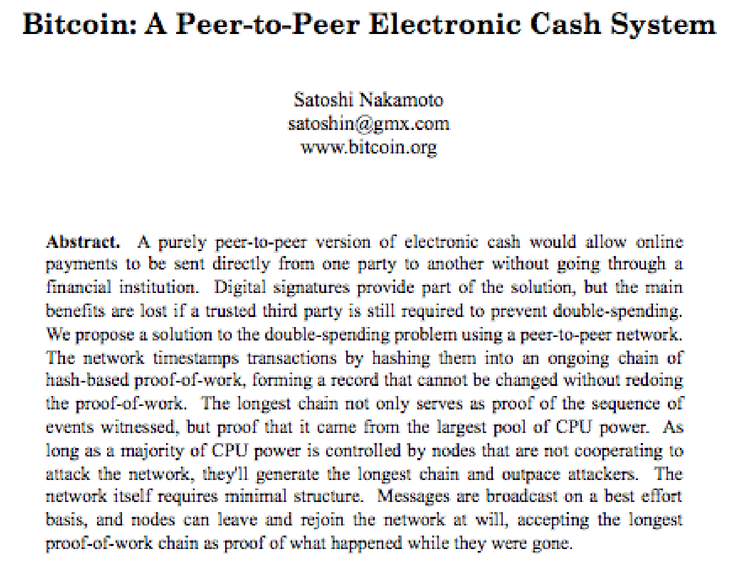
Ethereum

LUG @ NC State, 2/11/22
William Harrell

In the beginning...

- Several computer scientists develop data systems centered around cryptographic proofs in the 80s and 90s, leading to the creation of blockchains
- Some person (or group) going by Satoshi Nakamoto presents a way for users to add blocks to the system instead of a third party, allowing for decentralized blockchains and forming the basis of Bitcoin



Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

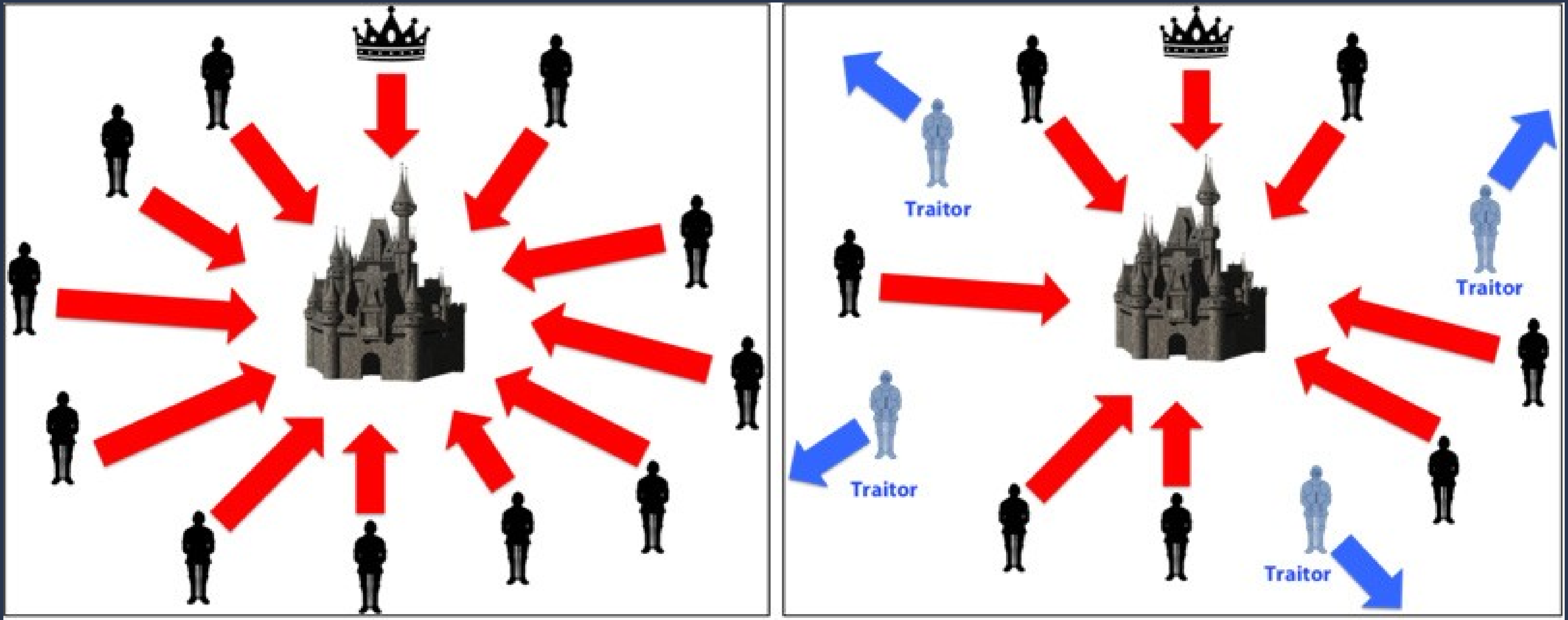
Blockchain basics

- It’s an append only linked list that you can do fancy cryptographic proofs on
- Each new block (node in list) contains the hash of the previous block
  - If you want to alter the history of the chain, you have to change your target block…
  - And the next one…
  - And the next one…
  - And the next one…
  - Which is infeasible for most chains
- You can put whatever you want in these linked lists
- Before Bitcoin, creating new blocks was limited to a central authority

Byzantine consensus

- The main barrier to decentralization was keeping the entire network on the same page
- For a financial system to work, your balance must be the same everywhere
- If nodes can’t agree, the whole thing falls apart
- At the same time, even if all nodes are good actors, there needs to be order in the network
  - Blocks need to be mined slowly, so entire network has time to agree
  - Blocks should also be hard to make, so harder for bad actors
- Each decentralized blockchain needs some consensus algorithm to allow nodes to come to agreement

Byzantine Generals problem



Coordinated Attack Leading to Victory

Uncoordinated Attack Leading to Defeat

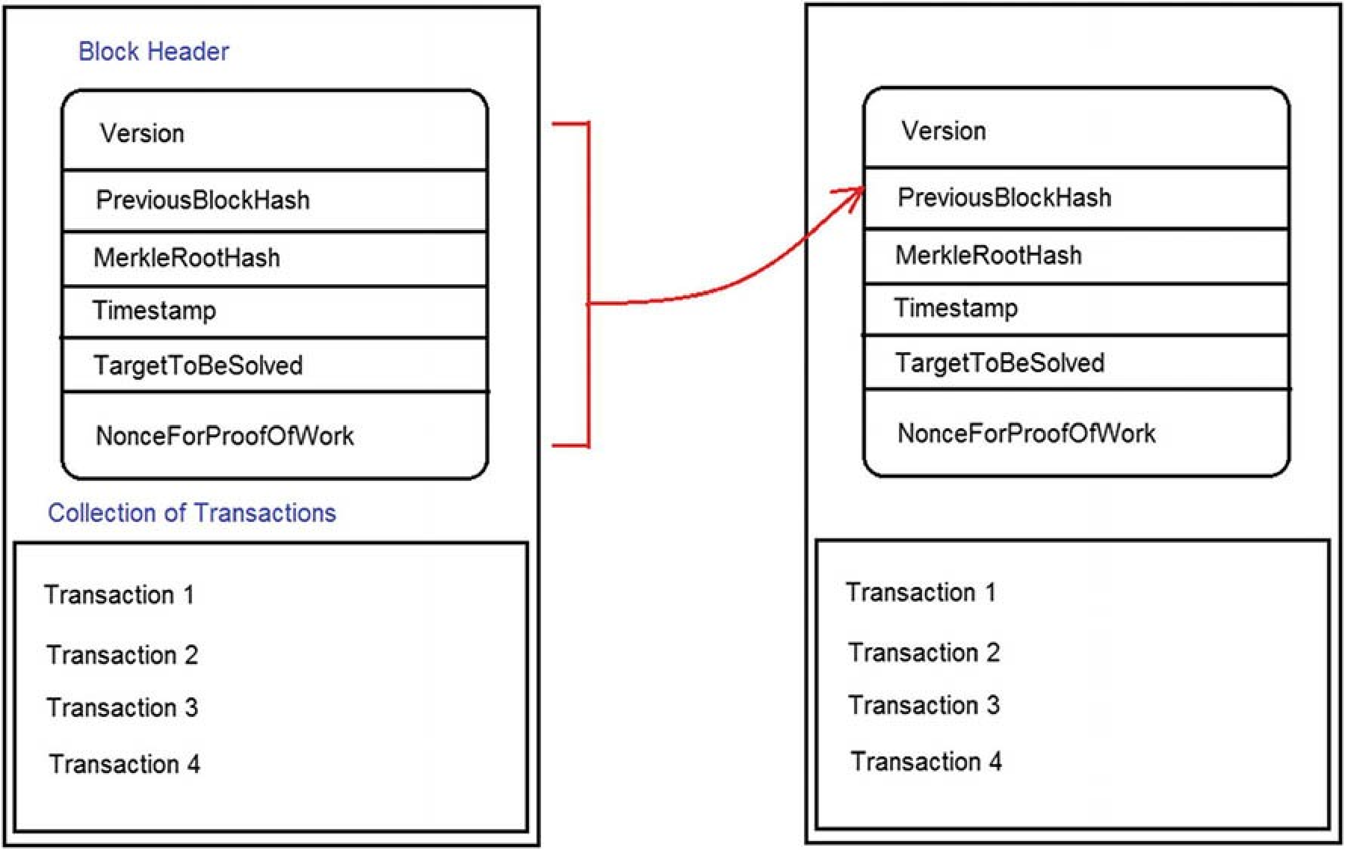
Bitcoin's consensus mechanism: Proof of Work

