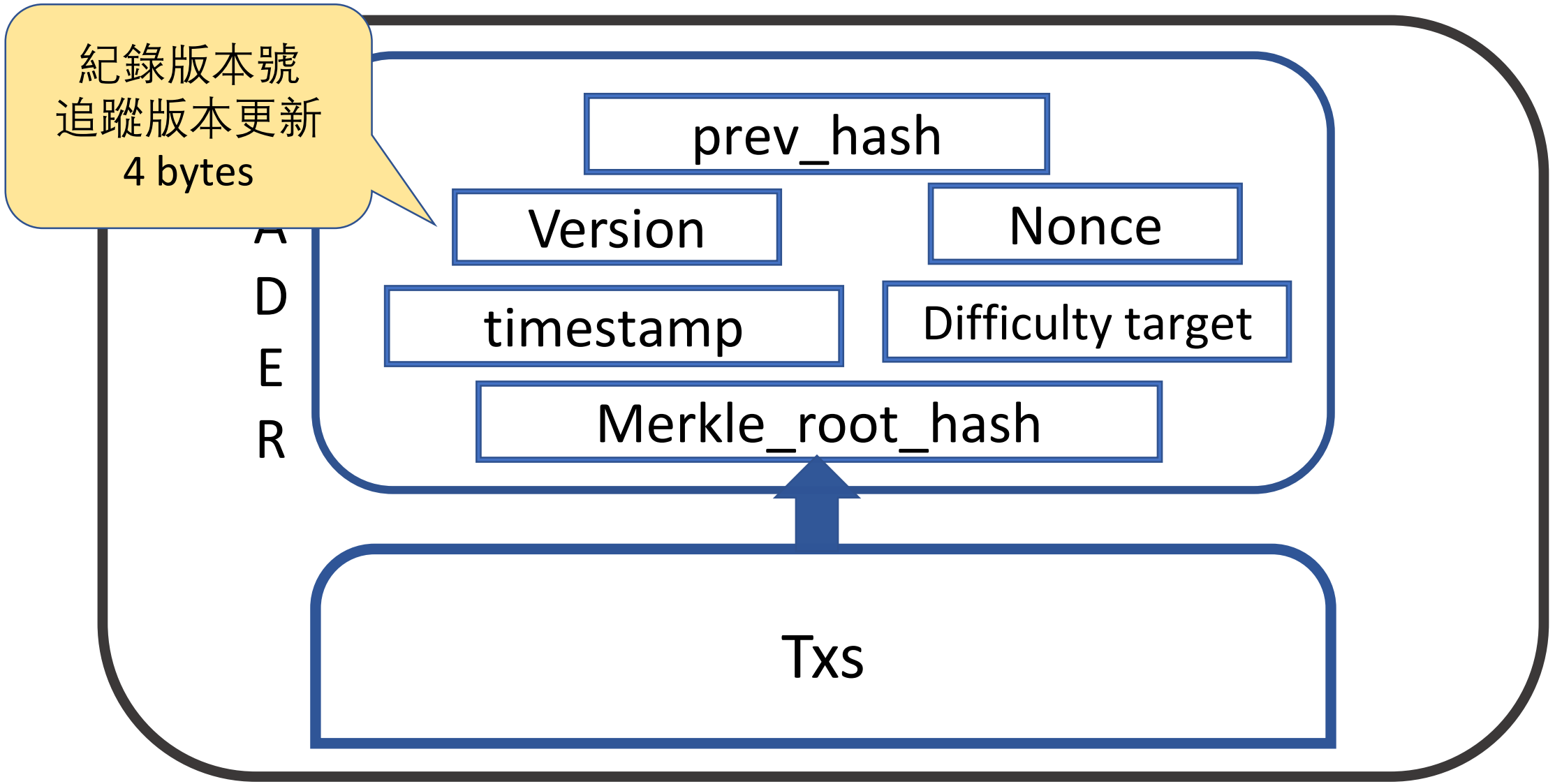
區塊到底是什麼？

- 容器資料結構，用於彙總包含在公共賬本（區塊鏈）中的交易。

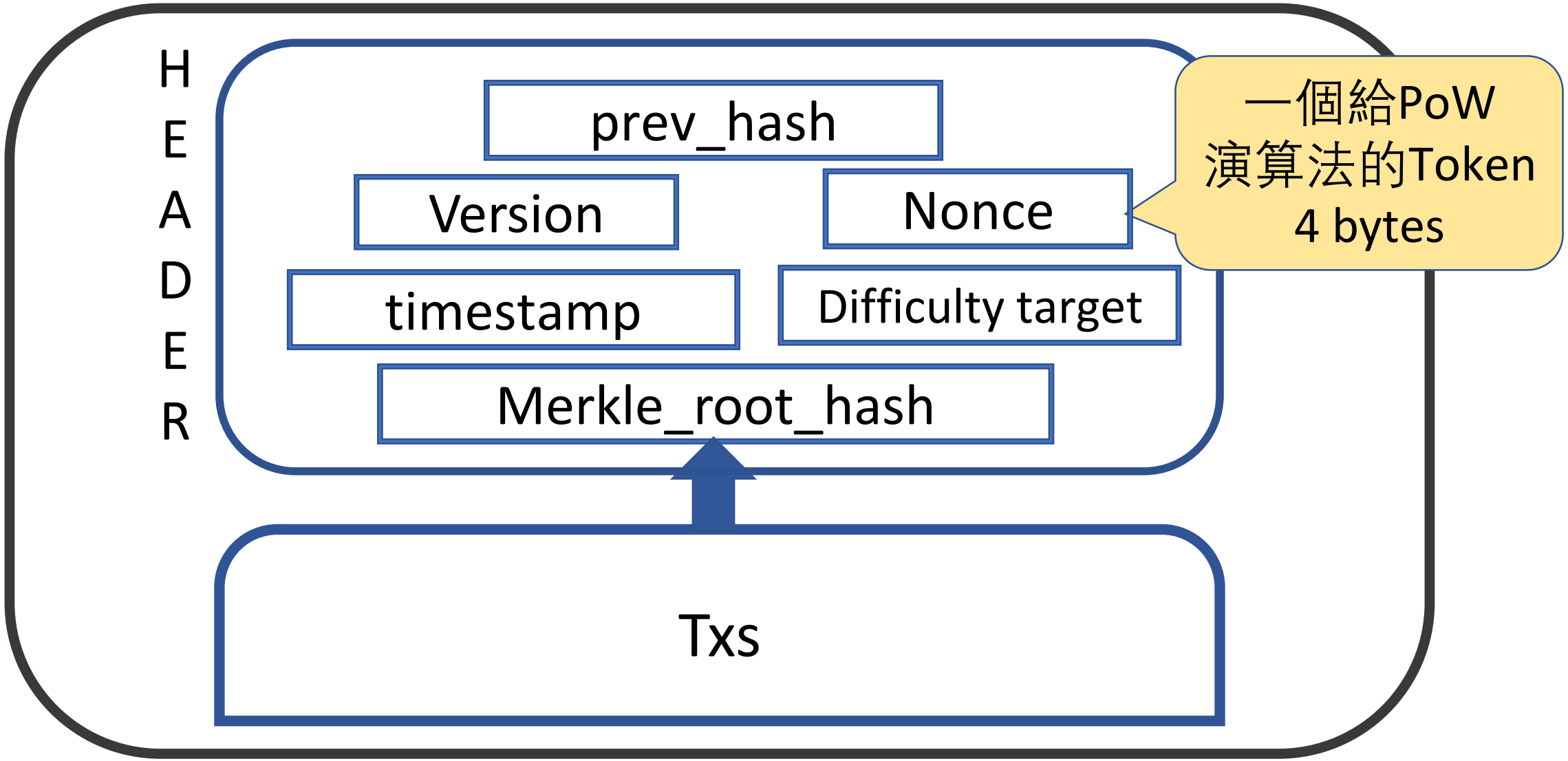
區塊內部有什麼


上一個區塊的
header的哈希值
上一個區塊的濃縮
確保不被竄改
32 bytes
prev_hash
Version
Nonce
timestamp
Difficulty target
Merkle_root_hash
Txs

