# Bitcoin and Blockchain Technology

# Fundamental Challenges of a Digital Currency

- Can I trust that the money is authentic and not counterfeit?

- Can I trust that the digital money can only be spent once (known as “double spend problem”)?

- Can I be sure that no one else can claim this money belongs to them and not me?

# History of Digital Currencies Leading up To Bitcoin

- Began in Late 1980s:
  - A currency that could be sent untraceably
  - Did not require central authorities (Banks)
- Bit Gold (1998): Could never solve the double spending problem
- DigiCash (1990s) : Advanced use of public and private key cryptography as a way to send electronic transactions

# Bitcoin (2008)

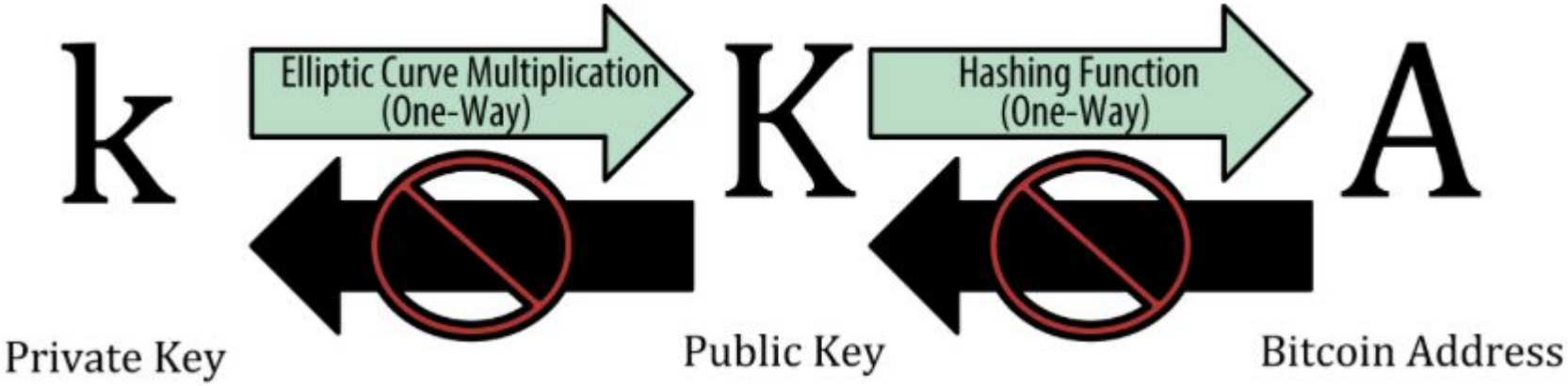
- A set of rules for independent transaction validation and currency issuance (consensus rules)
- A decentralized peer-to-peer network (the bitcoin network)
- A public transaction ledger (the blockchain)
- A mechanism for reaching global decentralized consensus on the valid blockchain (Proof-of-Work Algorithm)

