Cryptocurrency

A brief look at Bitcoin

Introduction to Computer Security

Naercio Magaia and Imran Khan

Outline

- Traditional Financial Arrangements
- Hash Pointers and Data Structures
- Distributed Consensus
- Incentives and Proof of Work
- Bitcoin Mechanics
- Bitcoin Transactions
- Bitcoin Blocks
- The Bitcoin Network
- Limitations

Traditional Financial Arrangements

- If there were no governments or currency, would it be possible to acquire services or goods?
- The barter system
  - If Alice has food that she's willing to trade for a tool
  - If Bob, who has a tool, doesn't have any need for food, but wants medicine instead
  - Alice and Bob can't trade with each other

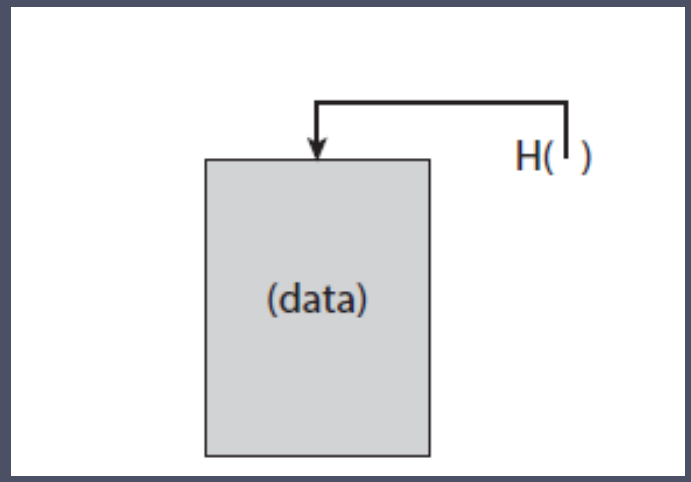
- The barter system's drawback is coordination.
  - Arranging a group of people, whose needs and wants align, in the same place at the same time
- Credit and cash emerged to solve coordination.
  - Which of these does Bitcoin fall under?

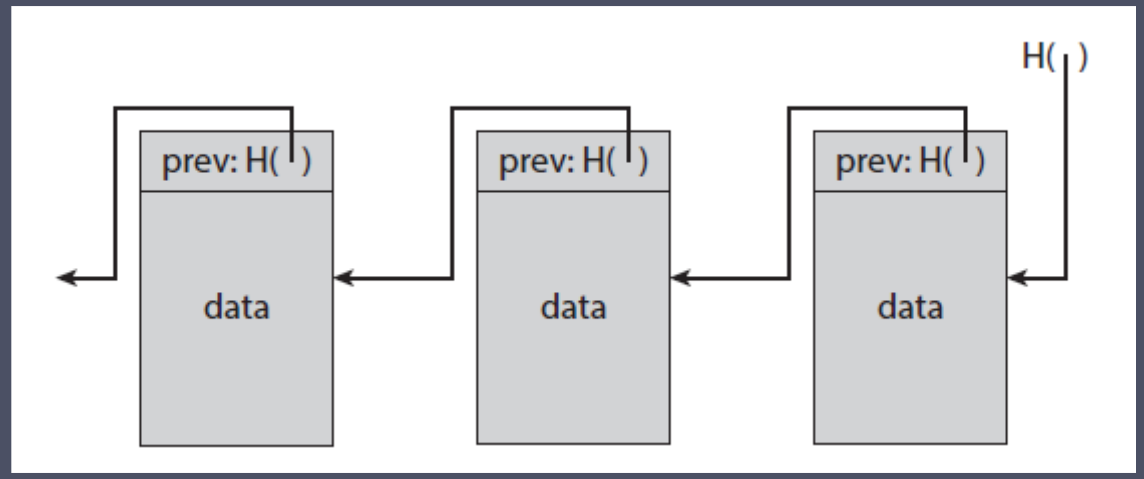
Cash

- It allows us to be precise about how much something is worth.
  - Enables using numbers to talk about value.
    - Even when using credit, debt is measure in the amount of cash it would take to settle it.
- A cash-based system needs to be bootstrapped with some initial allocation of cash, without which no trades can occur
- It offers two additional advantages
  - Better anonymity
  - Enable offline transactions