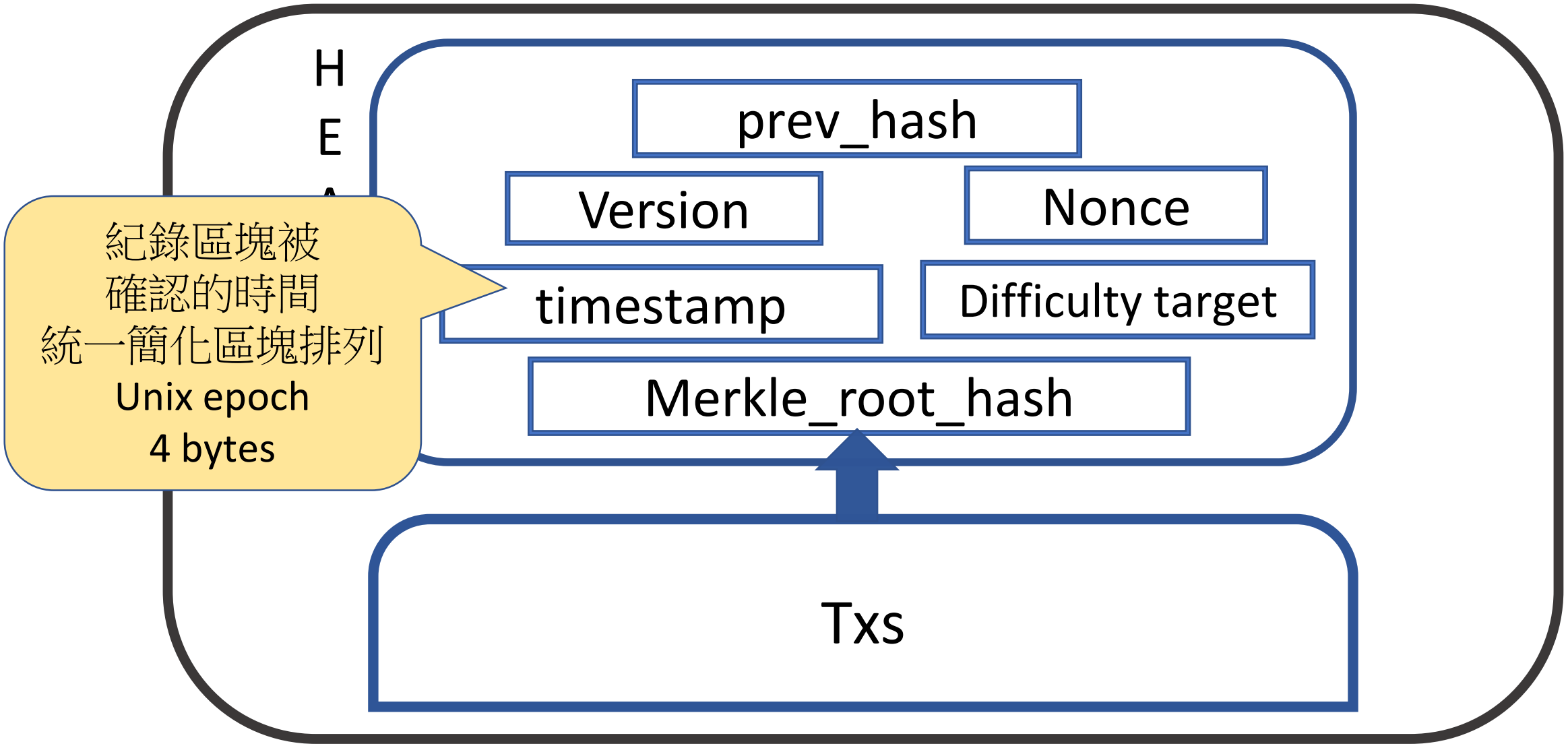
區塊內部有什麼


紀錄版本號
追蹤版本更新
4 bytes
A
D
E
R
prev_hash
Version
Nonce
timestamp
Difficulty target
Merkle_root_hash
Txs

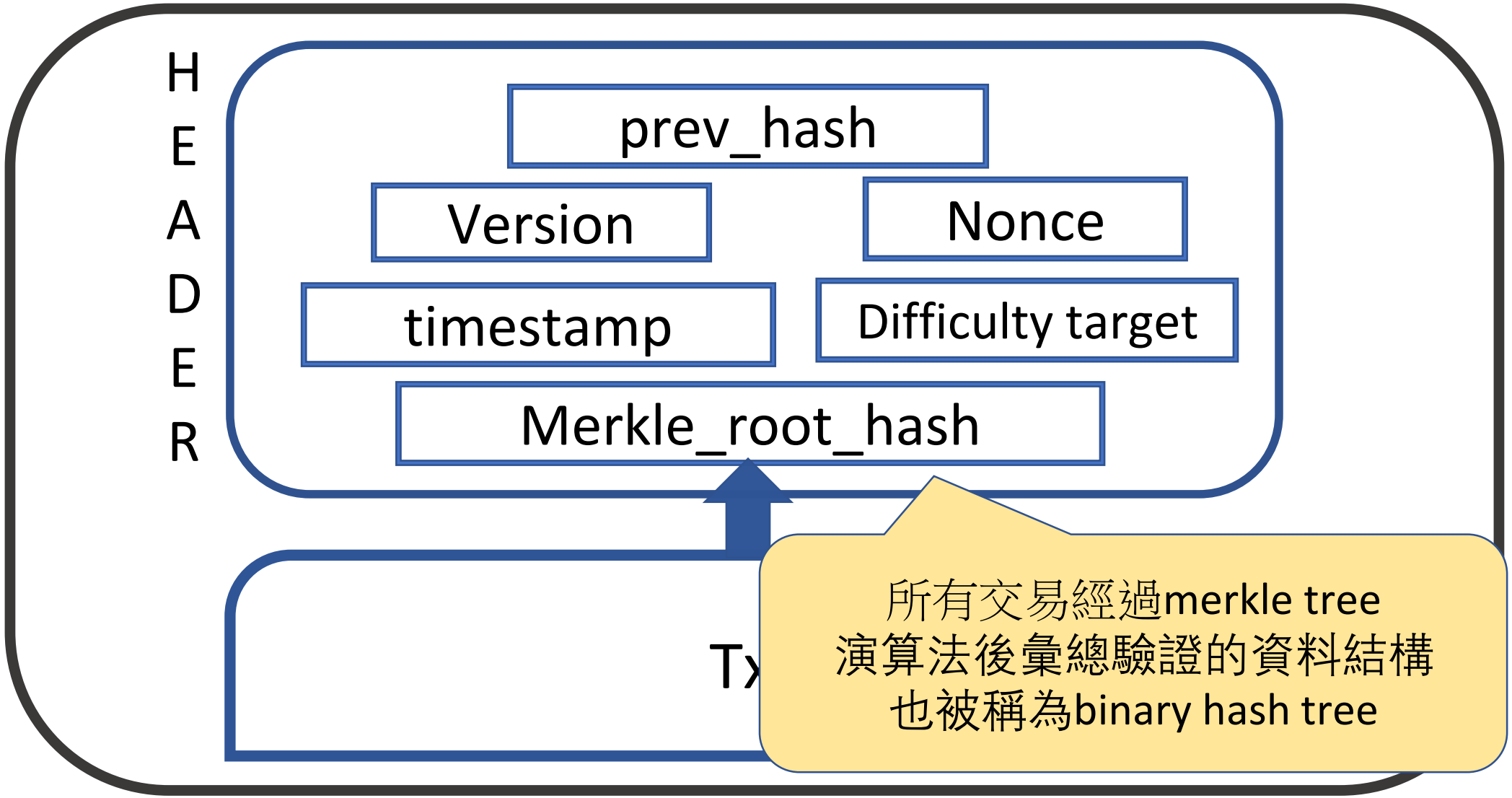
區塊內部有什麼


HEADER
prev_hash
Version
Nonce
timestamp
Difficulty target
Merkle_root_hash
Txs
一個給PoW
演算法的Token
4 bytes

