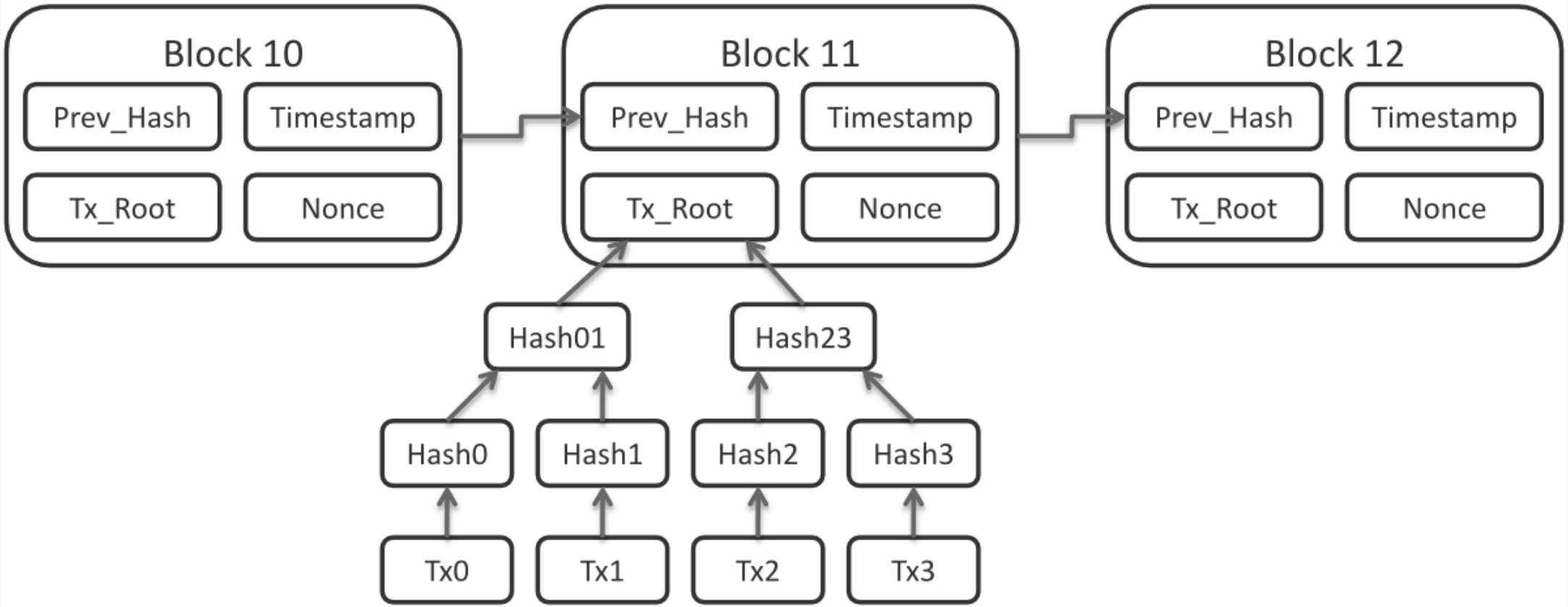
CSE446: Blockchain & Cryptocurrencies

Lecture – 15: Ethereum - 4

Agenda

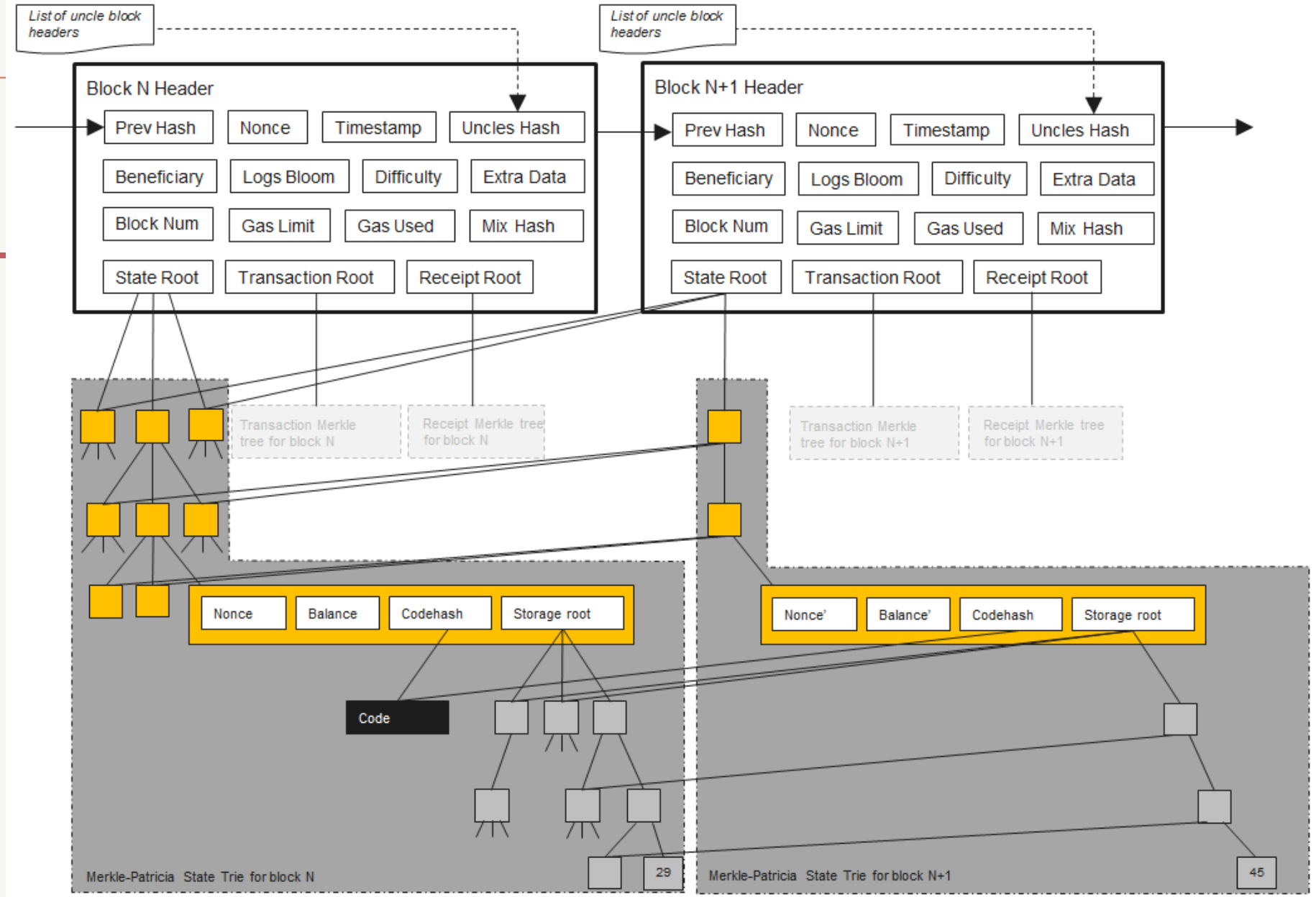
- Ethereum Tries
- Ethereum consensus

Bitcoin blockchain



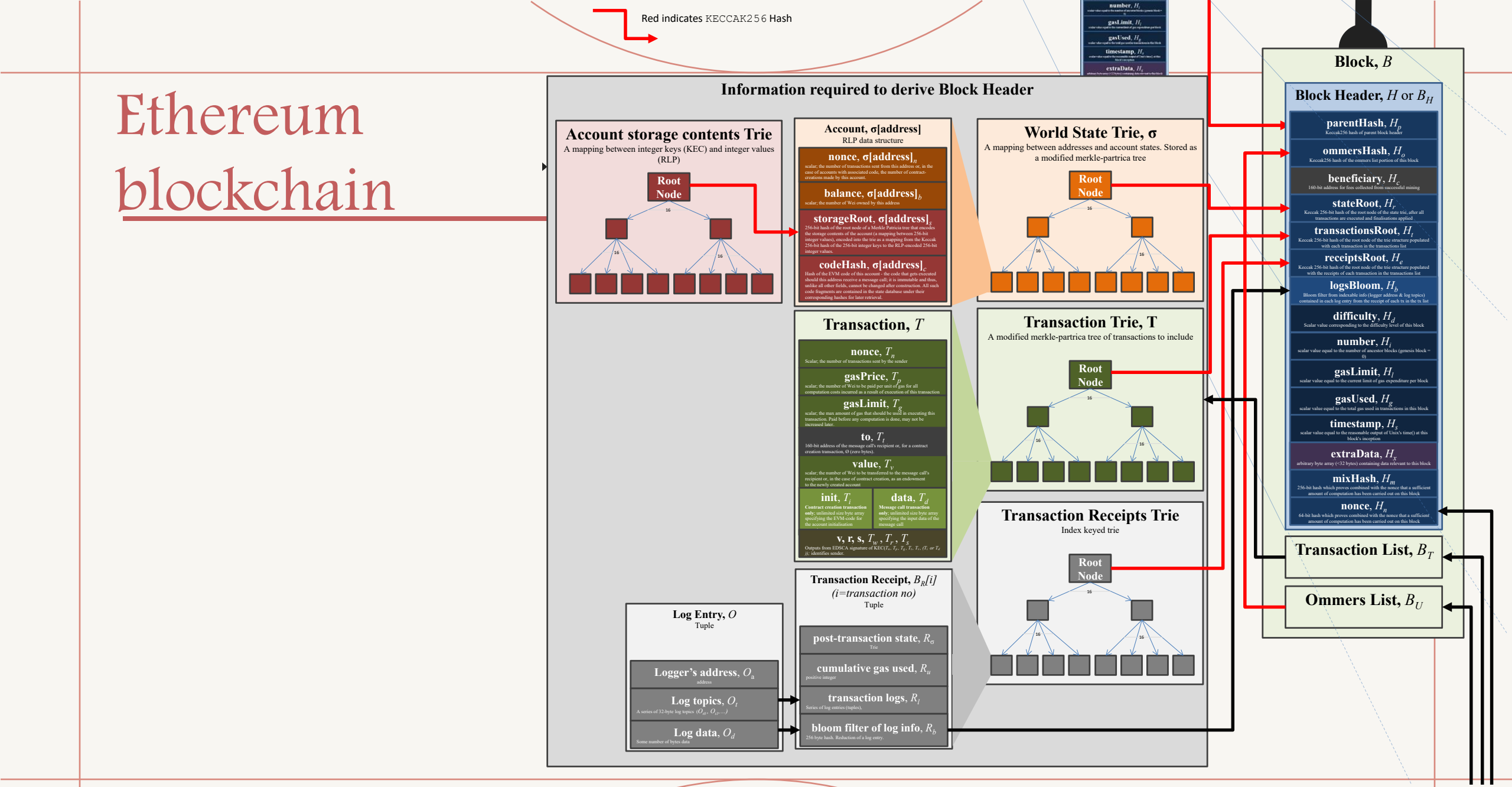
https://i.stack.imgur.com/HrKX0.png

Ethereum blockchain



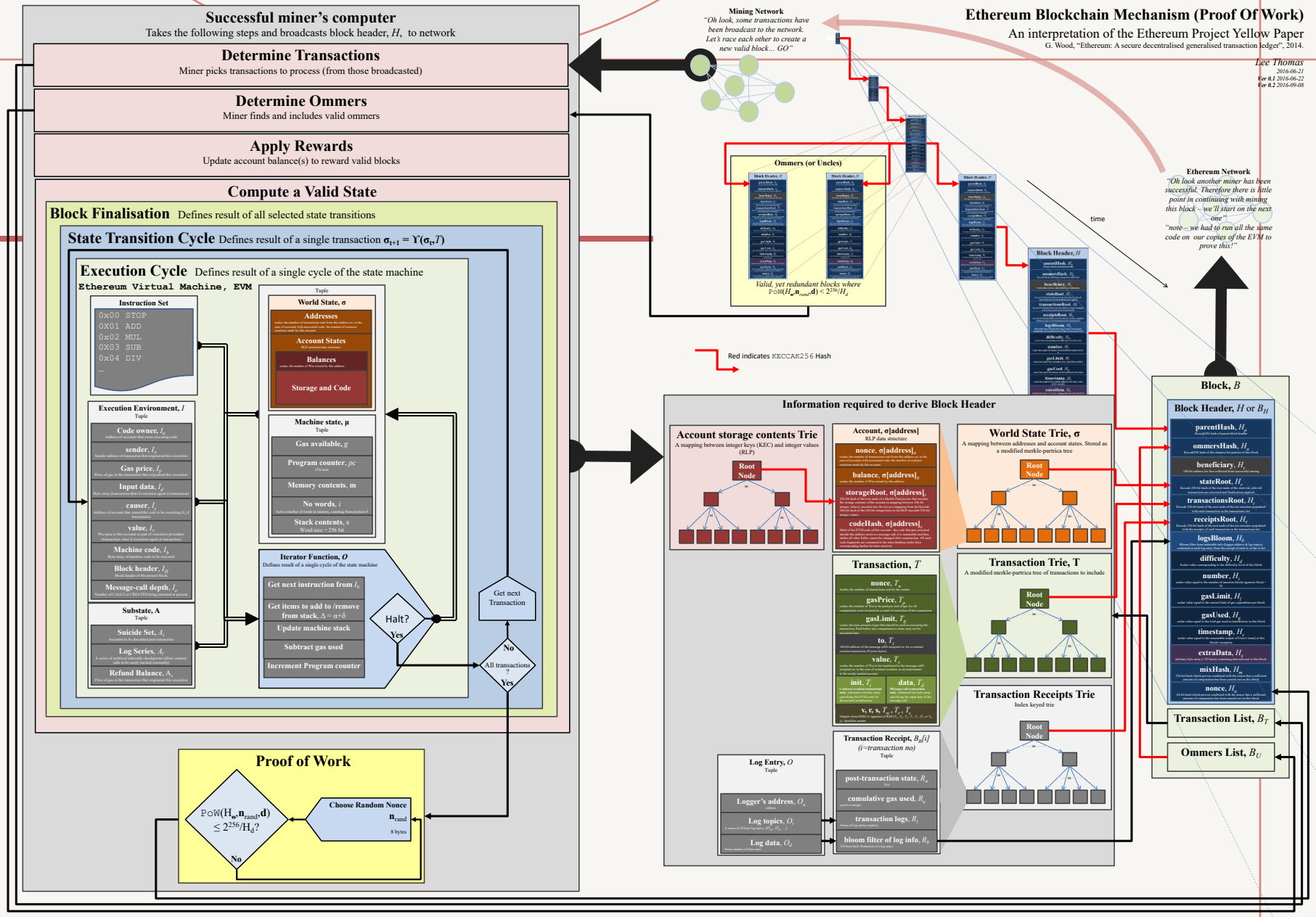
https://i.stack.imgur.com/eOwjD.png

Ethereum blockchain



https://github.com/4c656554/BlockchainIllustrations/blob/master/Ethereum/EthBlockchain5.svg

Ethereum blockchain



https://github.com/4c656554/BlockchainIllustrations/blob/master/Ethereum/EthBlockchain5.svg

Bitcoin consensus

- The PoW algorithm utilised in Bitcoin is called a Compute-bound consensus algorithm
- A Compute-bound PoW, also known as CPU-bound PoW, employs a CPU-intensive function
  - that carries out the required computational task by leveraging the capabilities of the processing units (e.g., CPU/GPU)
  - and it does not rely on the main memory of the system
- These particular characteristics can be massively optimised for faster calculation by using Application-specific Integrated Circuit (ASIC) rigs