Hash Pointers and Data Structures

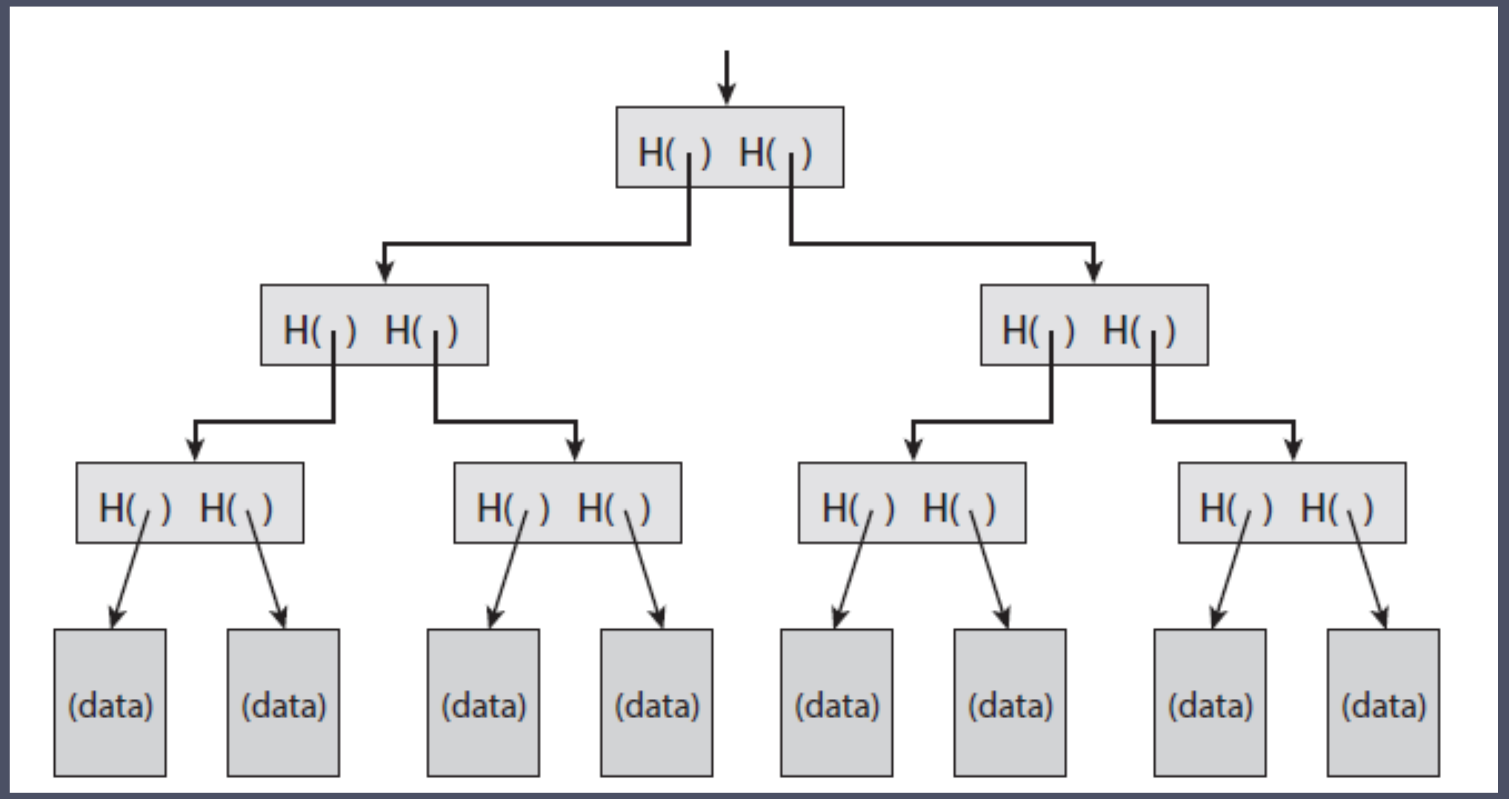
- Hash Pointer is simply a pointer to where some information is stored together with a cryptographic hash of the information.
  - It allows you to verify that the information hasn't been changed
- A block chain is a linked list that is built with hash pointers.
  - Each block only tells where the value of the previous block was
  - Contains a digest of that value, which allows to verify that the value hasn't been changed