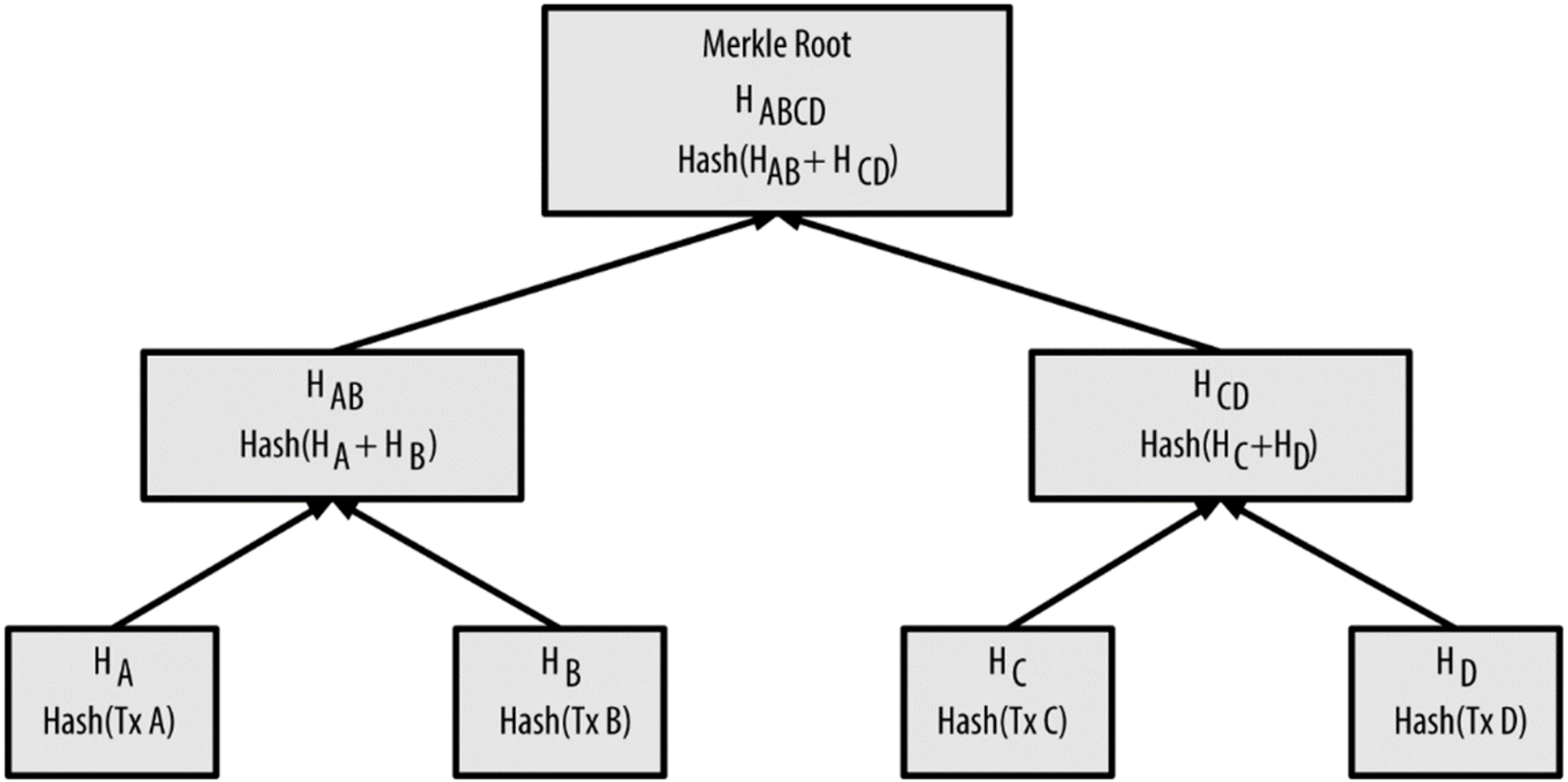
區塊內部有什麼


H
E
prev_hash
Version
Nonce
timestamp
Difficulty target
Merkle_root_hash
Txs
紀錄區塊被
確認的時間
統一簡化區塊排列
Unix epoch
4 bytes

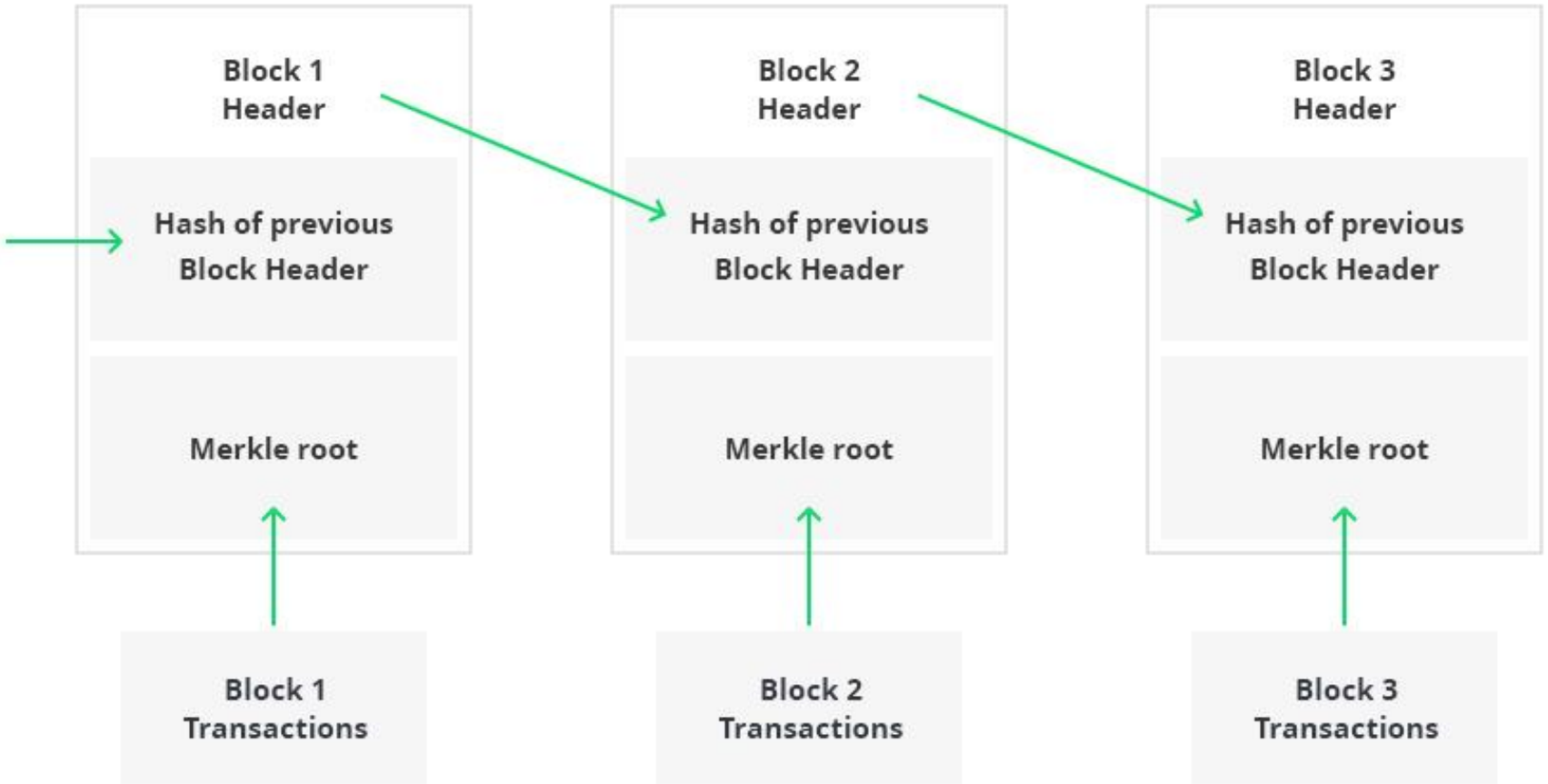
區塊內部有什麼


HEADER
prev_hash
Version
Nonce
timestamp
Difficulty target
Merkle_root_hash
Tx
所有交易經過merkle tree
演算法後彙總驗證的資料結構
也被稱為binary hash tree