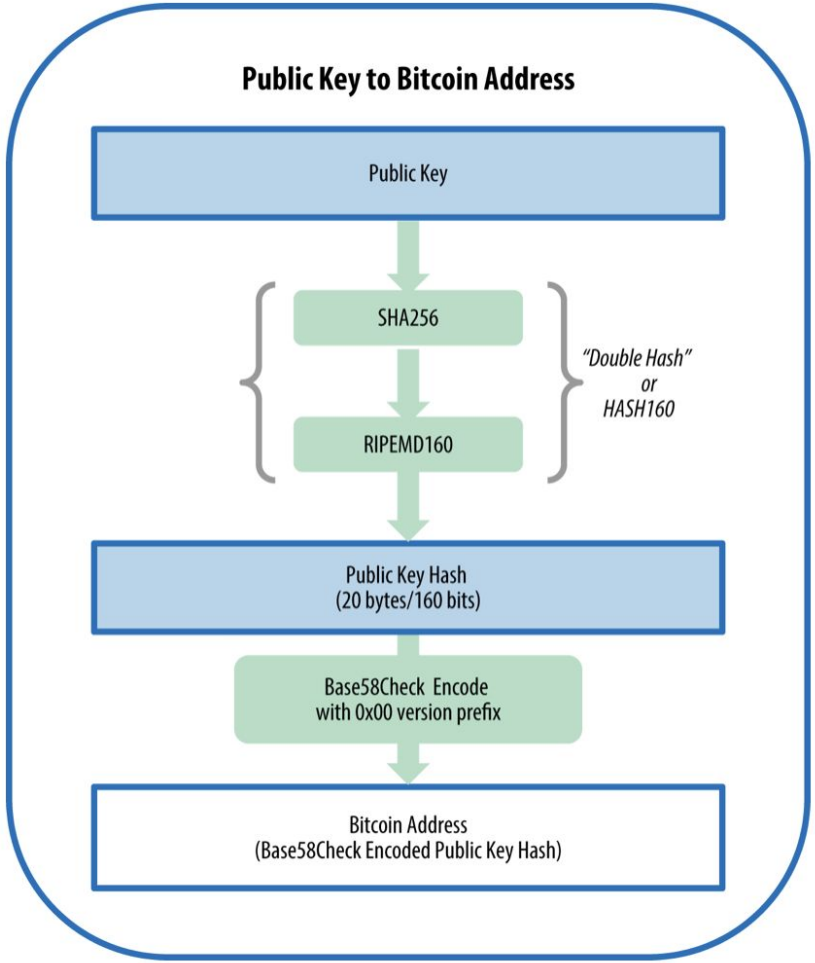
# Transactions

- Private/ Public Keys
- Wallets

# Private/ Public Key

- Ownership of bitcoin is established through:
  - Digital Keys
  - Bitcoin Addresses
  - Digital Signatures
- Digital Keys, or *private* keys, are not stored on the network, but in a file, or *wallet*


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


Public Key to Bitcoin Address
Public Key
SHA256
RIPEMD160
"Double Hash"
or
HASH160
Public Key Hash
(20 bytes/160 bits)
Base58Check Encode
with 0x00 version prefix
Bitcoin Address
(Base58Check Encoded Public Key Hash)

# Digital Signatures

- Signature proves that the owner of the private key, who is is the owner of the funds has authorized the spending of those funds
- Proof of Authorization is undeniable (nonrepudiation)
- Signature Proves that the transaction has not and cannot be modified by anyone else

# Wallets

- Application that serves as a primary user interface
- Bitcoin wallets do not contain coins, they contain keys
  - Keychain containing private/public keys
- Users Control the coins on the network by signing transactions with the keys in their wallet

# Transaction

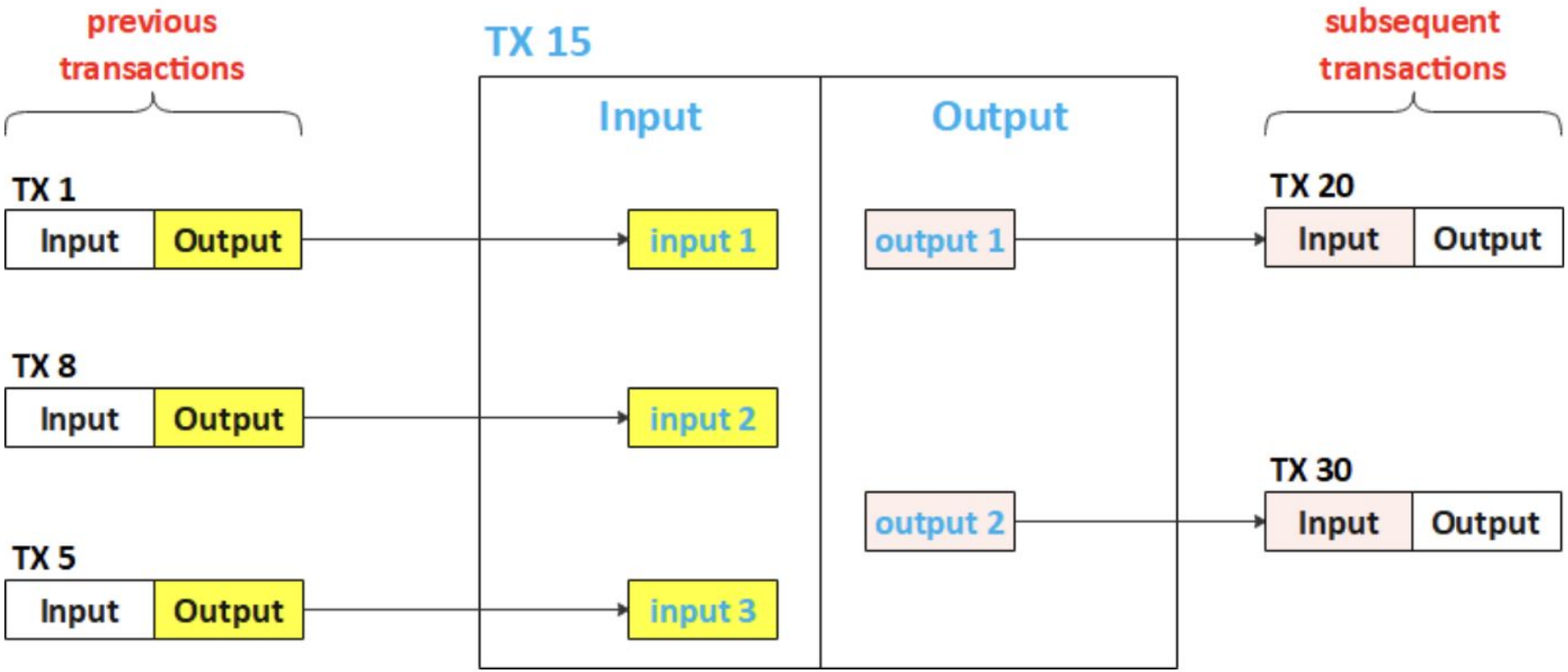
- Transaction Output - indivisible chunks of bitcoin currency, recorded on the blockchain
- UTXO - unspent transaction outputs
  - Available and spendable outputs

