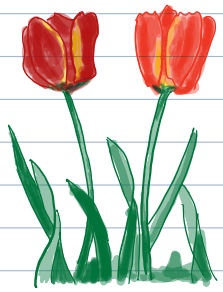
Secure Distributed Systems
CompSci 661 / 461



This video

- Nakamoto's blockchain consensus algorithm
  - consensus
  - coins
  - addresses
  - transactions
  - mining
  - vocabulary
  - Pros and Cons

Before we start — let's make sure we all understand a few terms.

Cryptographic Hash Functions

- One-way functions
- Input can be of any size
- Output is a (small) fixed length, e.g 256 bits or 2^{256} possible values!

- Let x be the input to our hash function

- If $h(\cdot)$ is our hash function, then $z = h(x)$

  z is the resulting "message digest" (or "hash")
  x is called the "pre-image" of z

IF we flipped a single bit of x, then z changes unpredictably

Say you have a set of inputs $x_1, x_2, \ldots, x_n$.

and let's say that all inputs are distinct (unique) but aren't very different.

$z_1 = h(x_1)$; $z_2 = h(x_2)$; ... ; $z_n = h(x_n)$

Each z is an integer chosen uniformally at random from

0 to $2^{256} - 1$

The same is true if the x values (are distinct but) are very different.

Lastly, given a particular z, it's computationally infeasible to find its pre-image

Two specitic cryptographic hash functions are RIPE-160 and SHA-256

Consensus

- We often trade tangible items with the rest of the WORLD

  cash, gold.

- Possession is equivalent to ownership

- Some tangible items require a bit more process

  houses, cars, shares of a company

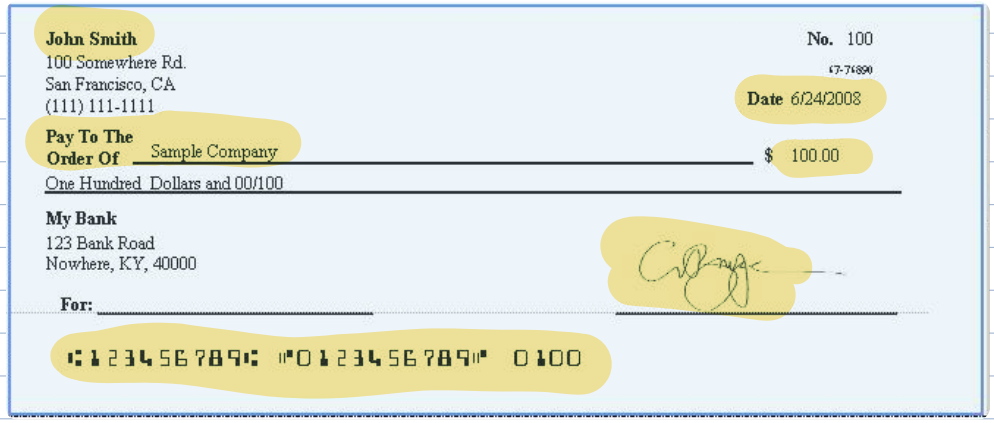
- We have many artifacts and mechanisms to regulate the trading process
  - Registries, titles, certificates
  - Arbiters, markets/exchanges, judges executives, authorities

- All in place to answer questions about Consensus
  - who owns this?
  - How much money is in this account?
  - Is this transaction authorized?

Furthermore, availability of an answer matters.

- Visa is open 24/7

- It's critical that money is owned by one person at a time.
- How do we manage consensus in real life?