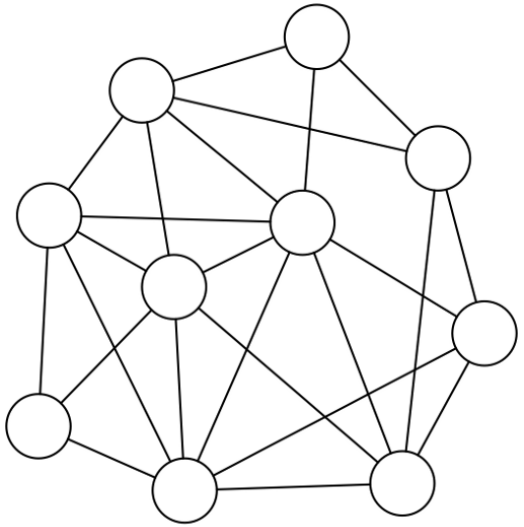
Merkle tree


Merkle Root
H_{ABCD}
$Hash(H_{AB} + H_{CD})$
H_{AB}
$Hash(H_A + H_B)$
H_{CD}
$Hash(H_C + H_D)$
H_A
Hash(Tx A)
H_B
Hash(Tx B)
H_C
Hash(Tx C)
H_D
Hash(Tx D)

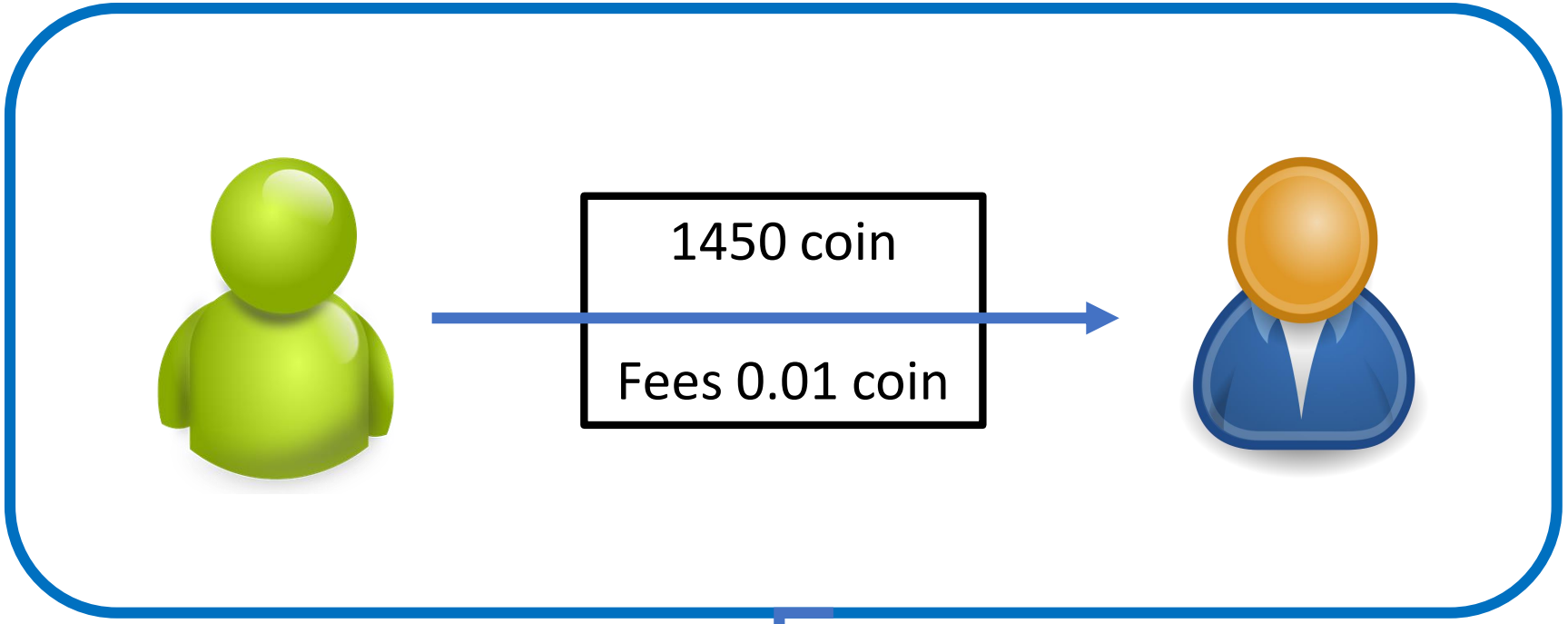
Whole view of relation between blocks

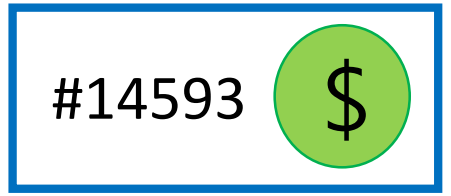


區塊鏈中的交易



Blockchain