Hash Pointers and Data Structures

- A Merkle tree is a binary tree with hash pointers.
  - Allow a concise proof of membership



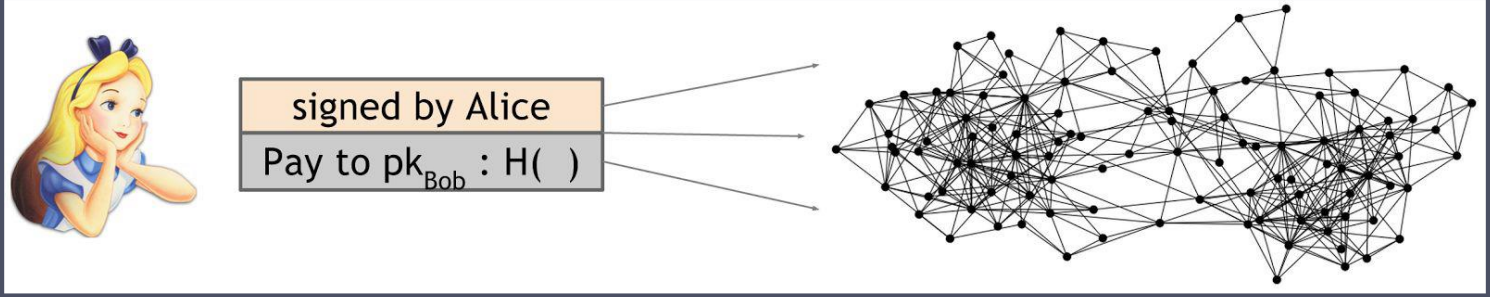
Distributed Consensus

- It has various applications, e.g., reliability in distributed systems.

Distributed consensus protocol. There are n nodes that each have an input value. Some of these nodes are faulty or malicious. A distributed consensus protocol has the following two properties:

  - It must terminate with all honest nodes in agreement on the value
  - The value must have been generated by an honest node

- In the context of Bitcoin
  - Bitcoin is a peer-to-peer system
  - When Alice wants to pay Bob, she broadcast a transaction to all the Bitcoin nodes that comprise the peer-to-peer network