Move_Money():

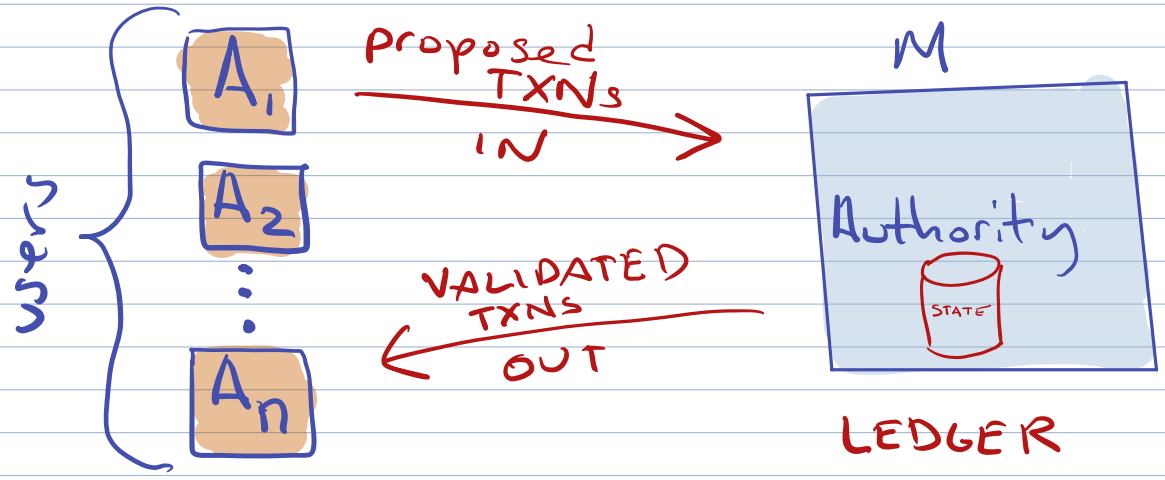
Input: value, source, destination

1. check account exists
2. check authorization
3. check destination exists
4. check sufficient funds
5. remove funds from source
6. insert fund into destination

problems?

- What if funds are removed by a concurrent call to move_money() before 5 but after 4?
- what if 5 completes but 6 fails?

It's easiest to correctly offer consensus services using a single authorized central authority.