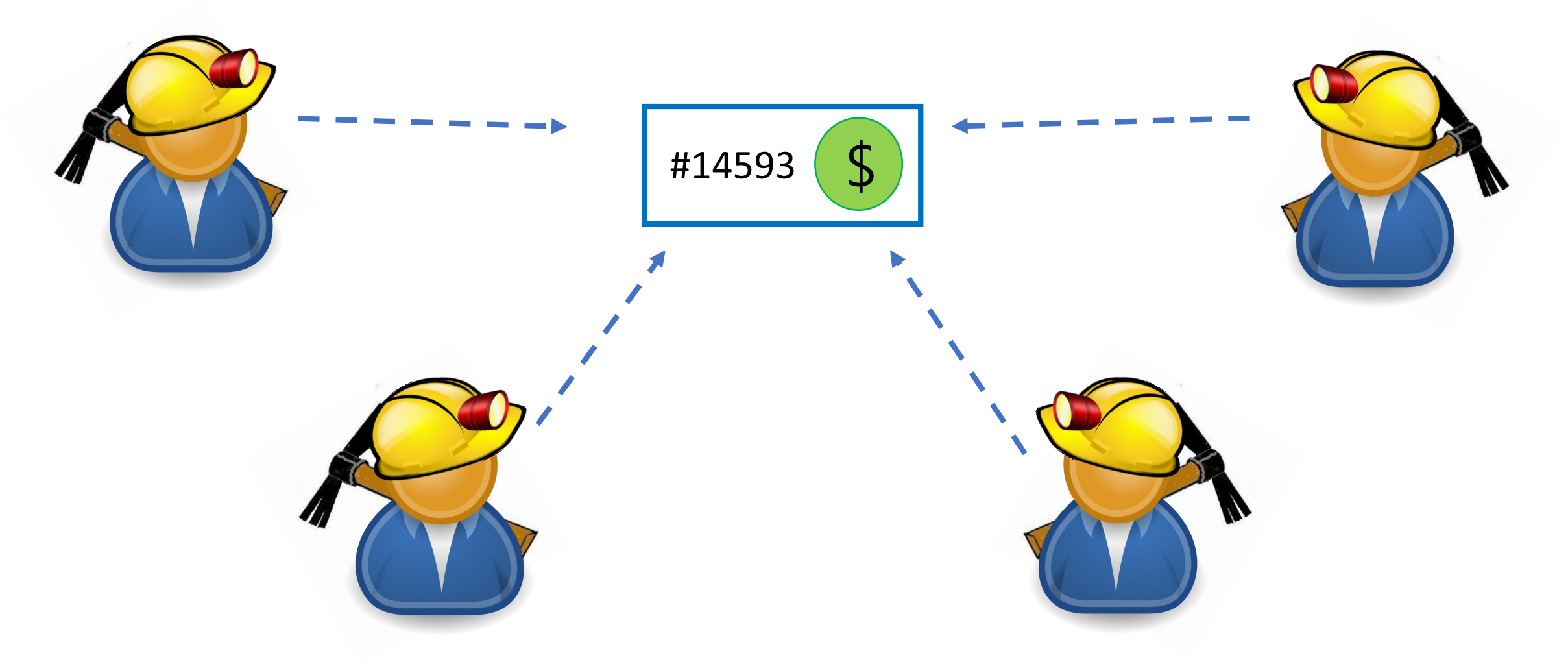




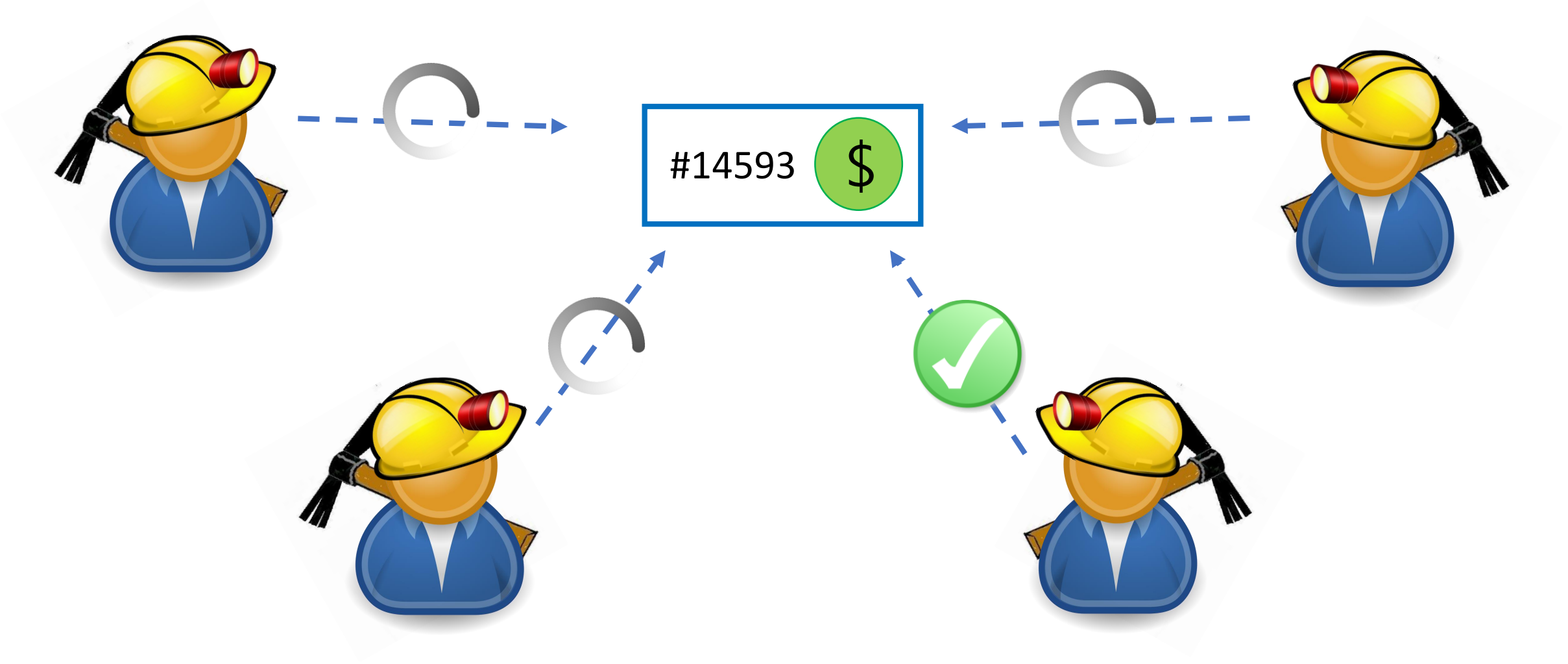
礦工

- 用自身電腦的算力來驗證區塊
- 第一個驗證成功的節點可以獲得獎勵
- 解決基於密碼雜湊演算法的數學難題 (PoW)
- 礦池

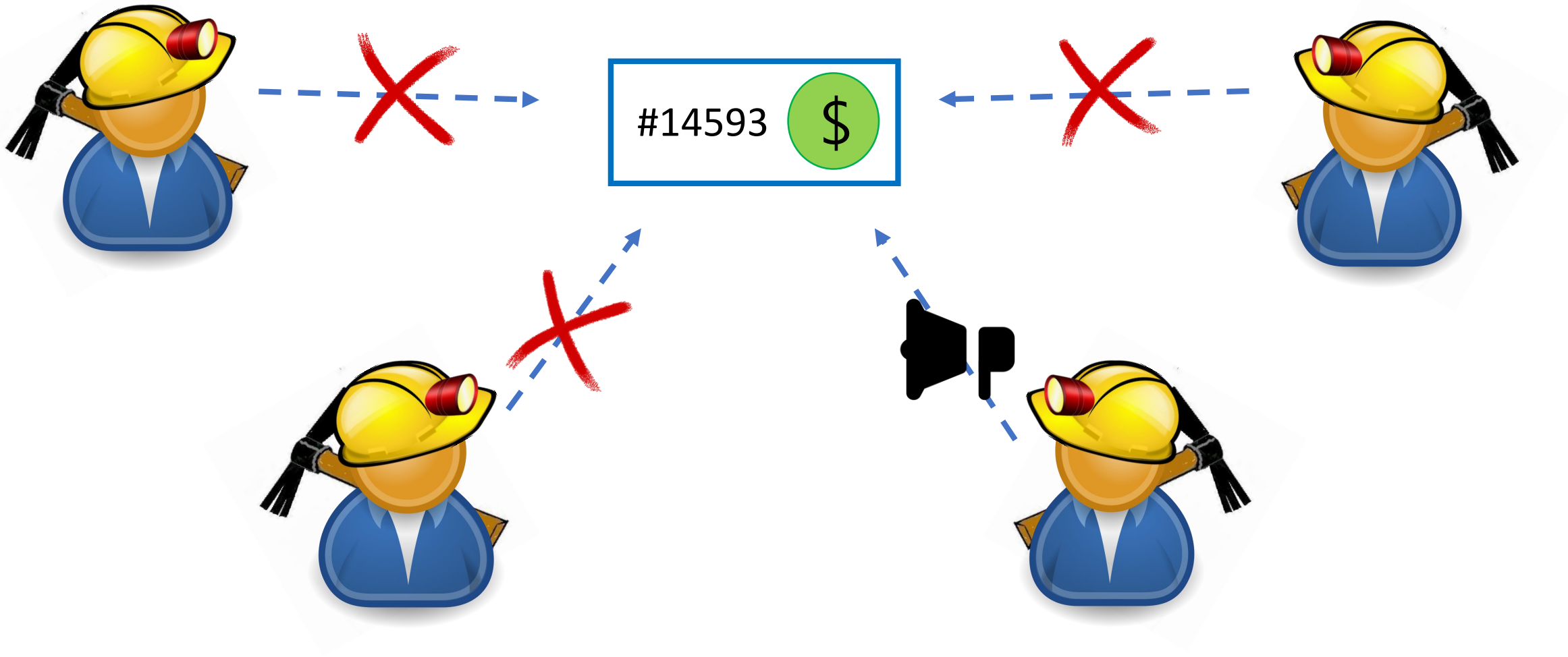
區塊鏈中的交易



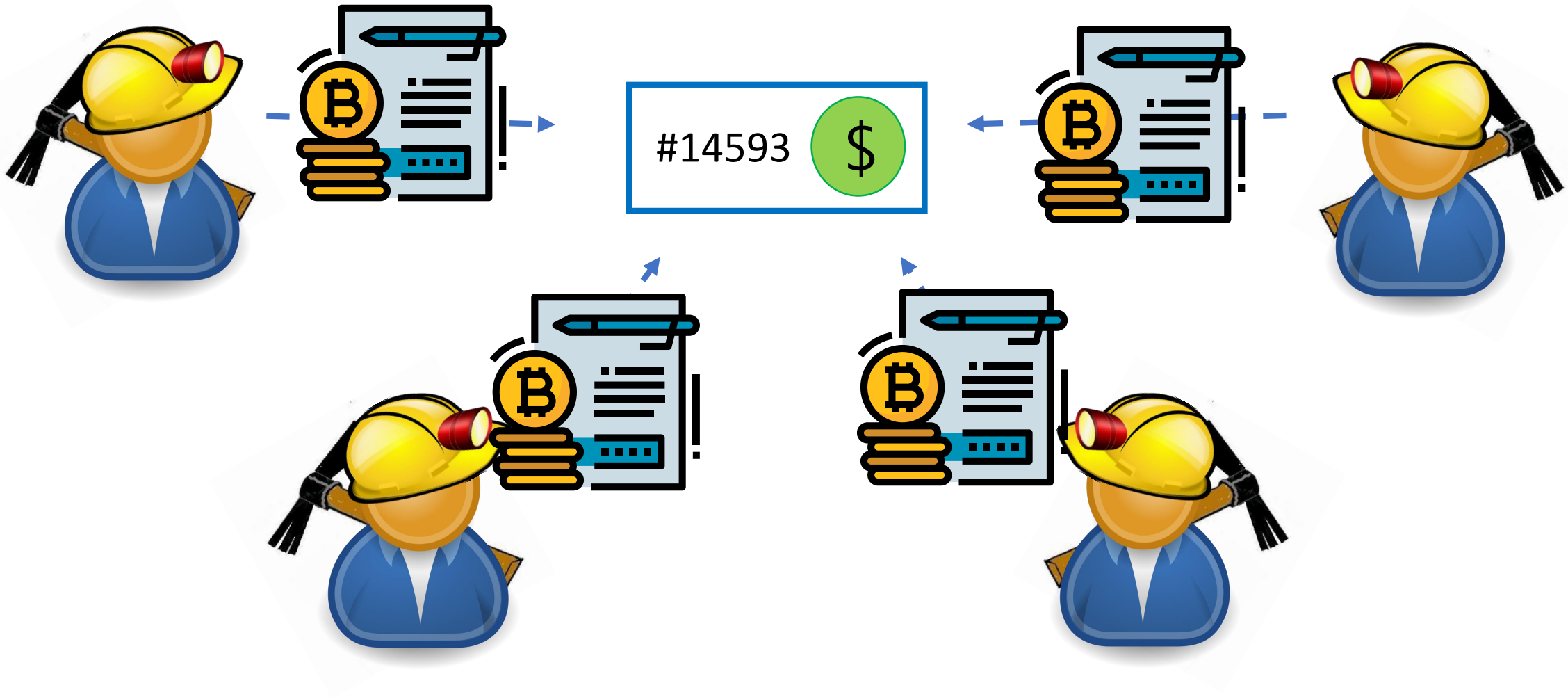
區塊鏈中的交易



區塊鏈中的交易



區塊鏈中的交易



區塊鏈中的交易

Coinbase Transaction

- 由礦工創建的交易，主要是為了獎勵進行PoW挖礦的努力
- 生成新的比特幣