Bitcoin consensus

- This is not an ideal scenario as now general people with their general purpose computer cannot participate in the mining process
- The mining process is mostly centralised among a group of mining nodes
- Many crypto-currency enthusiasts suggest that this is not a democratic process and facilitates the “rich getting richer” scenario

Memory-bound consensus algorithm

- To counteract this issue of Bitcoin's CPU-bound PoW algorithm, memory-bound PoWs have been proposed
- A memory-bound PoW requires the algorithm to access the main memory several times
- This ultimately binds the performance of the algorithm within the limit of access latency and/or bandwidth as well as the size of memory
  - The higher the memory the faster the performance

Memory-bound consensus algorithm

- This restricts ASIC rigs based miners not to have manifold performance advantage over CPU/GPU-based mining rigs
- The reason is even though thousands of ASICs could be combined they would have a performance threshold based on the size/bw/latency of the memory
  - Remember that you can't install unlimited memory within a PC
- This approach also limits the profit margin of the miners who have a mammoth ASIC-based mining rig
- Another motivation of this approach is to de-monopolise the mining concentrations around some central mining nodes

Ethereum consensus algorithm

- Ethash (DAGGER-HASHIMOTO)/DAGGER is the consensus algorithm designed for Ethereum
- Ethash is a memory-bound PoW algorithm with the goal to be ASIC-resistant for a long period of time
- Dagger is one of the earliest proposed memory-bound PoW algorithms which utilises a Directed Acyclic Graph (DAG)
  - a directed acyclic graph (DAG) is a directed graph with no directed cycles
- However it was found be vulnerable
- Ethereum combined Dagger and Hashimoto algorithms to be more secure

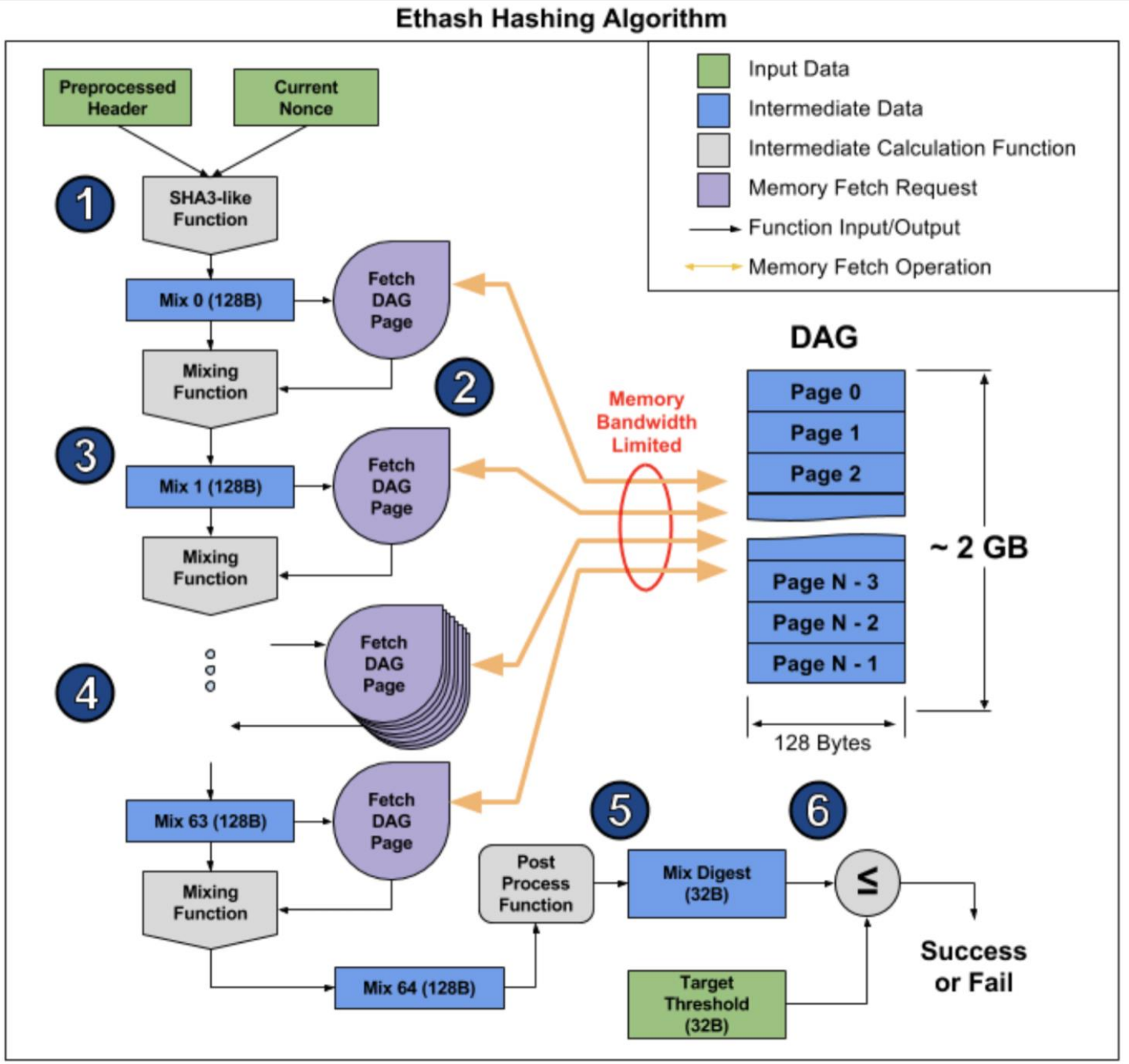
Ethash algorithm

- Ethash depends on a large pseudo-random dataset (the DAG), which is recomputed during each epoch
- Each epoch is determined by the time it takes to generate 30,000 blocks which is approximately five days
- During the DAG generation process, a seed is generated at first, which relies on the length of the chain
- The seed is then used to compute a 16 MB pseudo-random cache
- Then, each item of the DAG is generated by utilising a certain number of items from the pseudo-random cache

Ethash algorithm

- Then, the latest block header and the current candidate nonce are hashed using Keccak (SHA-3) hash function
- The resultant hash is mixed and hashed several times with data from the DAG, the mixHash data field
- The final hashed digest is compared to the difficulty target and accepted or discarded accordingly

Ethash algorithm



From Lecture Slides of Blockchains & Cryptocurrencies Course at Johns Hopkins University

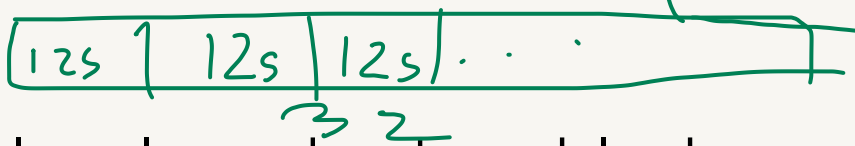
Ethereum PoS algorithm

- Ethereum moved to a PoS consensus algorithm in 2022
- Like any PoS algorithm there are a number of validators
- To be a validator, one has to deposit 32 eth to an escrow contract
- On depositing their ETH, the user joins an activation queue that limits the rate of new validators joining the network
- Once activated, validators receive new blocks from peers on the Ethereum network
- The transactions delivered in the block are re-executed, and the block is verified
- The validator then sends a vote (called an attestation) in favour of that block across the network

https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/

Ethereum PoS algorithm

- Under Ethash algorithm, the timing of blocks is determined by the mining difficulty
- In proof-of-stake, the block generation time is fixed
- Time in proof-of-stake Ethereum is divided into slots (12 seconds) and epochs (32 slots)



- One validator is randomly selected to be a block proposer in every slot
- This validator is responsible for creating a new block and sending it out to other nodes on the network