# UTXO Model

- Example: Bob wants to buy a soda for $1.50
  - $5 Bill
  - 6 Quarters
- Bitcoin payment must be Created with whatever denominations the user has available
  - If Alice wants to send Bob 1 BTC
    - 0.25 BTC * 4
    - 0.95 + 0.05 BTC
- Transaction consumes previous unspent transactions
  - These are linked to previous transactions incurred
- Wallet application takes care of this abstraction

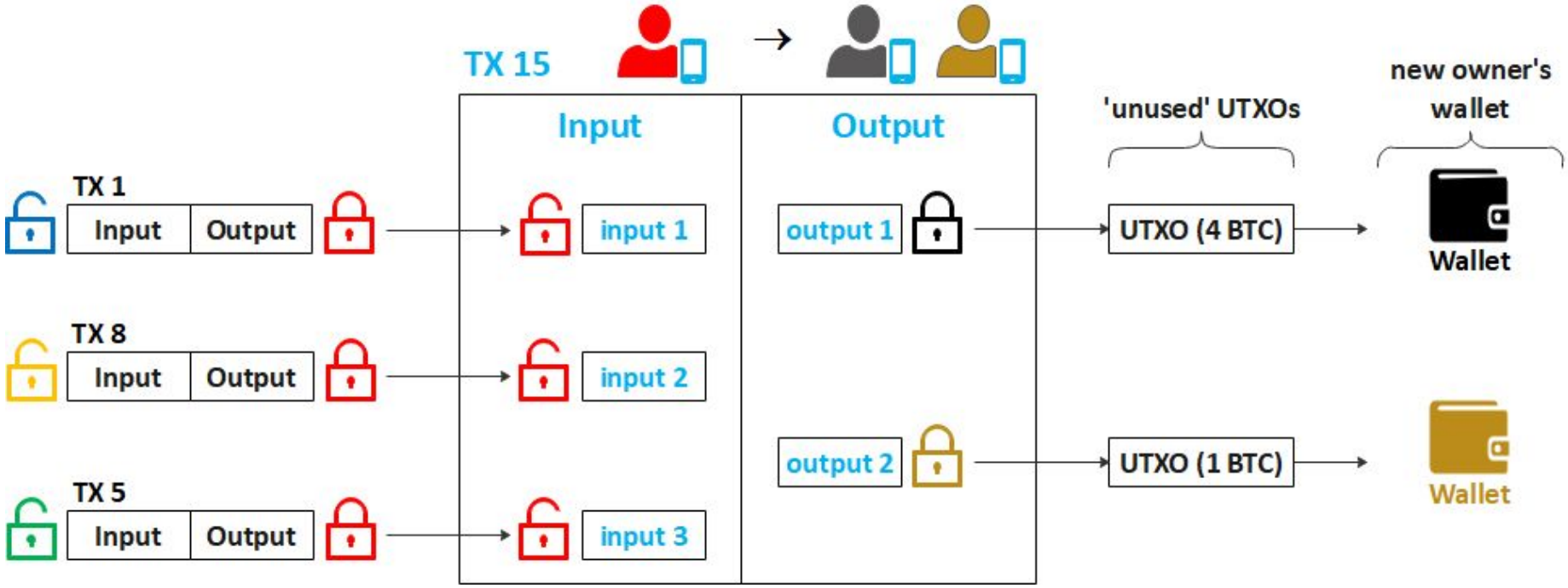
# Transaction Outputs

- Bitcoin transactions create an output
- If Bob sends Alice 0.2 BTC, but only has a full 1 BTC
  - Bob will be given ~0.8 BTC in return in unspent transactions in the output transaction


previous transactions
TX 15
subsequent transactions
Input
Output
TX 1
Input
Output
input 1
output 1
TX 20
Input
Output
TX 8
Input
Output
input 2
output 2
TX 30
Input
Output
TX 5
Input
Output
input 3