Bitcoin "Simplified" Consensus Algorithm

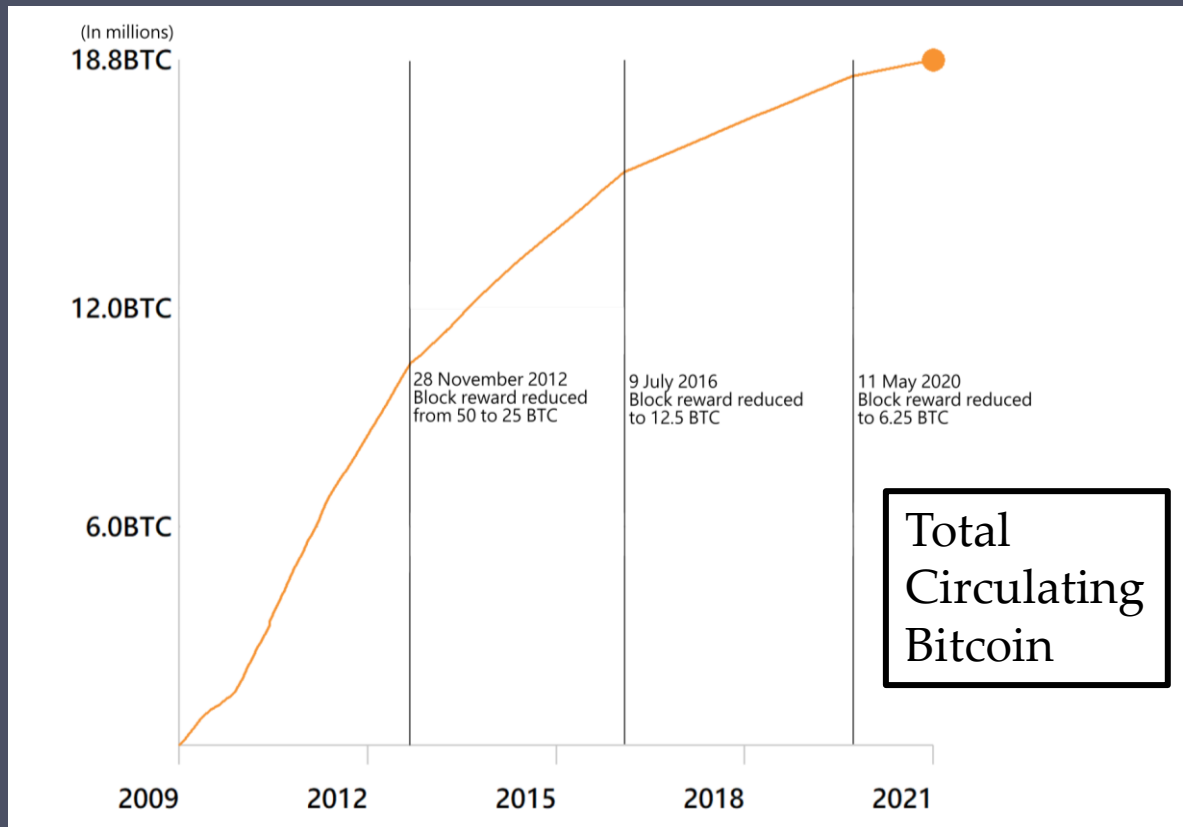
- The Algorithm
  1. New transactions are broadcast to all nodes
  2. Each node collects new transactions into a block
  3. In each round, a random node gets to broadcast its block
  4. Other nodes accept the block only if all transactions in it are valid (i.e., unspent, valid signatures)
  5. Nodes express their acceptance of the block by including its hash in the next block they create
- Does it work? Possible attacks include
  - Stealing Bitcoins
  - Denial of service attack
  - Double-spend attack

Incentives and Proof of Work

- Bitcoin's decentralization is partly a *technical mechanism* and partly clever *incentive engineering*.
  - Can we penalize, somehow, the node that created the block with the double-spend transaction?
  - Can we reward each of the nodes that created the blocks that did end up on the long-term consensus chain?
- There are two separate incentive mechanisms in Bitcoin
  - Block Reward
  - Transaction fees