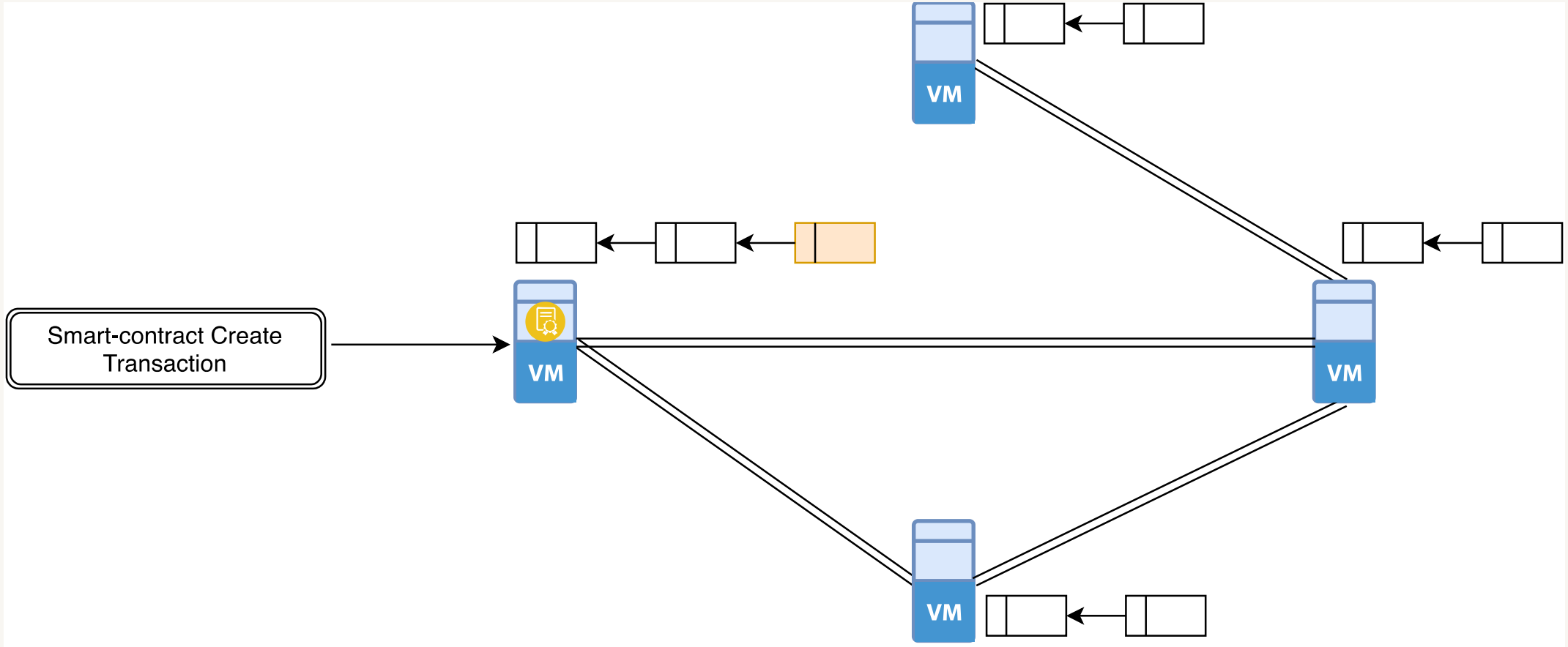
https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/

Ethereum PoS algorithm

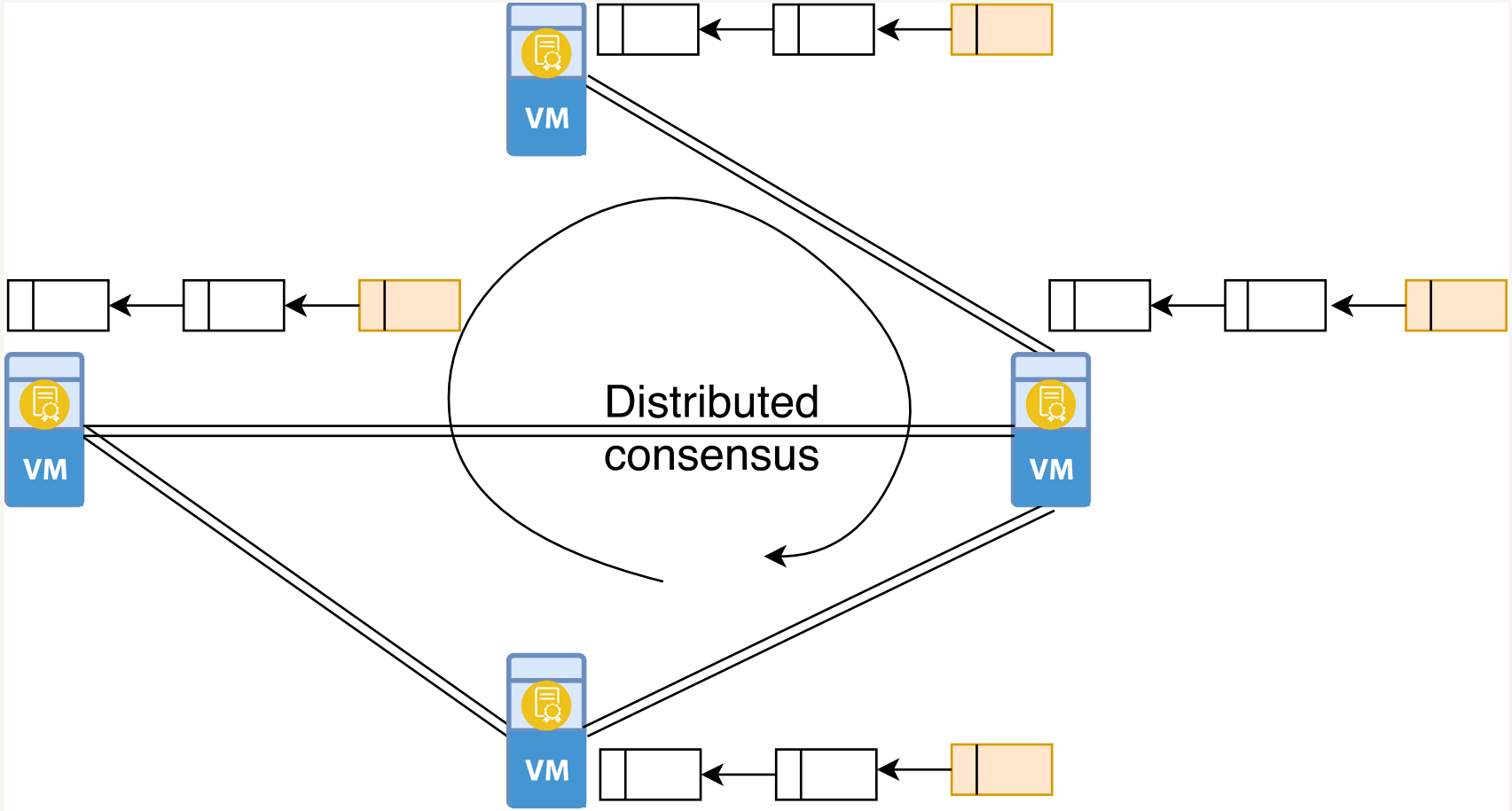
- Also in every slot, a committee of validators is randomly chosen, whose votes are used to determine the validity of the block being proposed
- If a validator is chosen to attest the next block, they are rewarded in ETH as a percentage of their stake
- Conversely, validators who do not perform their duties--if they are offline, for example--receive penalties, or slashes, in the form of small amounts of ETH subtracted from their stakes