Try it yourself!

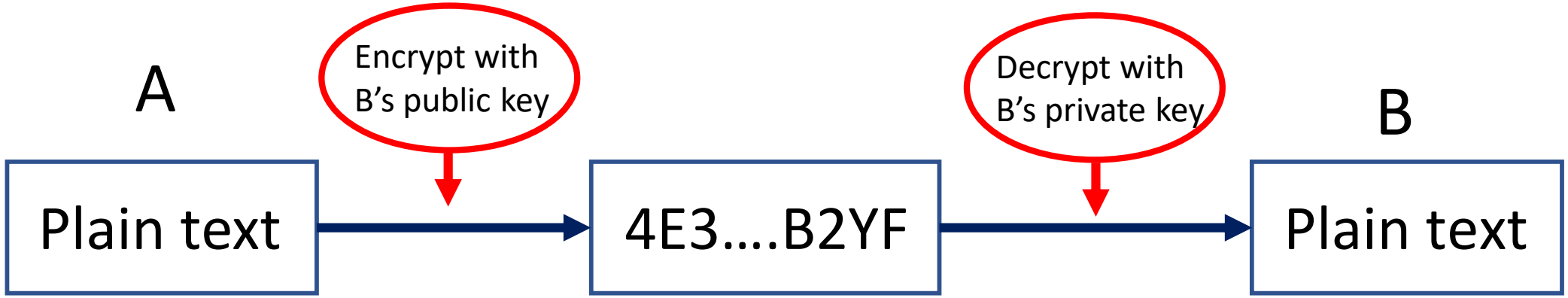
- https://anders.com/blockchain/

錢包

- 一個區塊鏈的錢包中必定會有三大要素：
  - 公鑰
  - 私鑰
  - 地址

公鑰與私鑰 --- 非對稱加密

- 可以公鑰加密私鑰解密，也可以私鑰加密公鑰解密
- 一定會有一個以上的密碼對



私鑰

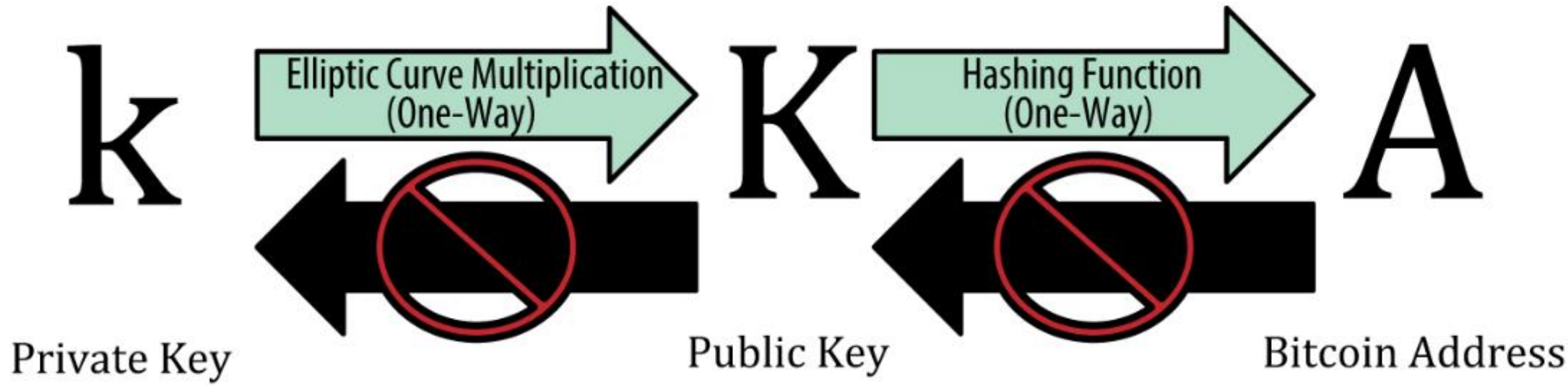
- Private Key
- 非公開
- 使用 SHA-256 算法隨機生成 32 bytes（256bits）的隨機數
- 能夠證明對該比特幣地址的所有資金具有的所有權及控制權
- 支付的時候，擁有者就必須要使用私鑰為交易"簽名“