- When a block is ready to be mined, it will be hashed
- If the hash is under the network's difficulty factor, the block is valid and will be accepted by the rest of the network
  - This difficulty factor scales as more miners join the network, to keep the average mining rate at ten minutes
- Chances are, the block by itself is not valid, so there's a space for a nonce that can be increment to get a new hash
- The mining process is taking the block of new transactions and changing the nonce until the block's hash is less than the difficulty factor
- This process is both time and energy intensive, making it difficult for an attacker to mount an attack against a large network

ASIC mining

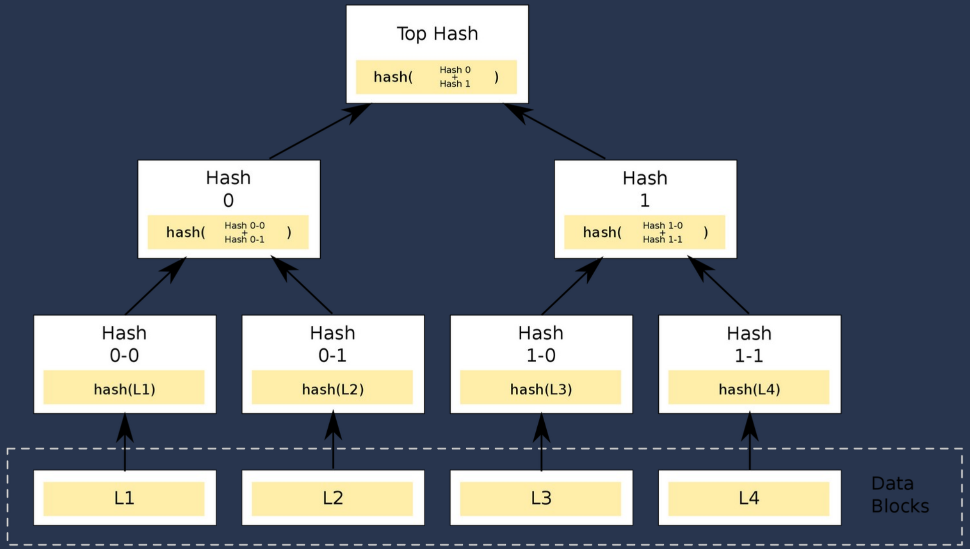
- Bitcoin uses SHA-256
- At first, it was done on CPUs…
- Then people realized GPUs were way faster…
- Then companies started making ASICs…
- Now almost all Bitcoin mining is done on ASICs

Bitcoin block structure



Merkle Trees

- To keep the actual blockchain small, all the transactions in a block are “combined” into a Merkle tree, a type of tree centered around hashing
- Data is added as leaves
- Each parent has the hash of its children
- Root node will be a single hash, which is stored in the block.
- A proof can be done to show that a certain piece of data is within the tree, using other hashes in the tree.