https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/

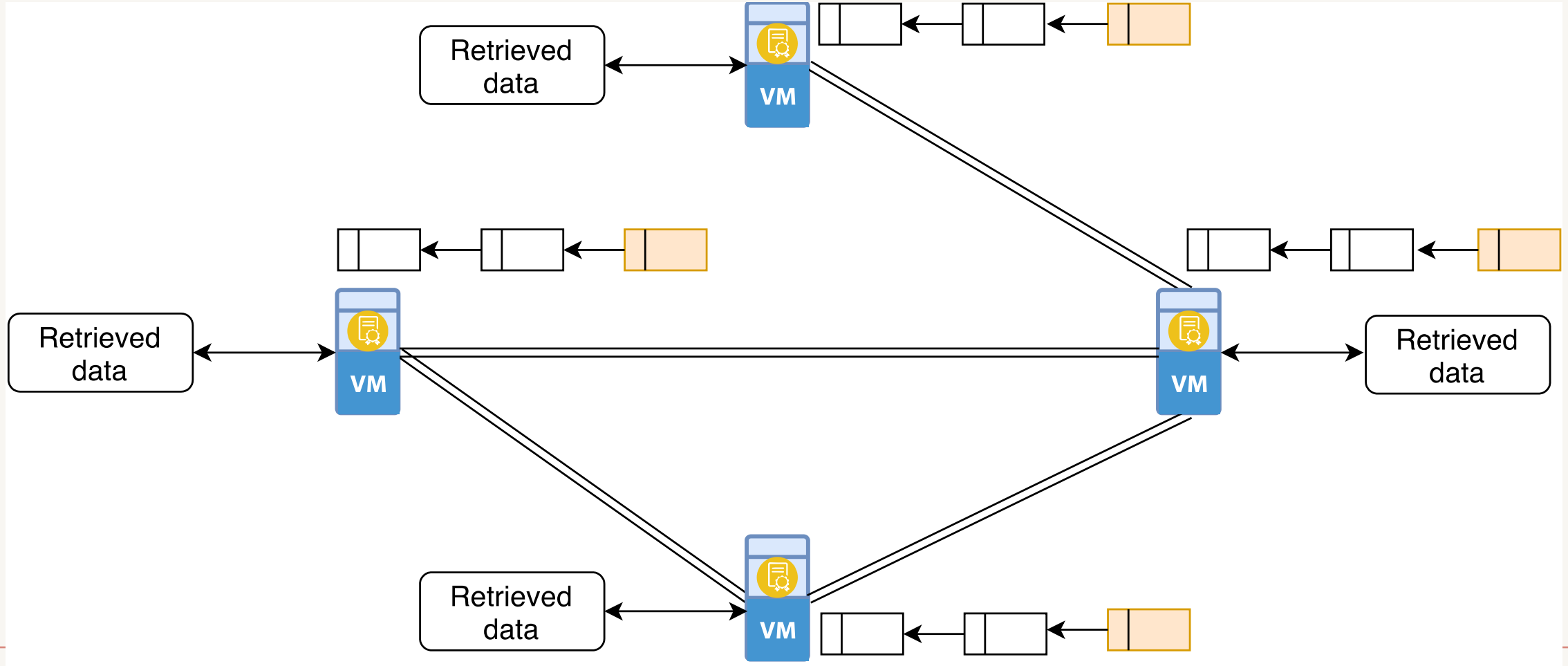
Ethereum illustration



Ethereum illustration

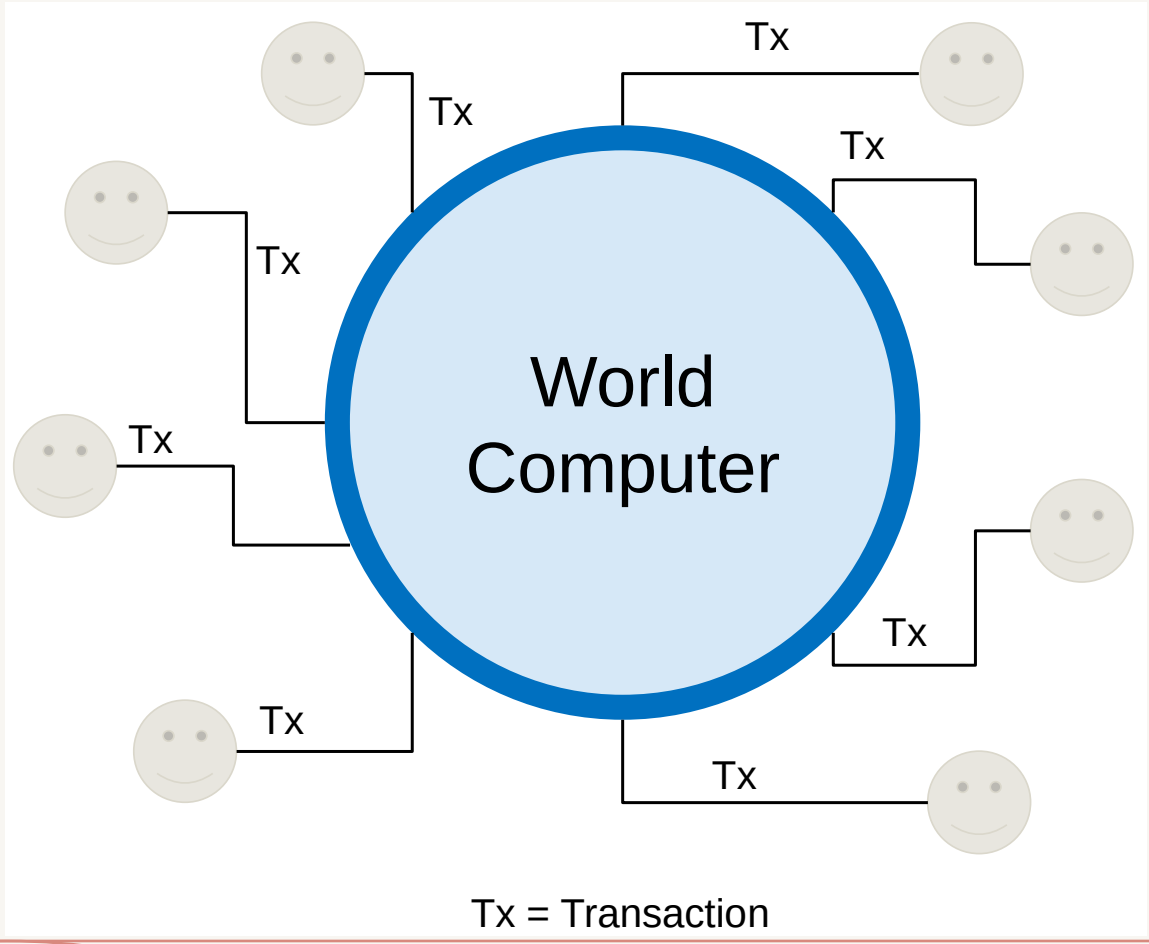


Ethereum illustration

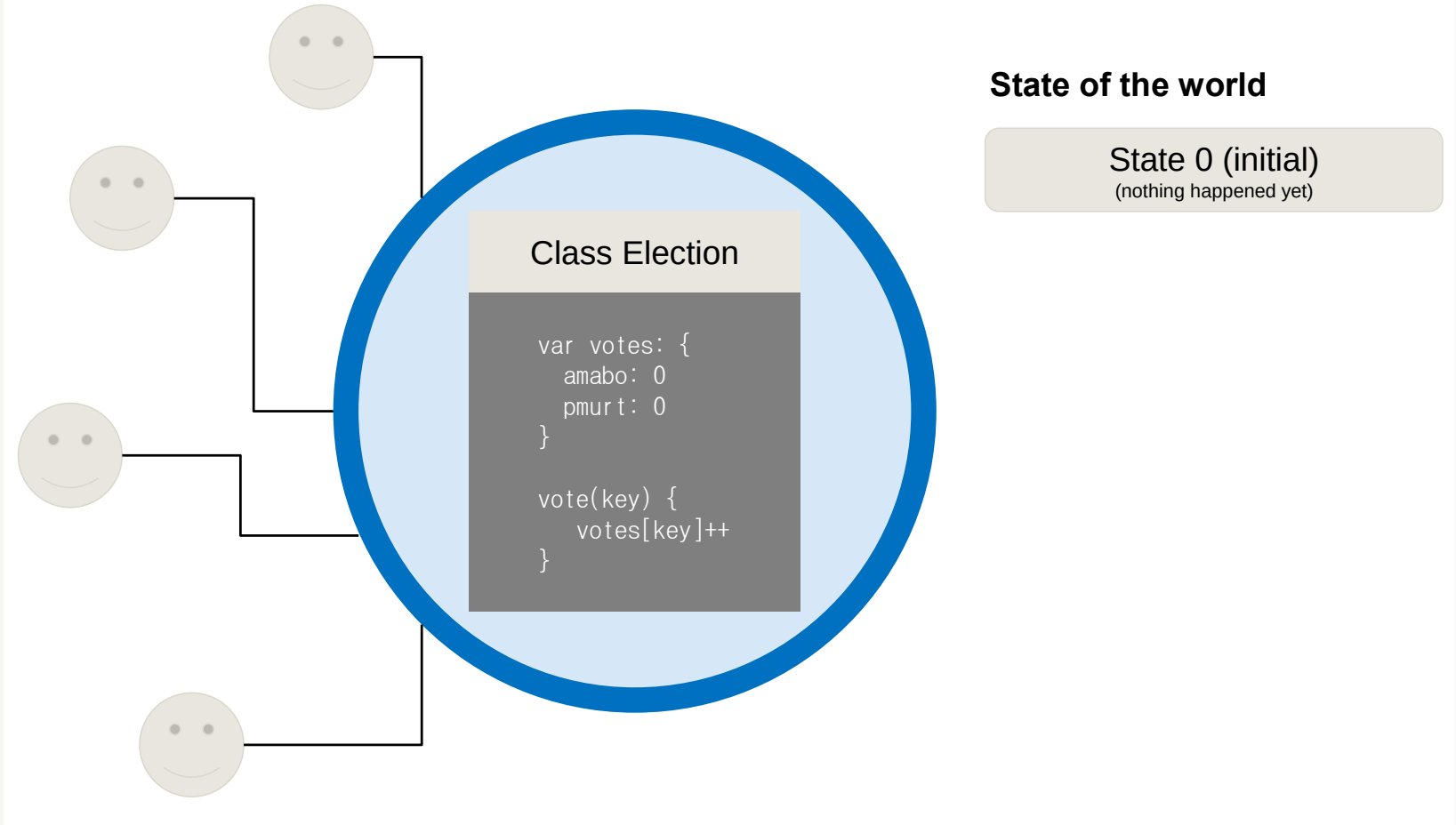


Ethereum visualisation