公鑰

- Public Key
- 公開
- 基於私鑰對應生成的
- 為交易的有效性進行驗證

地址 Address

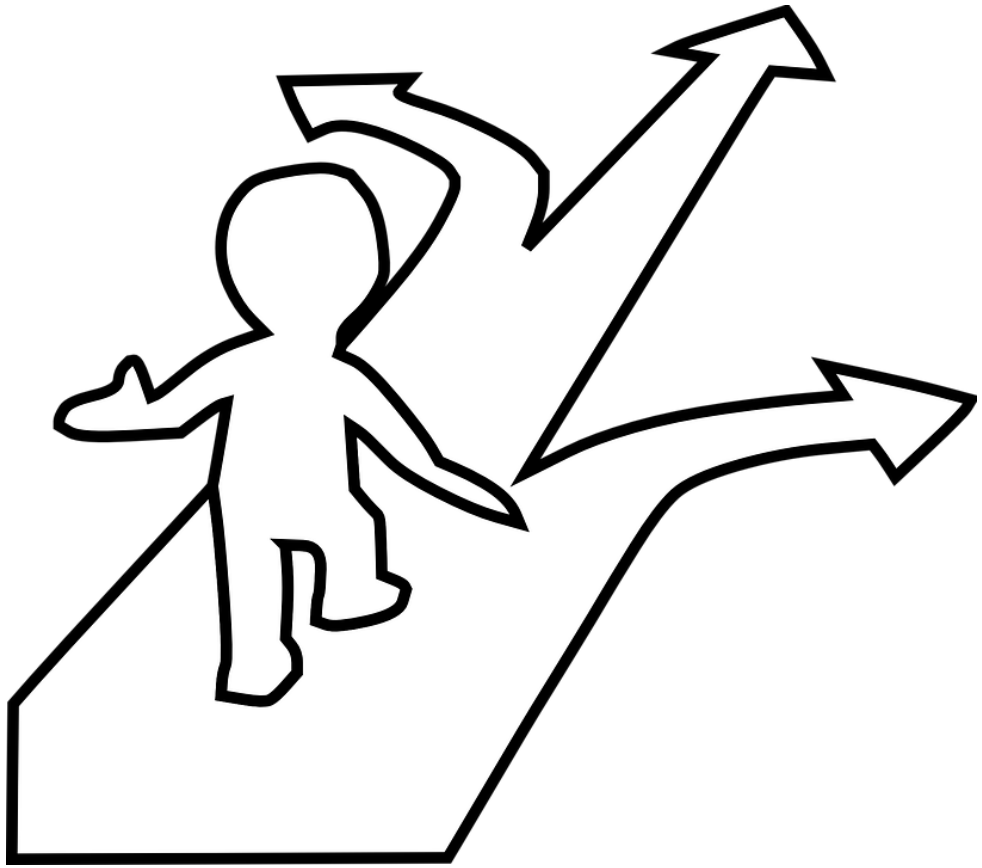
- 交易中最常見的資金“接收者”地址
- 公鑰單向加密雜湊而來的
- SHA256和RIPEMD160


k
Elliptic Curve Multiplication
(One-Way)
K
Hashing Function
(One-Way)
A
Private Key
Public Key
Bitcoin Address

分岔

分岔是甚麼?

- 分岔發生在當版本更新時，部分人選擇更新，部分人不更新



硬分岔

- 升級後的版本不能驗證未升級的版本的區塊，反之亦然
- 沒有向前兼容性
- 變成兩條鏈，舊鏈與新鏈
- 需要在某時間點全部同意分岔升級，不同意地將留在舊鏈。

軟分岔

- 升級了的版本可以驗證已經升級的節點生產出來的區塊，反之亦然
- 有較好兼容性

- 沒有分岔鏈

- 可讓舊區塊與新區塊共存

重大分岔事件

區塊鏈的型態

- 公有鏈
- 私有鏈
- 聯盟鏈

公有鏈

- 訪問門檻低
- 所有數據公開
- 沒有人能操弄用戶的資料

私有鏈

- 交易速度非常快
- 相較公有鏈有更好的隱私
- 交易成本大幅下降

聯盟鏈

- 部分去中心化

共識機制

Proof of Work (PoW)

- 大量進行嘗試計算，計算時間取決於機器的hash運算速度
- 礦工拿到代幣作為挖礦獎勵
- 機制高，可監管性弱
- 性能效率低
- 49%容錯

PoW pros and cons

- Pros

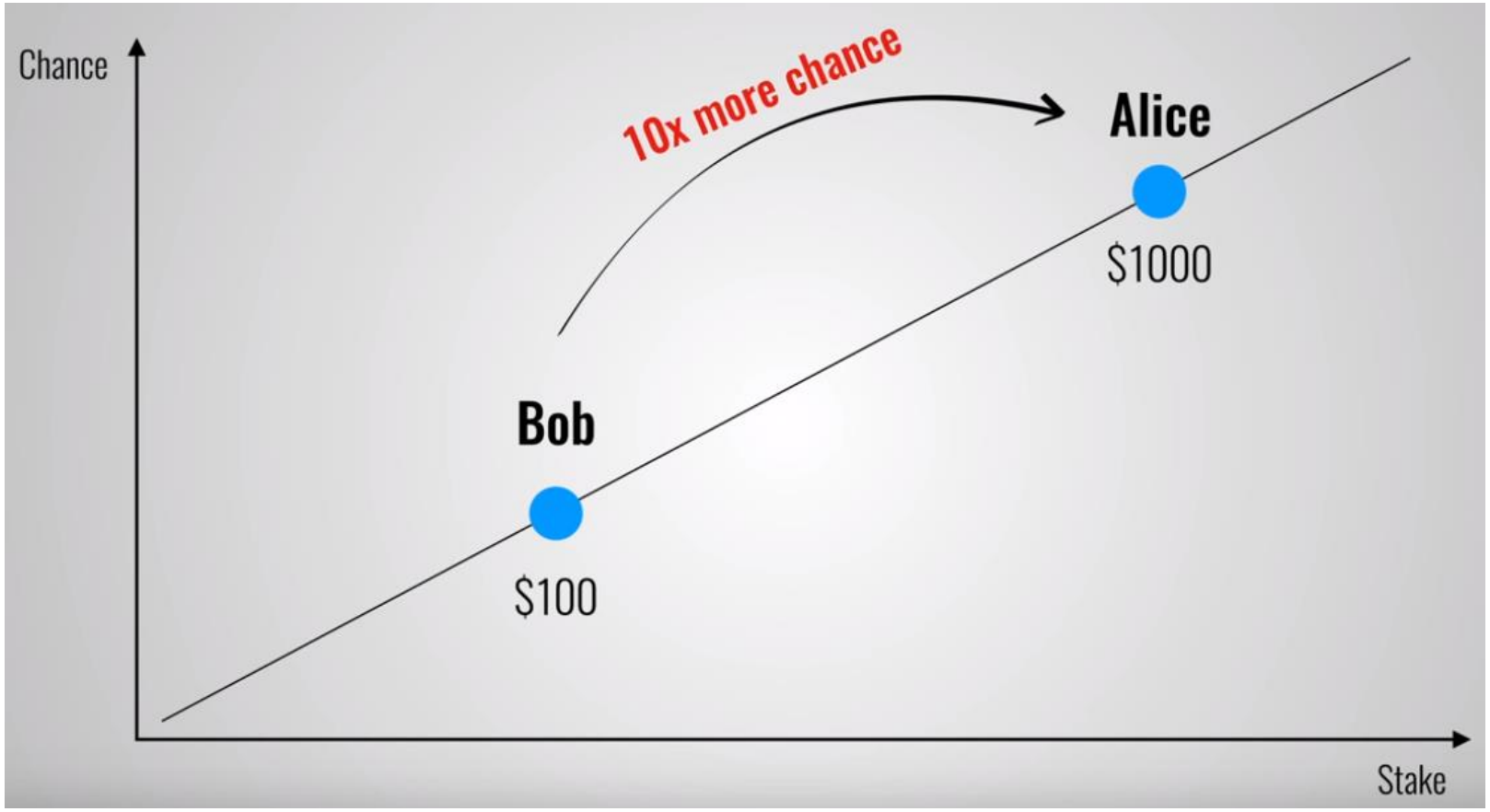
- Cons :
  - 嘗試運算耗費相當大的無謂資源
  - 由於礦場的存在，以及算力的買賣，富者恆富貧者恆貧
  - 容易發生 51% 攻擊