Unspent Transaction Outputs

- Having to review the entire blockchain to determine an addresses’ balance would be a nightmare
  - And that’s exactly what it does! (Though you can also use a DB to help out)
- Everytime you make a transaction, you have to use your UTXOs
  - Think of them like $1, $5, $10 etc bills
- To “send” 2 BTC to Bob, Alice will use a 1.5 BTC and a 1 BTC UTXO
- The 2 will be sent to a new UTXO allocated to Bob, and the remaining 0.5 will be sent to a UTXO allocated to her

Problems with Bitcoin

- Slow transaction time of ten minutes
  - Addressed by Lightning Network, but it has a lot of problems
- But above all else, it’s boring
  - Script, but very basic (no loops, not Turing complete)

The birth of Ethereum

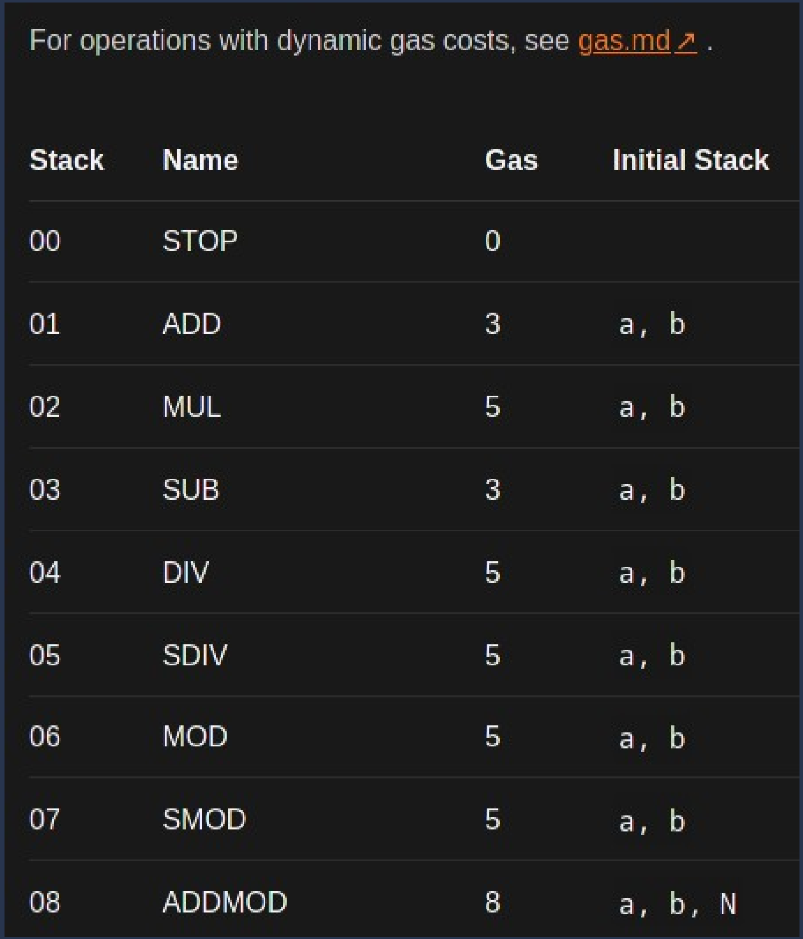
- Vitalik Buterin published the white paper in 2013 with the goal of building entire apps, not just a financial system
- Instead of maintaining a large list of transactions, an entire computer is simulated – The Ethereum Virtual Machine
- When a block is created:
  - The miner starts with the last blocks end state, S
  - It selects a number of transactions to process, [T]
  - For each transaction, it applies them to the previous state, getting the new state of the EVM
  - Eventually, all transactions have been processed, leaving the new state for the block S’
  - From here Proof of Work is performed

Ethereum block structure

- Everything from Bitcoin, plus:
  - BaseFeePerGas
  - StateRoot: Another Merkle tree with
    - Balances (no UTXOs)
    - App storage and code
    - Account nonce (transaction counter)
- Block target is 12-14 seconds, much faster than Bitcoin

Gas

- Each opcode of the EVM has a “gas” fee associated with it
  - The most expensive ones being those that store data in the blockchain
  - It costs 753,072 gas to store 1KB (in Feb 2022, this is about $88)



For operations with dynamic gas costs, see gas.md ↗ .