# Locking and Unlocking Scripts:

- Unlocking: Transaction consumes existing UTXO (at the input) by unlocking it with the current owner’s signature
  - Condition: Prove she/he is the holder of the designated public key by showing signature
- Locking: The newly generated UTXO (at the output) is locked to the new owners public key/bitcoin address
  - Only holder of the designated public key can claim this bitcoin


TX 15
Input
Output
'unused' UTXOs
new owner's wallet
TX 1
Input
Output
input 1
output 1
UTXO (4 BTC)
Wallet
TX 8
Input
Output
input 2
output 2
UTXO (1 BTC)
Wallet
TX 5
Input
Output
input 3

# Bitcoin Address

Addresses are identifiers which you use to send bitcoins to another

| Summary | |
|---|---|
| Address | 1933phfhK3ZgFQNLGSDXvqCn32k2buXY8a |
| Hash 160 | 582431b9e63d2394c8b224d1bc45d07ae95d2379 |
| Short Link | http://blockchain.info/fb/1933p |
| Tools | Taint Analysis - Related Tags - Unspent Outputs |

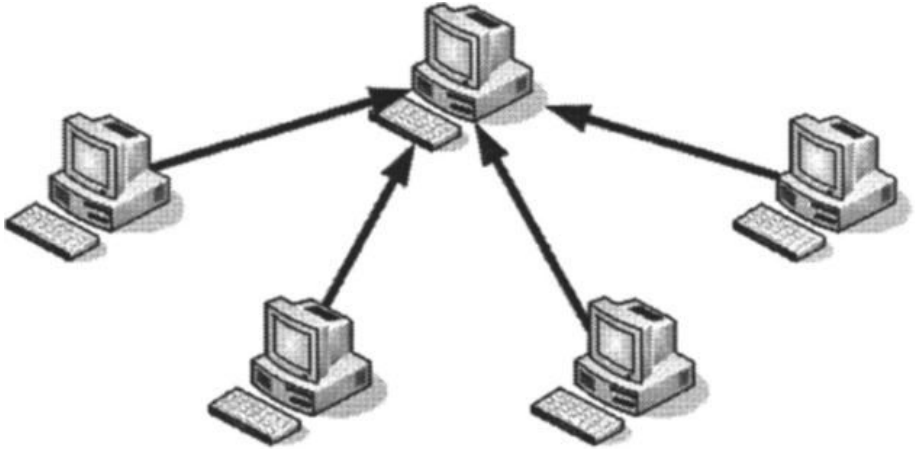
| Transactions | |
|---|---|
| No. Transactions | 76 |
| Total Received | 111,114.60035819 BTC |
| Final Balance | 111,114.60035819 BTC |

Request Payment | Donation Button

# The Bitcoin Network

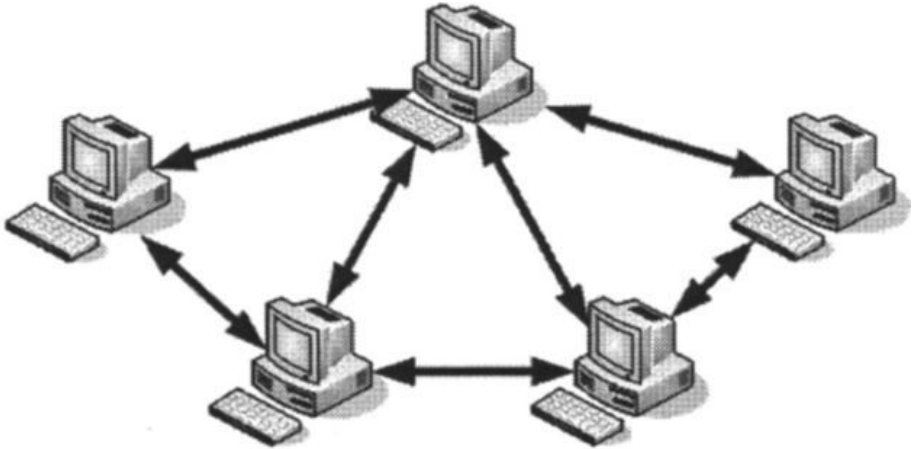
- P2P: Computers that participate in the network are “peers” to each other
  - They are all equal, and there are no special nodes
  - Decentralization
  - Example: Limewire
- Data is propagated to the nearest peer, etc
- Bitcoin network is a group of “peers” running the Bitcoin P2P Protocol
  - All nodes are equal, they may take on different roles
  - All peers have a copy of the blockchain