Proof of Stake

- 礦工 (Miner) → 驗證者 (Validator)
- 挖礦 (Mine) → 創造 (Forge)

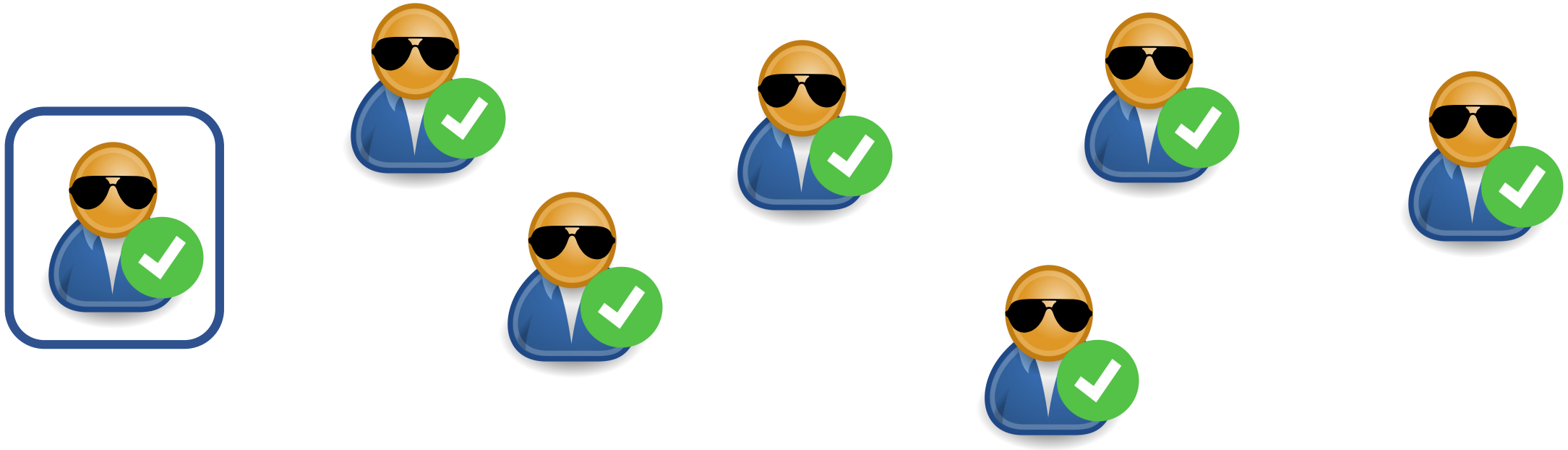
Proof of Stake Mechanics

- 抵押一定數量的 Token 來成為驗證者



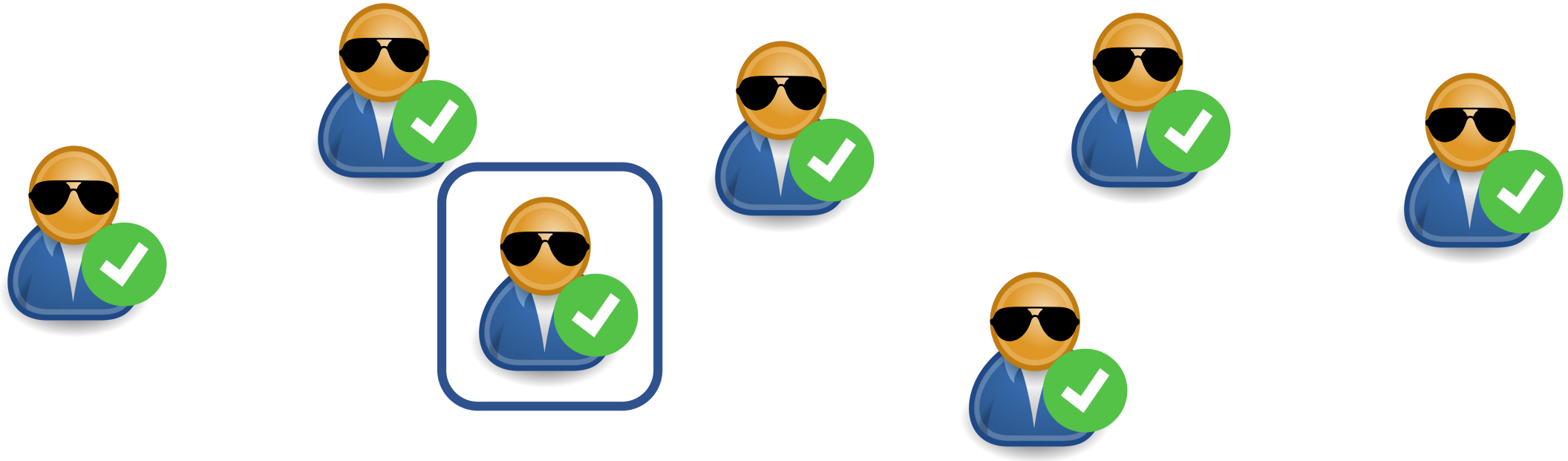
Proof of Stake Mechanics

- 隨機選取一位驗證者來驗證下一個區塊



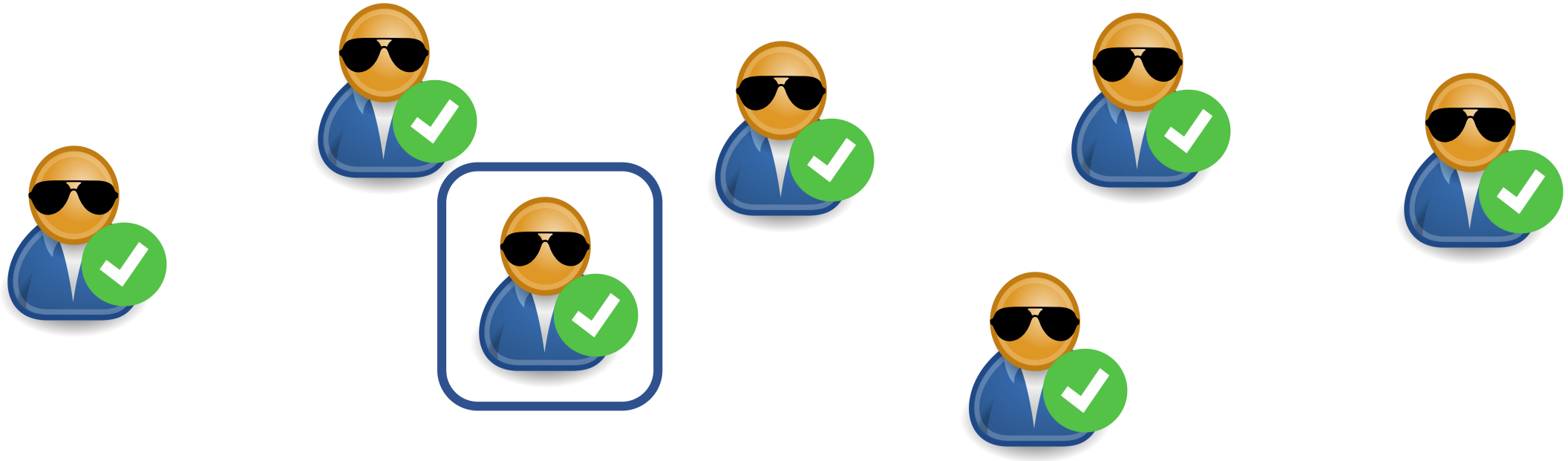
Proof of Stake Mechanics

- 隨機選取一位驗證者來驗證下一個區塊
- 驗證者可以取得所有區塊中的手續費作為獎勵



Proof of Stake Mechanics

- 如何信任驗證者?



PoS pros and cons