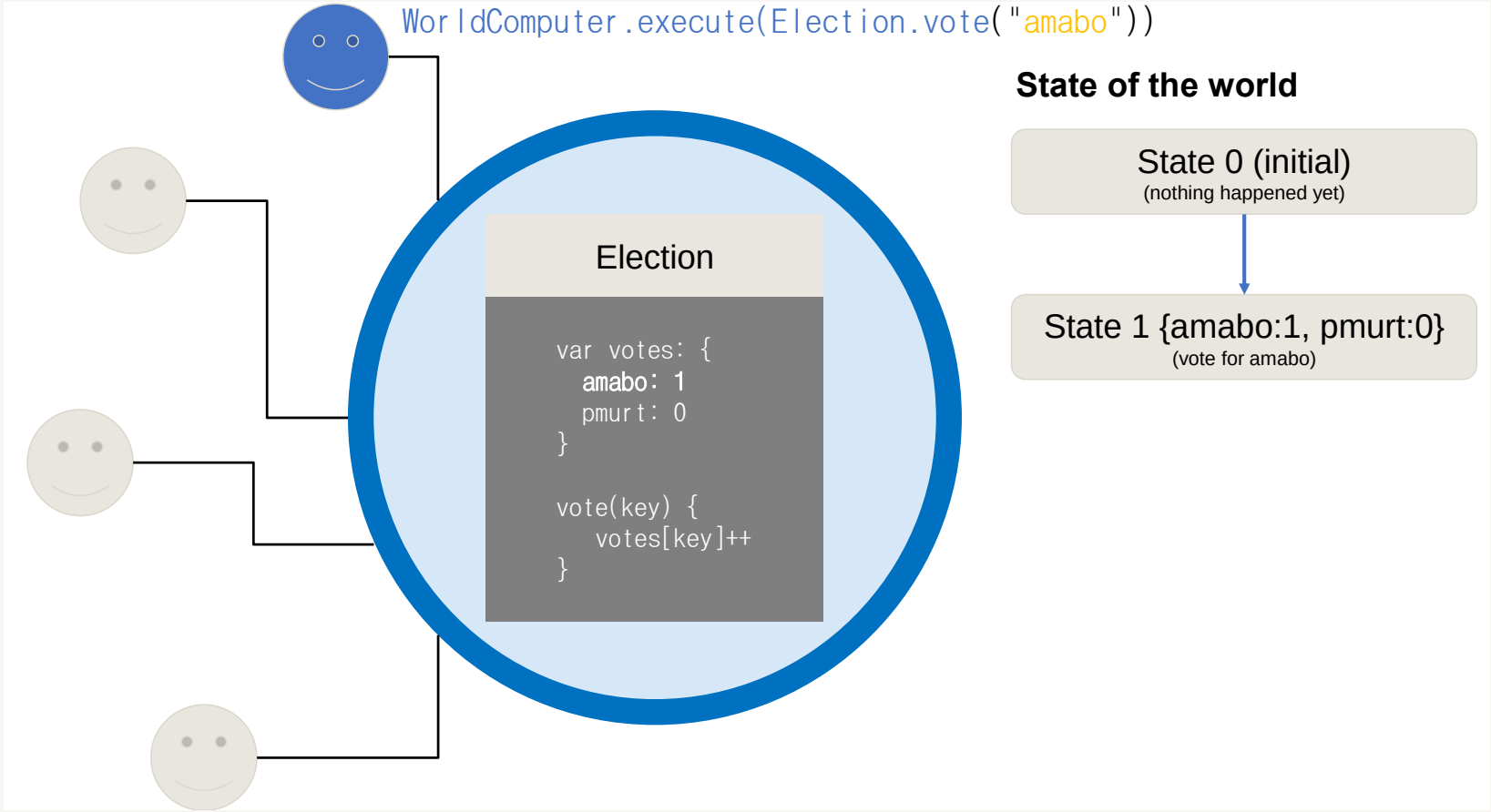
- All participants are using the “same” computer
- Users issue transactions to call programs on the computer
- Everyone shares the same resources and storage
- The computer has no explicit, single owner
- Using the computer’s resources costs money (eth)