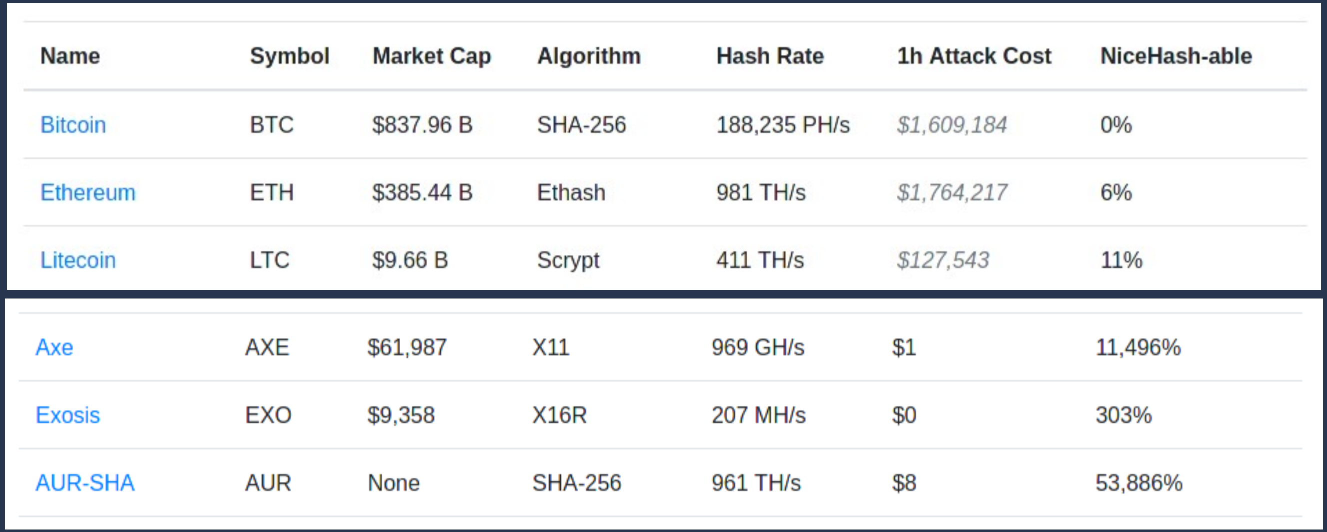
Stack	Name	Gas	Initial Stack
00	STOP	0	
01	ADD	3	`a, b`
02	MUL	5	`a, b`
03	SUB	3	`a, b`
04	DIV	5	`a, b`
05	SDIV	5	`a, b`
06	MOD	5	`a, b`
07	SMOD	5	`a, b`
08	ADDMOD	8	`a, b, N`

Economics

- Each transaction will take a certain amount of gas
- The user specifies how much ETH they'll pay per unit of gas, as well as a limit on how many units they'll pay
- This gas is a fee that the miners get for including your transaction in a block
- Miners get to choose which transactions to include in a block, so they're incentivized to choose the transactions that give them the highest reward
- Blocks are limited by how much gas they can include, not size

Problems with Proof of Work

- It is setting the planet on fire*
  - Currently has a carbon footprint “similar to Switzerland”, which is 0.8% of the US’s footprint
  - More importantly, it’s bad PR
- 51% attacks
- Decentralization



Name	Symbol	Market Cap	Algorithm	Hash Rate	1h Attack Cost	NiceHash-able
Bitcoin	BTC	$837.96 B	SHA-256	188,235 PH/s	*$1,609,184*	0%
Ethereum	ETH	$385.44 B	Ethash	981 TH/s	*$1,764,217*	6%
Litecoin	LTC	$9.66 B	Scrypt	411 TH/s	*$127,543*	11%
Axe	AXE	$61,987	X11	969 GH/s	$1	11,496%
Exosis	EXO	$9,358	X16R	207 MH/s	$0	303%
AUR-SHA	AUR	None	SHA-256	961 TH/s	$8	53,886%