Centralized



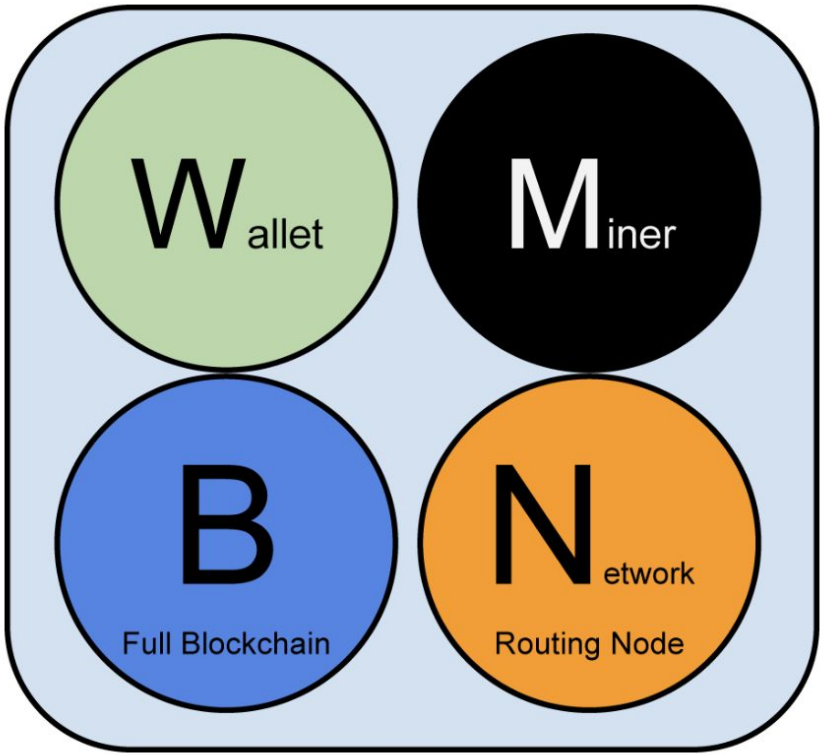
Client-server Model

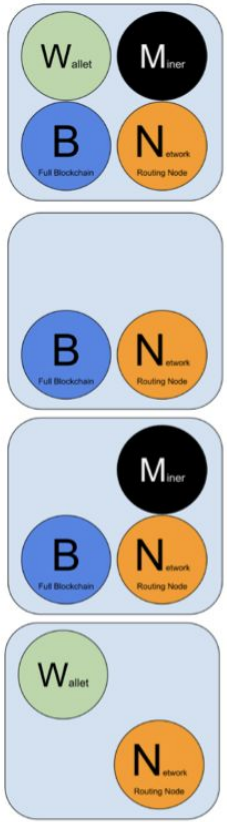
Decentralized Networks



Peer-to-peer Model

# Types of Nodes in Bitcoin P2P Network




**Reference Client (Bitcoin Core)**

Contains a Wallet, Miner, full Blockchain database, and Network routing node on the bitcoin P2P network.