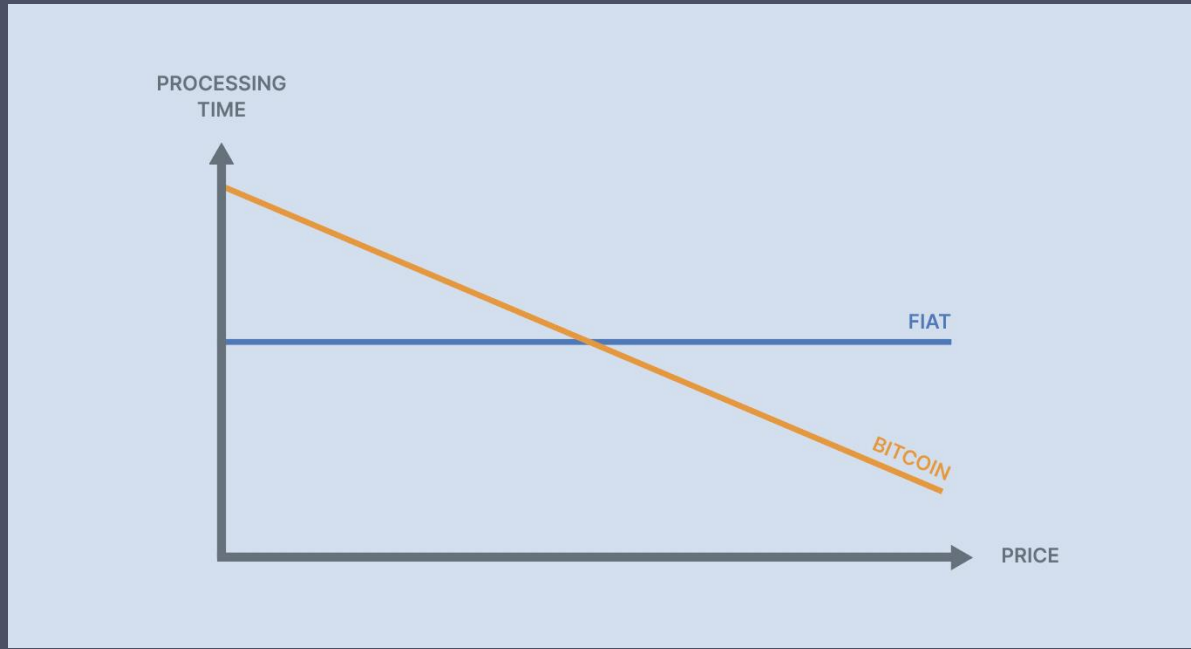
Block Reward

- According to the rules of Bitcoin
  - the node that creates a block gets to include a special transaction in that block.
  - this transaction is a *coin-creation transaction*, and
  - the node can also choose the recipient address of this transaction
    - typically choosing an address belonging to itself
- As of December 2020, the block reward for that successful hash was 6.25 bitcoins.
  - Each new block generates 6ish bitcoins per block (around £86k)
  - This is reducing over time, and by 2040, only reward will be fees



Transaction fees

- The transaction creator decides to make the total value of the transaction outputs less than the total value of its inputs.
  - Whoever creates the block that first puts that transaction into the block chain gets to collect the difference, which acts a transaction fee
- Bitcoin processes transactions by
  - propagating them around the network via nodes,
  - getting them included into a block by a miner, and
  - sharing, verifying, and storing the resultant blocks on every full node
- The transaction fee is purely voluntary
- Bitcoin allows for price discrimination and priority processing



Mining and Proof-of-Work

- The key idea behind proof-of-work is to approximate the selection of a random node by instead selecting nodes in proportion to a resource that hopefully nobody can monopolize.
  - If, for example, that resource is computing power, then it's a *proof-of-work* system.
  - if it is proportion to ownership of the currency, then that's called *proof-of-stake*.
- Bitcoin achieves proof-of-work using *hash puzzles*.
  - In order to create a block,
  - the node that proposes that block is required to find a number, or nonce, such that
  - when concatenating the **nonce**, the **previous hash**, and the **list of transactions** that comprise that block and take the hash of this whole string, then
  - hash output should be a number that falls into a **target space** that is quite small in relation to the much larger output space of that hash function.
  - It can define such a target space as any value falling below a certain target value.

$$H(nonce \,||\, prev\_hash \,||\, tx \,||\, tx \,||\, \ldots \,||\, tx) < target$$

Mining and Proof-of-Work

- Hash puzzles important properties
  1. They need to be quite difficult to compute
     - For example, the difficulty level is about 10^{20} hashes per block at the end of 2014
     - This process of repeatedly trying and solving these hash puzzles is known as **Bitcoin mining**, and the participating nodes are called **miners**.
  2. The cost need to be parameterizable, not a fixed cost for all time
     - All the nodes in the Bitcoin peer-to-peer network **will automatically recalculate the target**, that is the size of the target space as a fraction of the output space, every 2016 blocks
     - A lot of attacks on Bitcoin **are infeasible** if the majority of miners (at least a 51 percent), weighted by hash power, are following the protocol or are honest.
  3. It is trivial to verify that a node has computed proof of work correctly
     - It is thus trivial for any other node to look at the block contents, hash them all together, and verify that the output is less than the target

Mining and Proof-of-Work

- Cost of mining

```
If
    mining reward > mining cost
    then miner profits
where
    mining reward = block reward + tx fees
    mining cost = hardware cost + operating costs (electricity, cooling, etc.)
```



Bitcoin Mechanics

- The Bitcoin consensus mechanism gives an append-only ledger, a data structure that can only be written to
  - Once data is written to it, it's there forever
- There's a decentralized protocol for establishing consensus about the value of that ledger,
- There are miners who perform the protocol and validate transactions
- All these mechanisms ensure that
  - transactions are well formed,
  - they aren't already spent, and
  - the ledger and network can function as a currency
- The currency exists to motivate the miners

Bitcoin Transactions

Transactions are Bitcoin's fundamental building block.