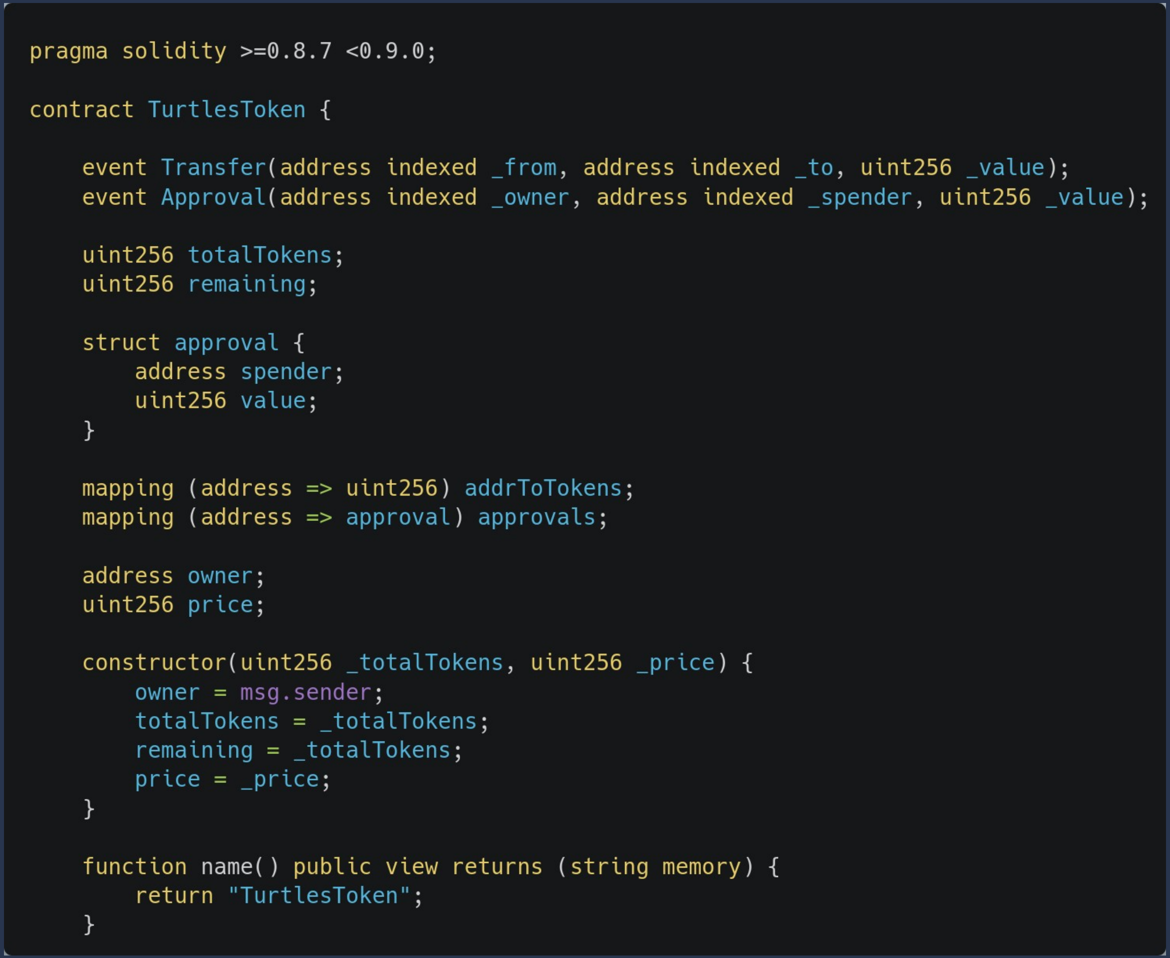
Proof of Stake

- Replaces the energy and time requirement with the threat of losing money
- Validators put at least 32 ETH up for stake
- Every 32 blocks, a committee of at least 128 validators is randomly selected
- One validator will propose the block, while others will check it and attest to it
- If the committee says yes, the block is added
- If a validator doesn’t show up to their job, they lose a portion of their stake
- If a validator proposes or attests to a malicious block, they lose all of their stake
  - Trying to replace a block you already finalized
  - Attesting to a block that doesn’t exist
  - Refusing to propose blocks
  - Deliberately not allowing transactions into the block

Sharding and the Beacon Chain

- Ethereum will be split into 64 separate “shard” chains, each processing transactions on their own
- Shards only record
- A central beacon chain will record attestations from validators and make them available to other shards
- This allows a theoretical transaction rate of 100,000 TPS
- Also, any computer can be a node even if it doesn’t have an RTX 9000

Smart Contracts



```
pragma solidity >=0.8.7 <0.9.0;

contract TurtlesToken {

    event Transfer(address indexed _from, address indexed _to, uint256 _value);
    event Approval(address indexed _owner, address indexed _spender, uint256 _value);

    uint256 totalTokens;
    uint256 remaining;

    struct approval {
        address spender;
        uint256 value;
    }

    mapping (address => uint256) addrToTokens;
    mapping (address => approval) approvals;

    address owner;
    uint256 price;

    constructor(uint256 _totalTokens, uint256 _price) {
        owner = msg.sender;
        totalTokens = _totalTokens;
        remaining = _totalTokens;
        price = _price;
    }

    function name() public view returns (string memory) {
        return "TurtlesToken";
    }
```