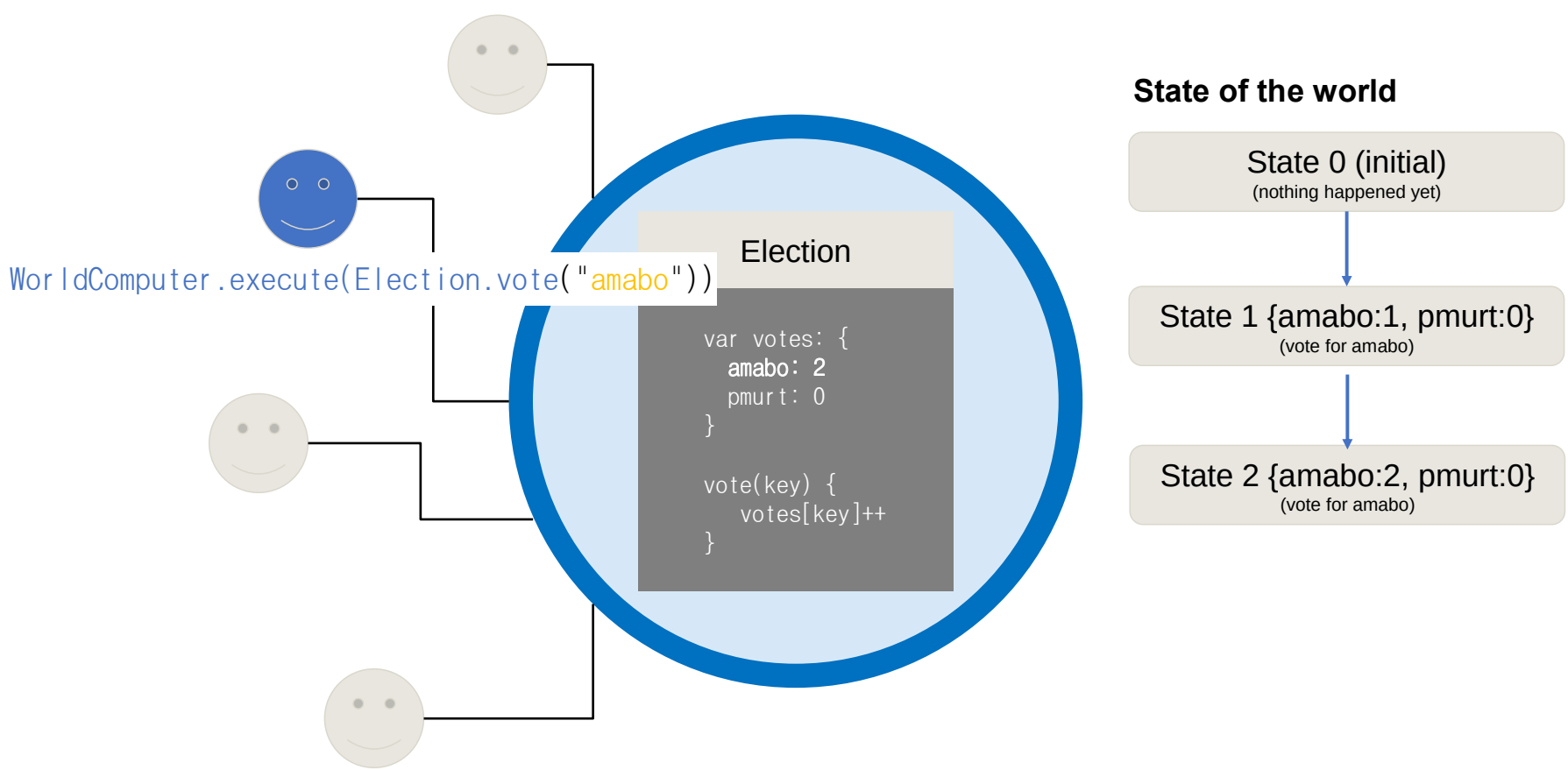
Ethereum visualisation: election example



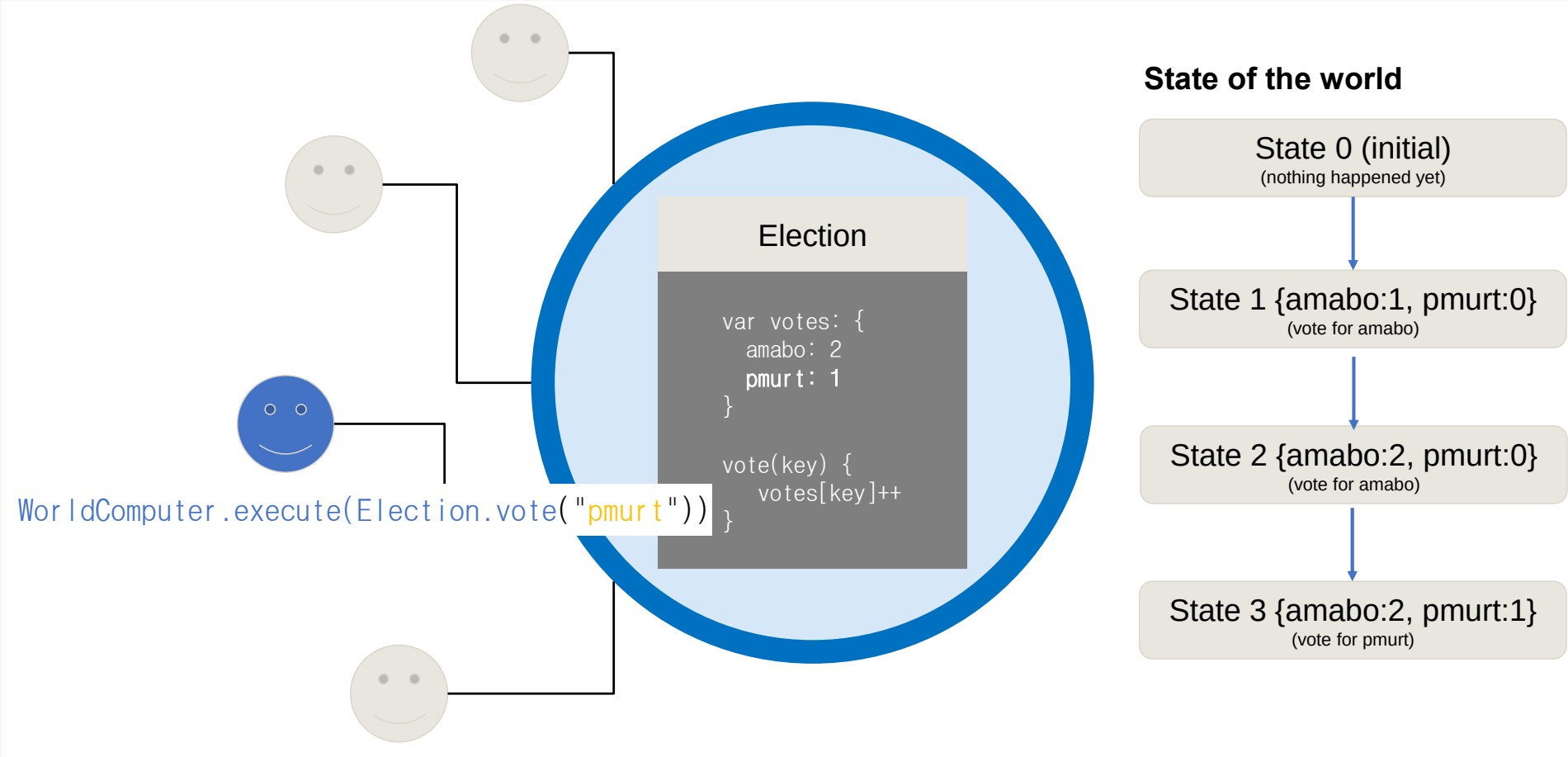
Ethereum visualization: election example