But - single point of failure
Single point of attack
Single point of trust

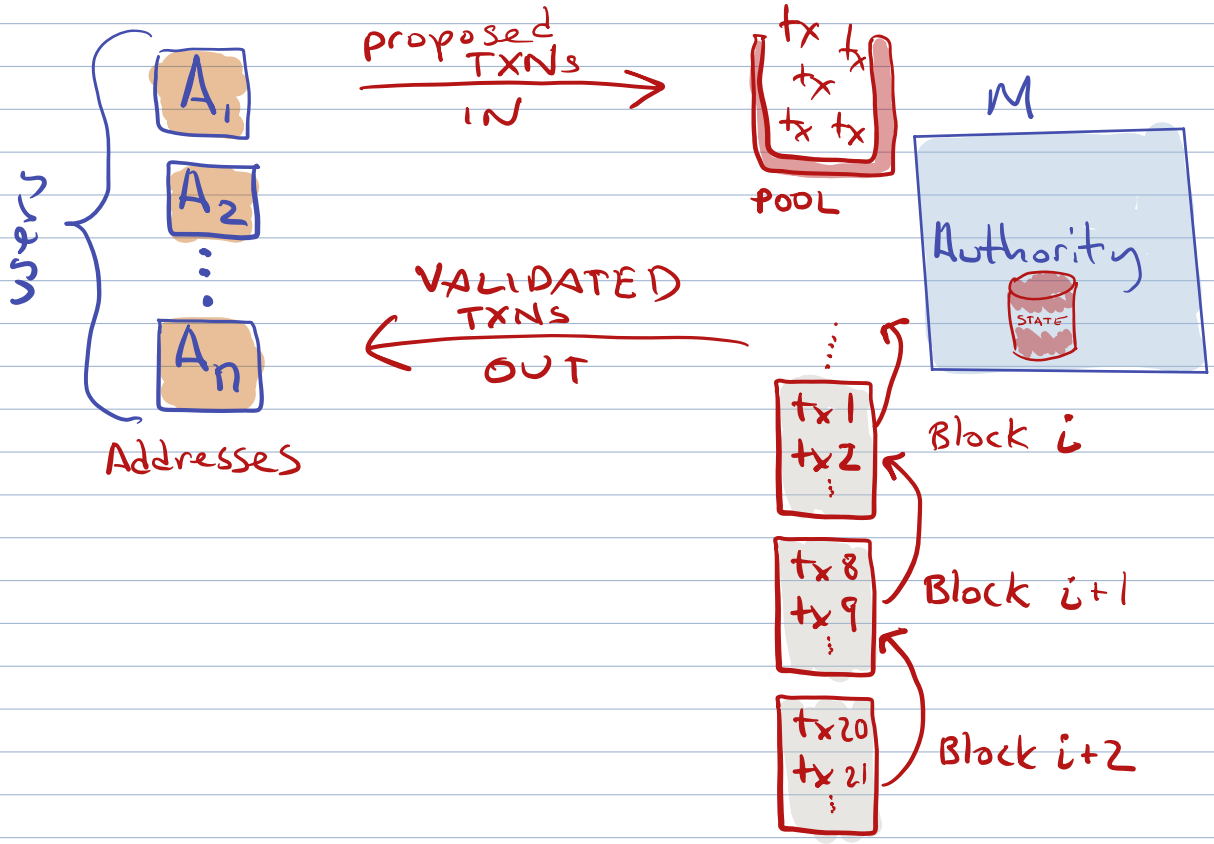
Blockchains

(Satoshi Nakamoto, 2008)

The original paper is about Bitcoin.

- But the idea is more general.
- From now on, we'll use the term "Bitcoin" to refer to the software and protocol

Nakamoto's blockchain is a secure distributed system that manages consensus about the state of a set of addresses (accounts) and authorized transactions among them.



Blockchains have a special data type called **coin**.

- coin is a numeric amount
- divisible into fractions
- transferable
- limited in quantity globally
- cannot be counterfeit
- requires special authority to generate

just like real money

Blockchains also manage other data types, but let's put that aside for now.

- Blockchains manage consensus about how much coin is in accounts called addresses

Addresses

Are created and controlled by users.

- Each is based on an elliptic curve crypto public/private key pair.
- Free to create, but controls no coin at first.
- Blockchains are typically open systems
  - Anyone can create accounts
  - More importantly, anyone can mine
  - Participation requires no authorization

- Addresses are identified by the hash of the public key.
  - Bitcoin key pairs are based on ECDSA and addresses are the RIPEMD-160 hash of the public key.

- A signature by the corresponding private key authorizes the transfer of coin to another address.

Transactions

Similar to checks, above! Here's how they are processed:

1. Input: amount, source address, destination address, blockchain
2. check authorization via source's ECDSA signature
3. check sufficient funds based on history encoded in blockchain
4. transfer control of funds from source address to destination address.
5. Add to a new block in the chain

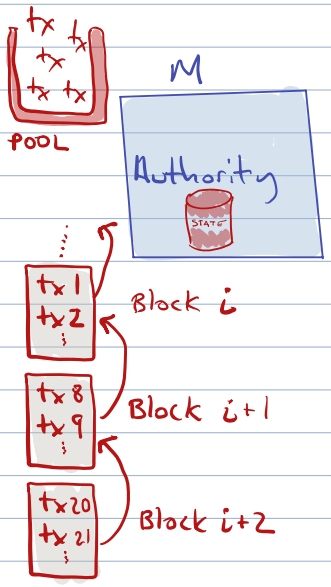
Transactions are processed in batches
- a block of transactions cannot conflict with each other, and must be consistent with all prior blocks.

- Each transaction can elect to pay a fee to the entity that validates the txn (miners).
- A higher fee may garner higher priority.

Validation and Mining

Blockchains are most commonly based on a process called **Proof-of-Work**.

- PoW proposed by Cynthia Dwork & Moni Noar (1993) to thwart Denial of Service attacks
- Adam Back proposed Hashcash as a proof of work based cryptocurrency (1997)



There is no single authority!

← M is actually a collection of entities, each vying to be the one who next adds to the blockchain

$$M = m_1, m_2, \ldots, m_n$$

They are called miners because of the process used to elect which next adds a block:

1) Each miner constructs a candidate block with a set of transactions and the following header.