**Full Block Chain Node**

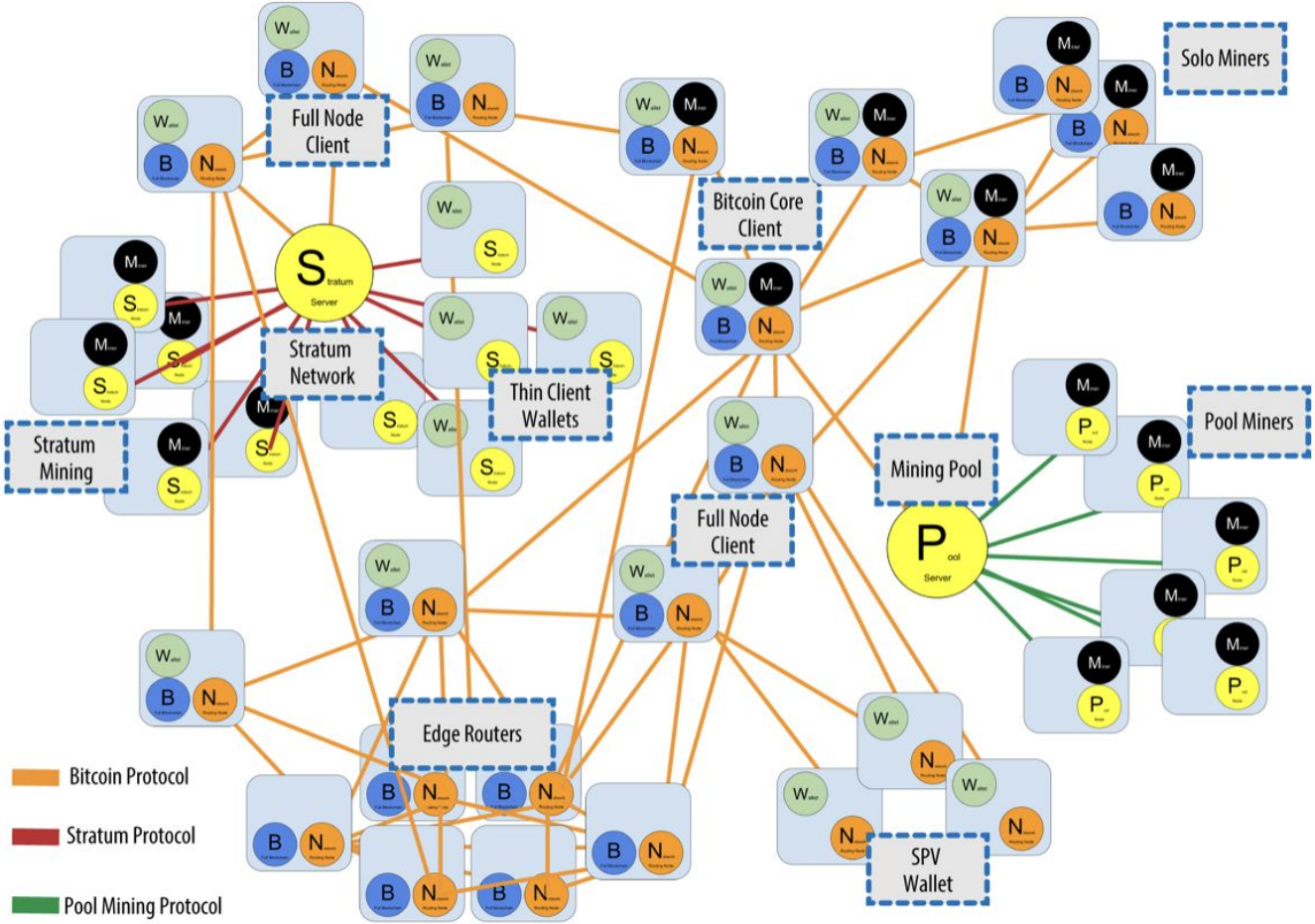
Contains a full Blockchain database, and Network routing node on the bitcoin P2P network.

**Solo Miner**

Contains a mining function with a full copy of the blockchain and a bitcoin P2P network routing node.

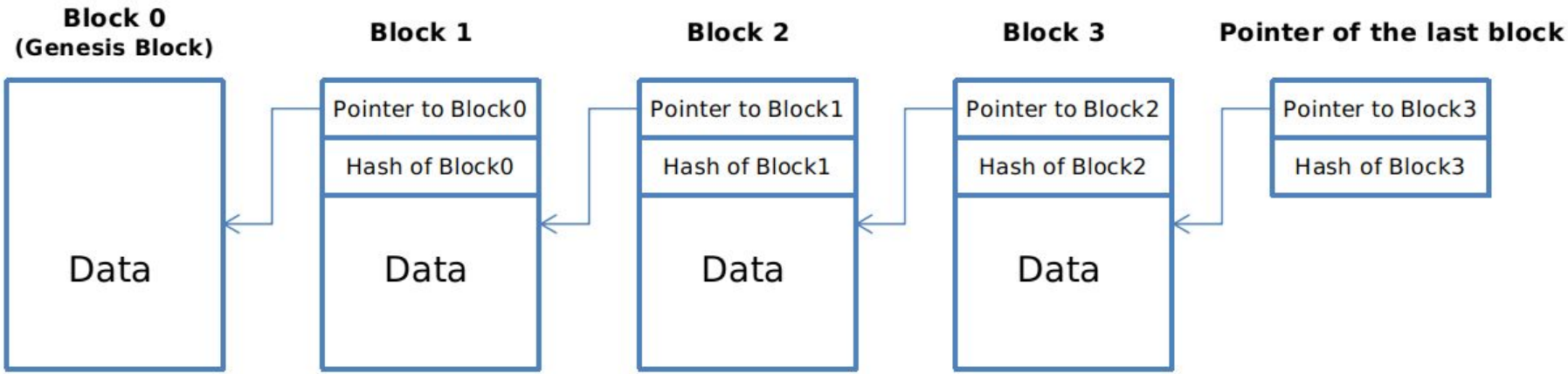
**Lightweight (SPV) wallet**

Contains a Wallet and a Network node on the bitcoin P2P protocol, without a blockchain.