Bitcoin uses a ledger that just keeps track of transactions

The entirety of a transaction output must be consumed by another transaction, or none of it

Bitcoin Transactions

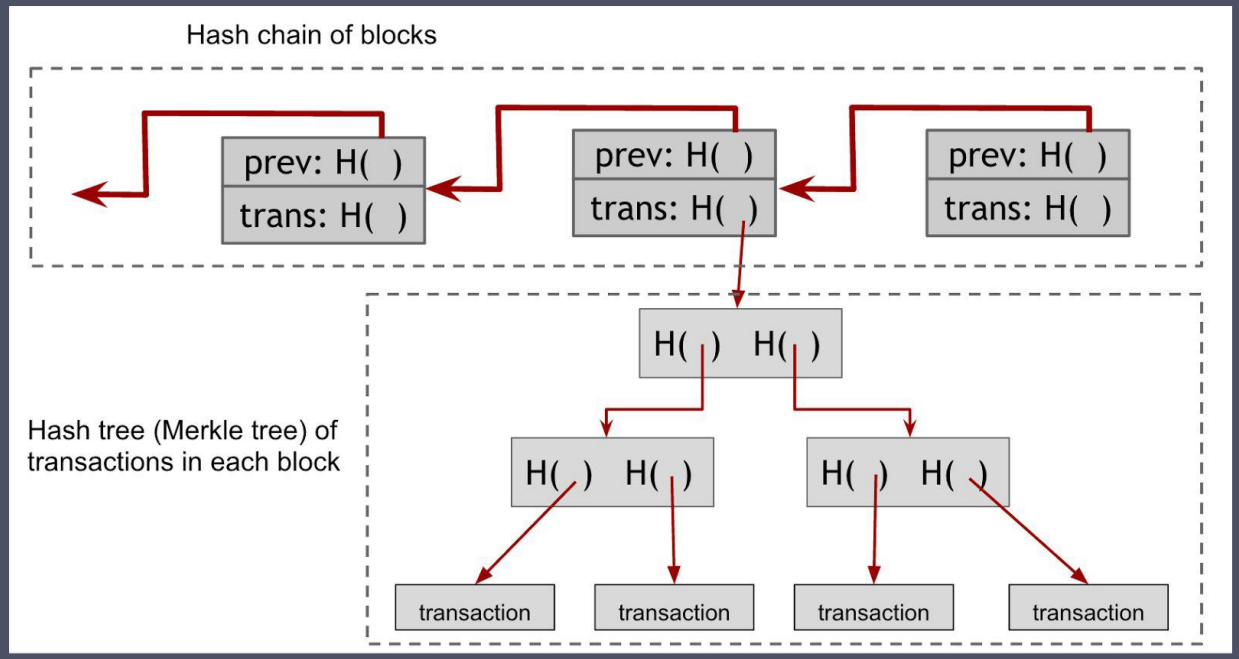
- The three parts to a transaction
  - *Metadata*. There's some housekeeping information
    - the size of the transaction,
    - the number of inputs, and
    - the number of outputs
  - *Inputs*. The transaction inputs form an array, and each input has the same form. It specifies a previous transaction
    - a hash of that transaction
    - the index of the previous transaction's outputs that's being claimed
    - a signature

Bitcoin Transactions

- The three parts to a transaction
  - Outputs. The outputs are again an array. It contains
    - a value, and
    - the sum of all the output values has to be less than or equal to the sum of all the input values