- version
- h(prior block)
- Difficulty
- Merkle Root of all transactions
- time
- nonce

These values are concatenated and hashed

$B = h(\text{HEADER})$

B is a 256-bit value

(BITCOIN actually hashes twice)

2) Let D = difficulty; and let $t = 2^{256}/D$

The block represents valid Proof of Work if the mining criteria is met

$$h(\text{HEADER}) \leq t$$

IF not, select a new nonce or adjust the time (or the merkle root)

As we'll see, the difficulty is adjusted occasionally so that it takes on average T seconds to find a valid header given the target.

T = 600 seconds in Bitcoin

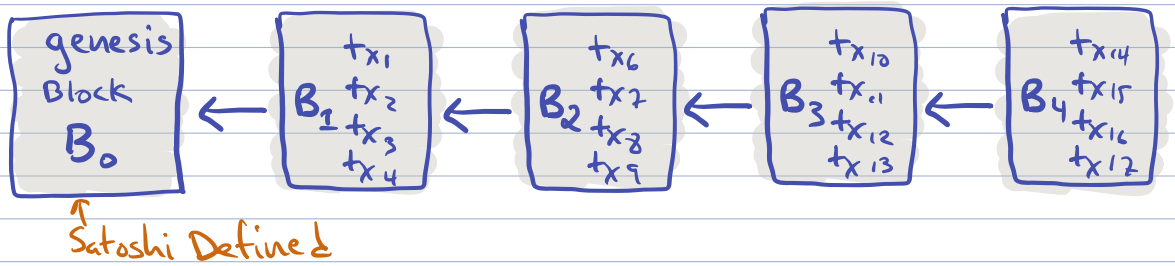
Note that the miner includes a special transaction that creates new coin in her own address - this **coinbase** is a PoW reward.

3) IF the mining criteria is met, the block is announced to all miners and users.

- The transactions that created the Merkle Root are part of the block.
- If other miners confirm that
  1. h(header) ≤ t
  2. each transaction is authorized
  3. each transaction is valid given prior block
  4. All header values are valid*

Then they will add it to the blockchain after its prior.

(*There are smaller details that we'll skip for now.)



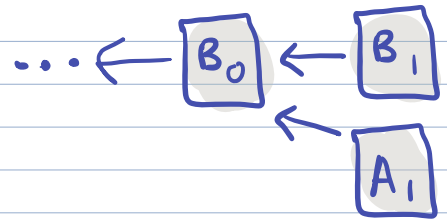
By adding the new block to their copy of the chain,

1. The miners then remove the transactions within from the pool of unconfirmed txs.
2. Start mining a new candidate
   - new set of txns from the pool
   - new prior in header.

Proof of work is probabilistic.

- There is no guarantee that more than one miner performed more than a single hash of one header.
- But for a sequence of n blocks, the probability grows quickly that number of hashes is close to expected

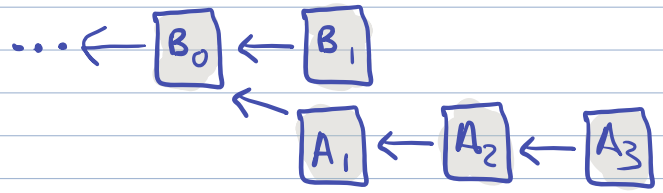
Q: An Internet-based peer-to-peer network connects miners (and users). It takes some time for messages to propogate. What if two miners simultaneously announce a new block?



Each miner in the network will mine on the block they've received first.

Say a set of miners with fraction p of the mining power is mining with B_1 as its prior;
and $(1-p)$ is mining on A_1.

- Eventually, one subset will produce a block first breaking the tie.



The probability of a tie increases as either

- the targeted mean interblock time decreases.

OR

- the network propogation delay increases. (why?)

E.g., Ethereum targets a delay of 15 seconds; Bitcoin 600 seconds.

Blockchain consensus

The state of the system is defined by the set of valid blocks, from the genesis block to the end; at each fork, we take the subtree with the most work completed.