Full Node Client
Solo Miners
Bitcoin Core Client
Stratum Server
Stratum Network
Thin Client Wallets
Stratum Mining
Full Node Client
Mining Pool
Pool Server
Pool Miners
Edge Routers
SPV Wallet
Bitcoin Protocol
Stratum Protocol
Pool Mining Protocol

# Public Transaction Ledger (The Blockchain)

- Blockchain Data Structure: Back Linked List of Blocks of Transactions
- Each block within blockchain is identified by hash of its header
- Each block also references its parent block, through previous hash block



# Structure of a Block

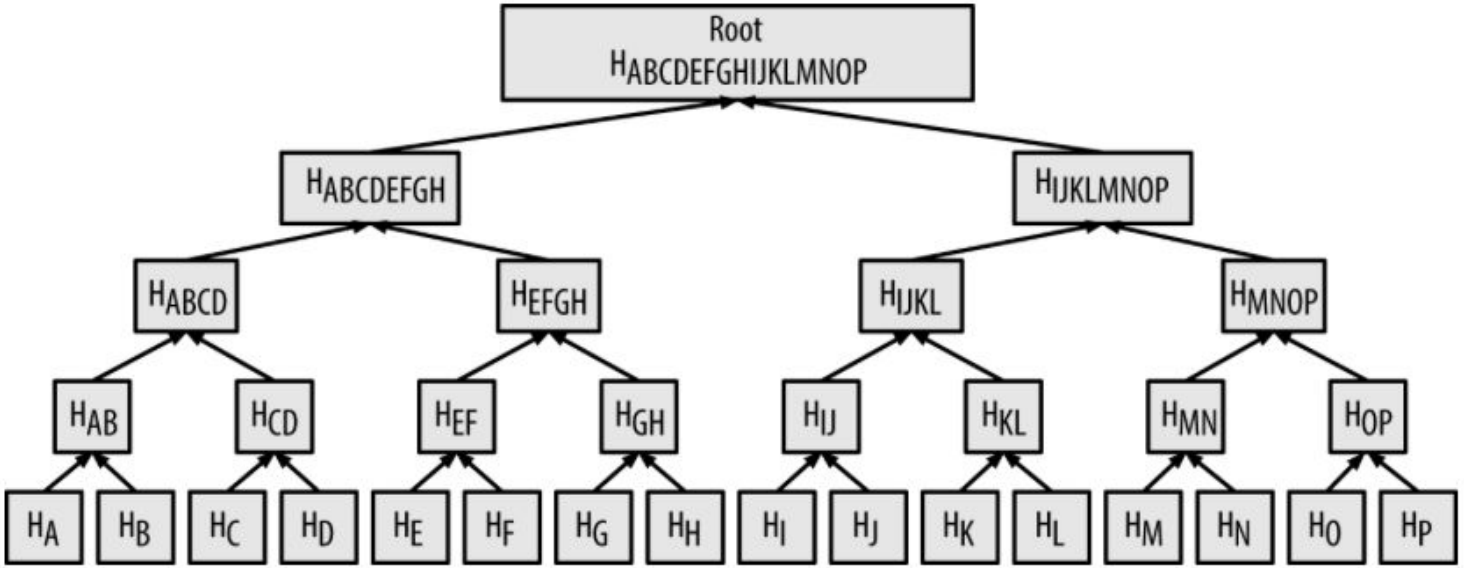
| Size | Field | Description |
|---|---|---|
| 4 bytes | Block Size | The size of the block, in bytes, following this field |
| 80 bytes | Block Header | Several fields form the block header |
| 1–9 bytes (VarInt) | Transaction Counter | How many transactions follow |
| Variable | Transactions | The transactions recorded in this block |

# Block Header

| 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 |

# Merkle Trees

- Each header contains a merkle root of all block's transactions



# Decentralized Consensus

- How can we get everyone in the network to agree on a single universal “truth” about who owns what, without having to trust anyone?
- Emergent Consensus
  - No fixed moment when consensus occurs, instead it emerges from thousands of independent nodes following the same set of rules

# Independent Verification of Transactions (Network Nodes)

- Each node verifies every transaction against a long list of criteria (protocol)
  - Syntax and Data Structure is Correct
  - Reject if sum of input values is less than sum of output values
  - For each input, the referenced output must exist and cannot already be spent
    - Double Spending Problem
  - Etc
- If a transaction is valid, every node builds a pool of valid (but unconfirmed) transactions

# Mining Nodes (Proof-Of-Work Algorithm)

- Mining validates new transactions and records them on the blockchain
- Incentive: New Bitcoin
- Mining is the mechanism by which the bitcoin network is decentralized
- A New Block is “mined” roughly every 10 minutes

# Building a Block

- Mining node will aggregate these valid transaction from the transaction pool, and create a *candidate block*
- Candidate is not valid yet, and only becomes valid if the miner succeeds in finding a solution to the proof of work algorithm
- Coinbase transaction - Transaction input is different than a traditional transaction
  - This is the reward for the miner
    - BTC reward
    - Fees from each transaction
  - Payment (or output) of this transaction goes to the miner’s wallet
  - First transaction in a block