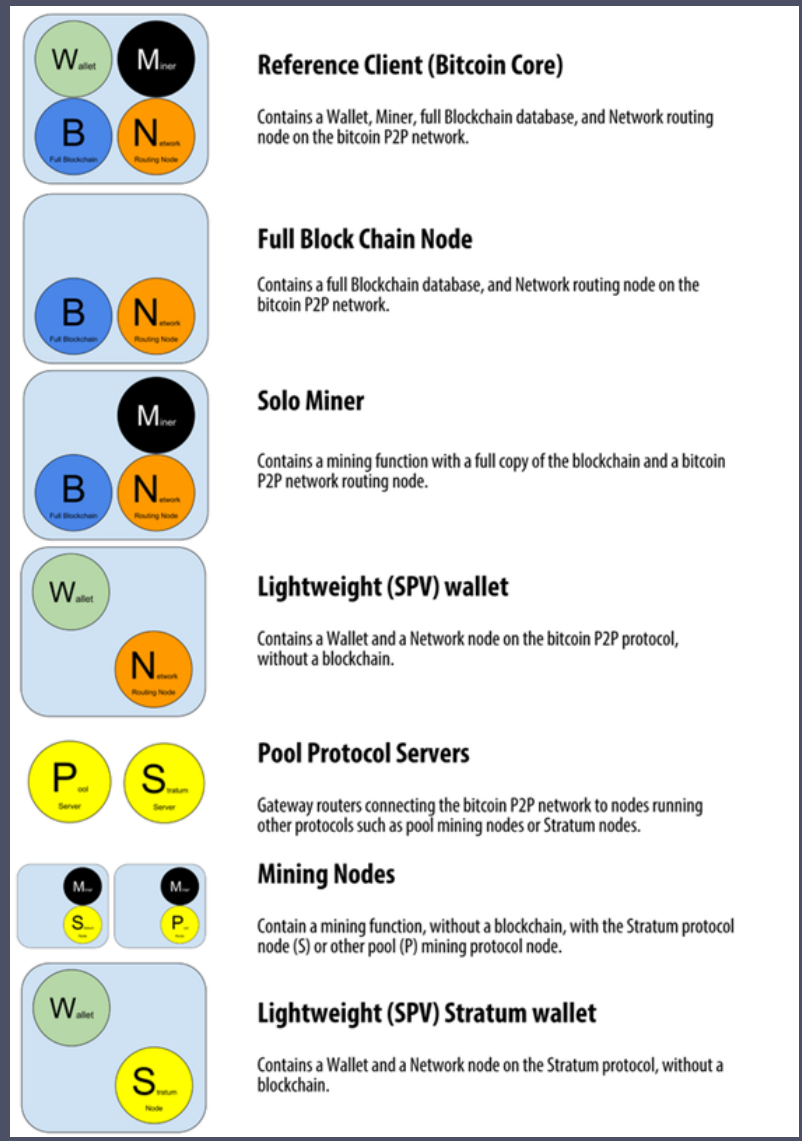
Bitcoin Blocks

- Transactions are grouped together into blocks
- The block chain is a clever combination of two different hash-based data structures
  - The hash chain of blocks. Each block has a block header, a hash pointer to some transaction data, and a hash pointer to the previous block in the sequence
  - A per-block tree of all of the transactions that are included in that block. This is a Merkle tree and allows to have a digest of all the transactions in the block in an efficient way.