- New blocks can come at any time! Against even the genesis block
- As new blocks are added, the work to write an alternate fork increases.
- Blockchains ensure availability of an answer, but achieve consensus only as blocks are added.

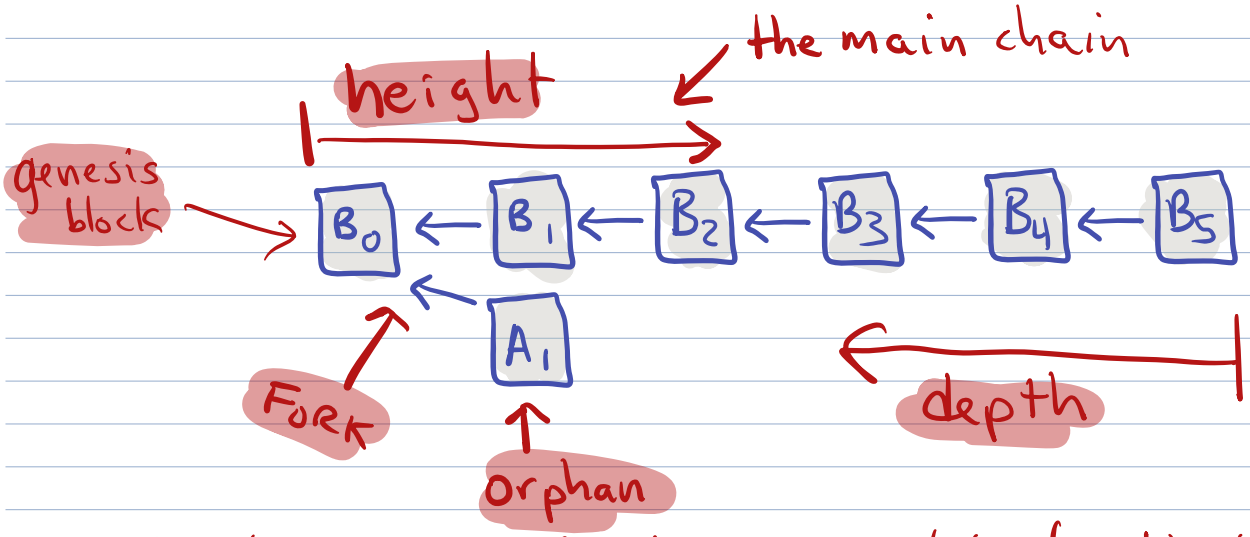
In fact, a variety of scenarios related to forks in the blockchain are critical to understanding its performance and security

Fundamentally, one or more miners with at least ½ the total mining power will win any tie.

And anyone can join as a miner!
(at least on open blockchains)

Even so, it's many, many, many orders of magnitude harder to break the crypto authorizing transactions.

Blockchain Vocabulary



We also might refer to the weight of a block or blocks, which refers to the amount of work performed.

You should memorize/understand all of these terms.

↑ will be on the midterm exam.

Downsides to blockchains and cryptocurrency (and some responses)

1. Proof of work is an inefficient use of energy for providing security for commerce/finance.

   True! "Proof of Stake" is an approach we'll investigate that uses low/no energy to do blockchain.

2. Why do we need a new solution. What's wrong with Paypal, Visa, banks, and cash?

   Nothing is wrong. Blockchains are a fascinating new approach that offers potentially lower fees. It also lowers the barriers to commerce and finance, just as internet lowered barriers to info exchange.

3. Cryptocurrencies are used by only criminals.

   - Not true. In general, Blockchains offer auditable records of financial exchange. Not good for crime. It's a research challenge to provide a truly anonymous cryptocurrency that mimics cash. But it will happen, and it's important to recognize that harms from this technology exist.

4. Cryptocurrencies are not supported by fiscal policies, and their values (exchange rates) are volitile and manipulated by malicious actors.

   - True! We'll look a solutions from finance later in the course.

In general, those against this approach feel it offers solutions to a problem where for most cases it is already solved — many of remaining cases are socially undesirable.