# Building the Block

1. Aggregate transactions from transaction pool
2. Add Previous Block Hash
3. Summarize all transactions in the block in the merkle root
4. Add timestamp
5. Fill in the target (Difficulty of PoW)
6. Nonce, initialized to zero

# Proof of Work Algorithm

- Mining- the process of hashing (SHA256) the block header repeatedly, while changing one parameter (nonce), until the resulting hash is less than a specific target
- Target - determines the difficulty of getting a hash correct
  - The lower the target value is, the more difficult it is to find a correct hash
- The goal of mining is to find a hexadecimal value (hashed header) that is less than the target
- By incrementing the nonce, and rehashing a new value is found

# Example:

Find a hash that is lower than : “I am Satoshi Nakamoto”

Hash:
5d7c7ba21cbbcd75d14800b100252d5b428e5b1213d27c385bc141ca6b47989e

# Example:

```
$ python hash_example.py
I am Satoshi Nakamoto0 =>
a80a81401765c8eddee25df36728d732...
I am Satoshi Nakamoto1 =>
f7bc9a6304a4647bb41241a677b5345f...
I am Satoshi Nakamoto2 =>
ea758a8134b115298a1583ffb80ae629...
I am Satoshi Nakamoto3 =>
bfa9779618ff072c903d773de30c99bd...
I am Satoshi Nakamoto4 =>
bce8564de9a83c18c31944a66bde992f...
I am Satoshi Nakamoto5 =>
eb362c3cf3479be0a97a20163589038e...
I am Satoshi Nakamoto6 =>
4a2fd48e3be420d0d28e202360cfbaba...
I am Satoshi Nakamoto7 =>
790b5a1349a5f2b909bf74d0d166b17a...
I am Satoshi Nakamoto8 =>
702c45e5b15aa54b625d68dd947f1597...
I am Satoshi Nakamoto9 =>
7007cf7dd40f5e933cd89fff5b791ff0...
I am Satoshi Nakamoto10 =>
c2f38c81992f4614206a21537bd634a...
I am Satoshi Nakamoto11 =>
7045da6ed8a914690f087690e1e8d66...
I am Satoshi Nakamoto12 =>
60f01db30c1a0d4cbce2b4b22e88b9b...
I am Satoshi Nakamoto13 =>
0ebc56d59a34f5082aaef3d66b37a66...
I am Satoshi Nakamoto14 =>
27ead1ca85da66981fd9da01a8c6816...
I am Satoshi Nakamoto15 =>
```

- “I am Satoshi Nakamoto13” generates a hash that is less than the target

# Proof of Work

- Work cannot be cheated:
  - Brute force is the only way to get a valid hash
- Easily Verifiable:
  - Nonce is a part of the block header
  - Any node in the network can hash using that nonce and verify that the value is less than the target
- Increasing the difficulty by 1 bit, you decrease the search space by half
- Effectively, doubles the amount of time it takes to find a solution

# Proof of Work

- As more mining nodes compete, the more difficult it gets to mine BTC
- Current Rates:
  - 206 EH/s
  - 206 Exahashes (1 quintillion) per second is being computed in the bitcoin network

# Adjusting the Target

- Blocks should be generated every 10 minutes
- Target is a dynamic parameter to maintain those 10 minutes per block
  - Every 2,106 blocks
- How do you do that in a decentralized network?
  - Time it took to find the last 2,016 blocks and compare that to the expected 20,160 minutes
  - The ratio between actual and adjusted timespan if proportionately adjusted

# BTC Hashrate: 203.35 EH/s

Mar 30, 2022 01:37 AM UTC – 203,346,177,306,329,800,000 H/s

# Current Statistics on Mining

- 6.25 BTC per Block
- Miners were paid $42.3 Million in the last 24 hours to secure the bitcoin network
- Amount of Bitcoin Reward is halved every 210,000 blocks
  - Roughly every 4 years
- Expected to Run out of Bitcoin by 2140
  - Lowest Denomination: 0.00000001 BTC
  - 1 satoshi, which is the smallest unit in Bitcoin network


Bitcoin Money Supply
20,000,000
18,000,000
16,000,000
14,000,000
12,000,000
10,000,000
8,000,000
6,000,000
4,000,000
2,000,000
Bitcoin Created
1/1/2010
1/1/2015
1/1/2020
1/1/2025
1/1/2030
Time

# Successful Mining of a Block

1. Mining node transmits the block to all of its peers
2. The peers validate the block, and then propagate the block
3. Each node adds the block to its local blockchain

# Thoughts on the future of digital currencies

- Money: Medium of Exchange
- Digital Currencies:
  - Lower latency
  - Lower fees
- Moving to a Digital Economy
  - Apple Pay, Google Pay, Venmo