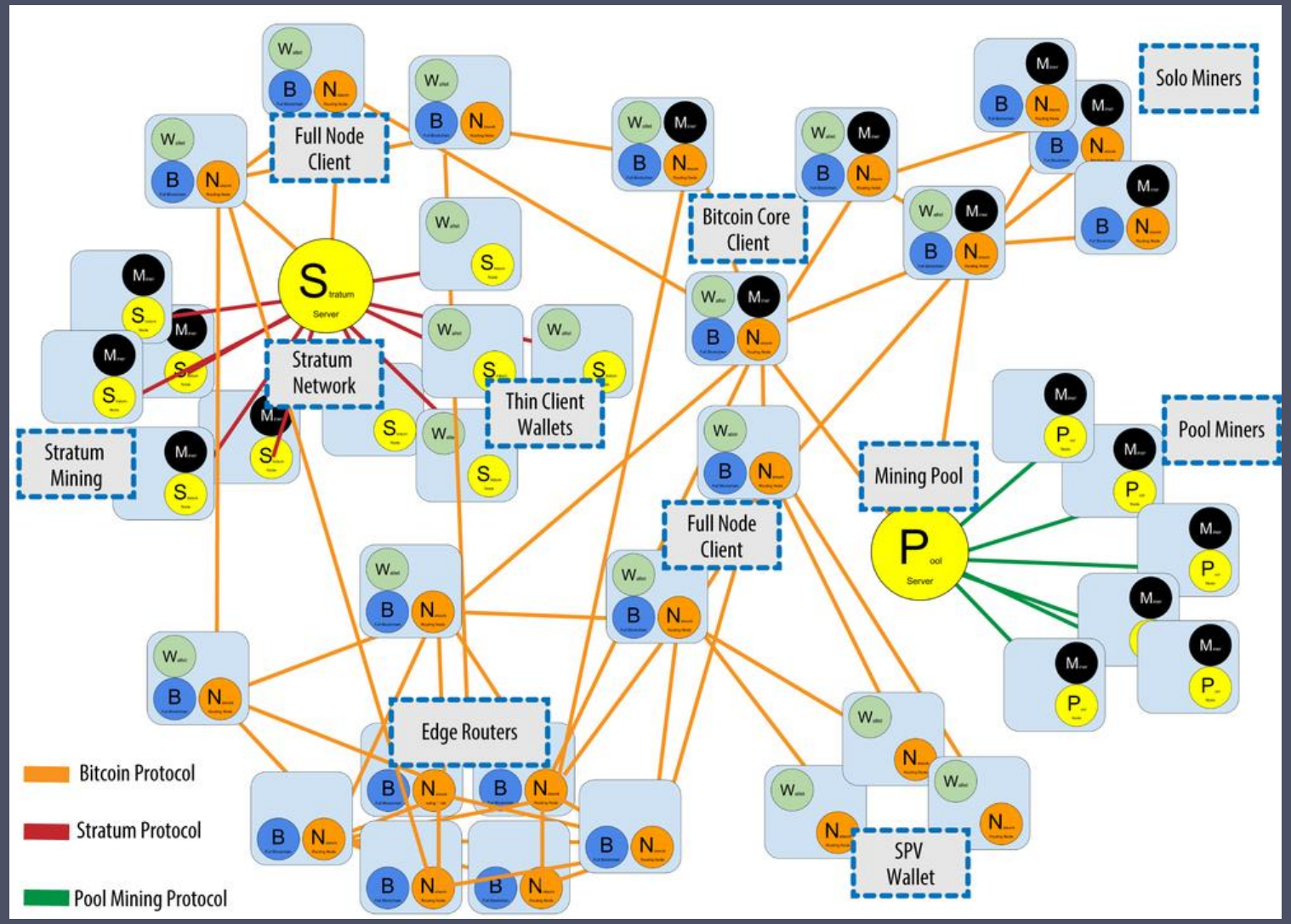


The Bitcoin Network

- It is a peer-to-peer network
- All nodes are equal
  - There is no hierarchy, and there are no special nodes or master nodes
- It runs over TCP and has a random topology
- The network changes over time and is quite dynamic due to nodes entering and leaving.
- Node functions are:
  - *Wallet* holds the private keys of user identities
  - *Full Blockchain Nodes* hold copies of the current blockchain, and can verify transactions
  - *Mining Nodes* compete to form Blocks from transactions, and get paid the transaction fees and a number of bitcoins per block
  - *Routing Nodes* passe transactions across the whole network



The Bitcoin Network



Limitations

- There are some built-in limitations to the Bitcoin protocol, and many constraints hard-coded into the protocol. For example,
  - the size of blocks
    - Each block is limited to a megabyte,
    - Each transaction is at least 250 bytes.
  - the limits on the average time per block,
    - Each block has a limit of 4,000 transactions
    - Blocks are found about every 10 minutes, and
    - Therefore, the Bitcoin network can handle about 7 transactions per second
  - the choices of cryptographic algorithms in Bitcoin are fixed
    - Only one signature algorithm, ECDSA
  - the total number of Bitcoins, and the block reward structure
    - the economic implications of changing them **are too great**

Summary

- Traditional Financial Arrangements
- Hash Pointers and Data Structures
- Distributed Consensus
- Incentives and Proof of Work
- Bitcoin Mechanics
- Bitcoin Transactions
- Bitcoin Blocks
- The Bitcoin Network
- Limitations