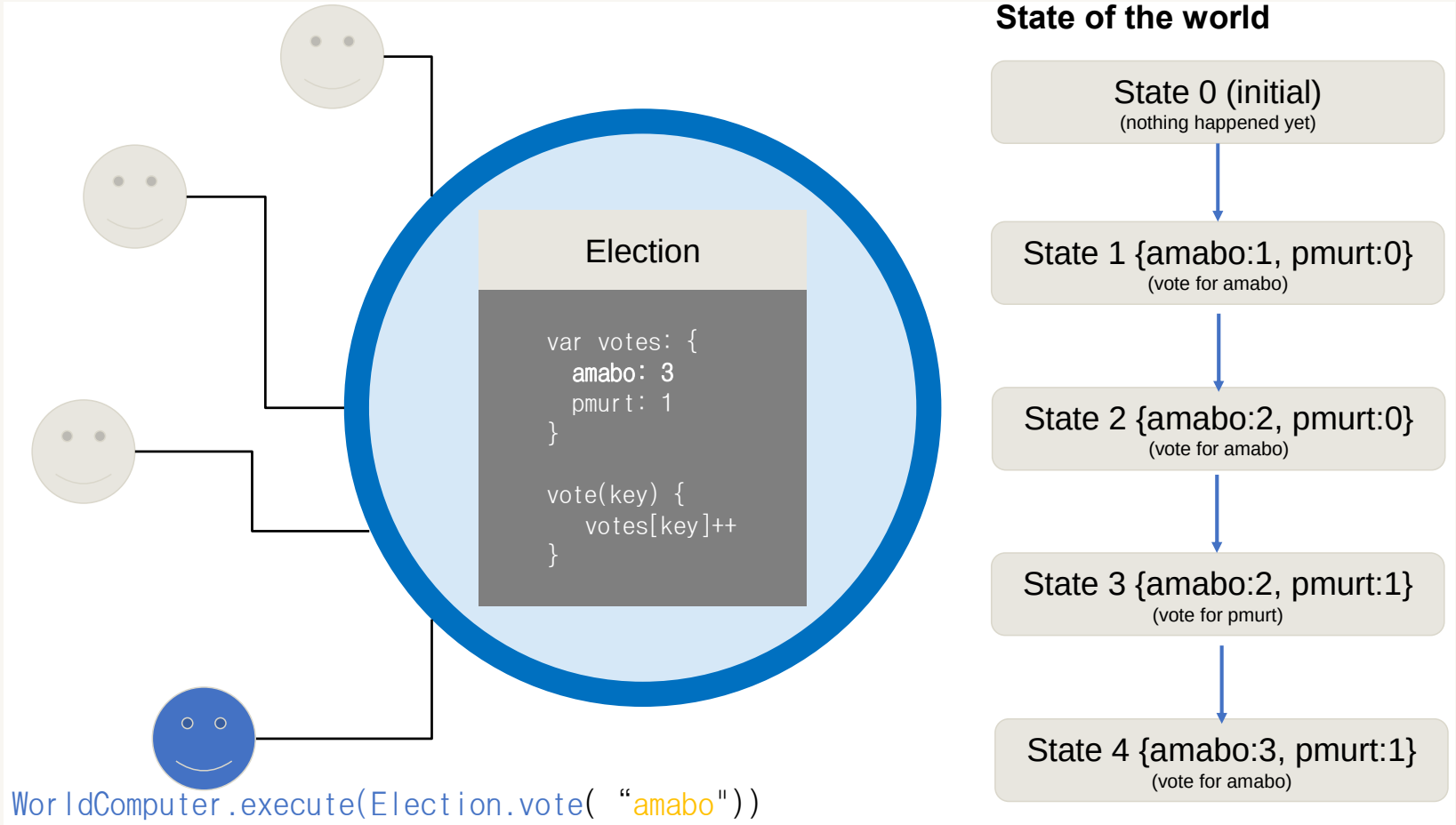
Ethereum visualization: election example



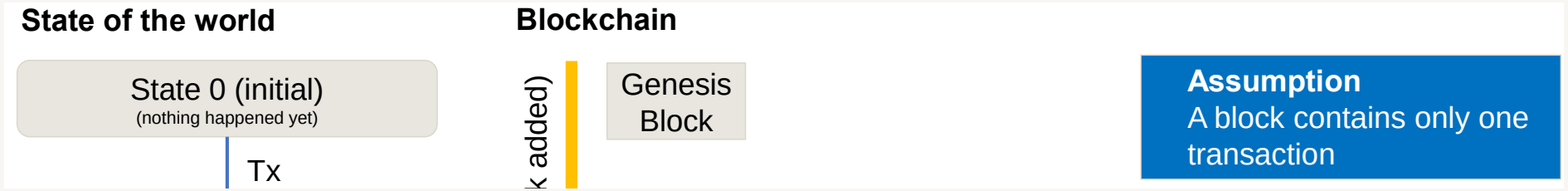
Ethereum visualization: election example



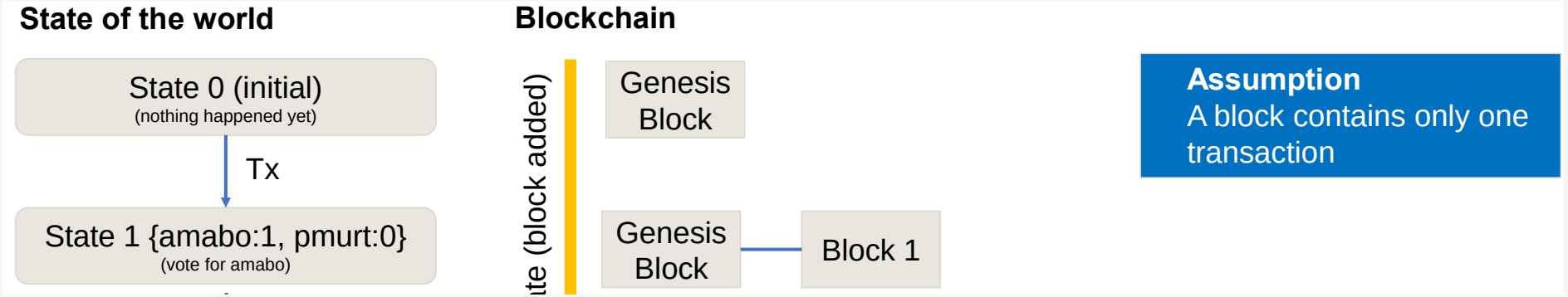
Ethereum visualization: election example



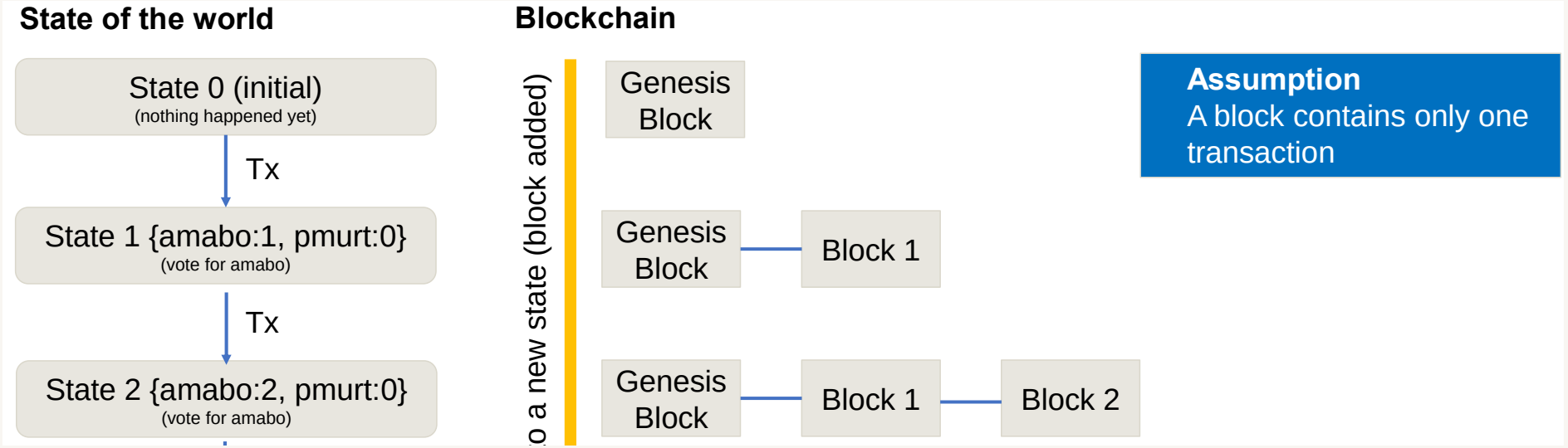
Ethereum visualization: election example



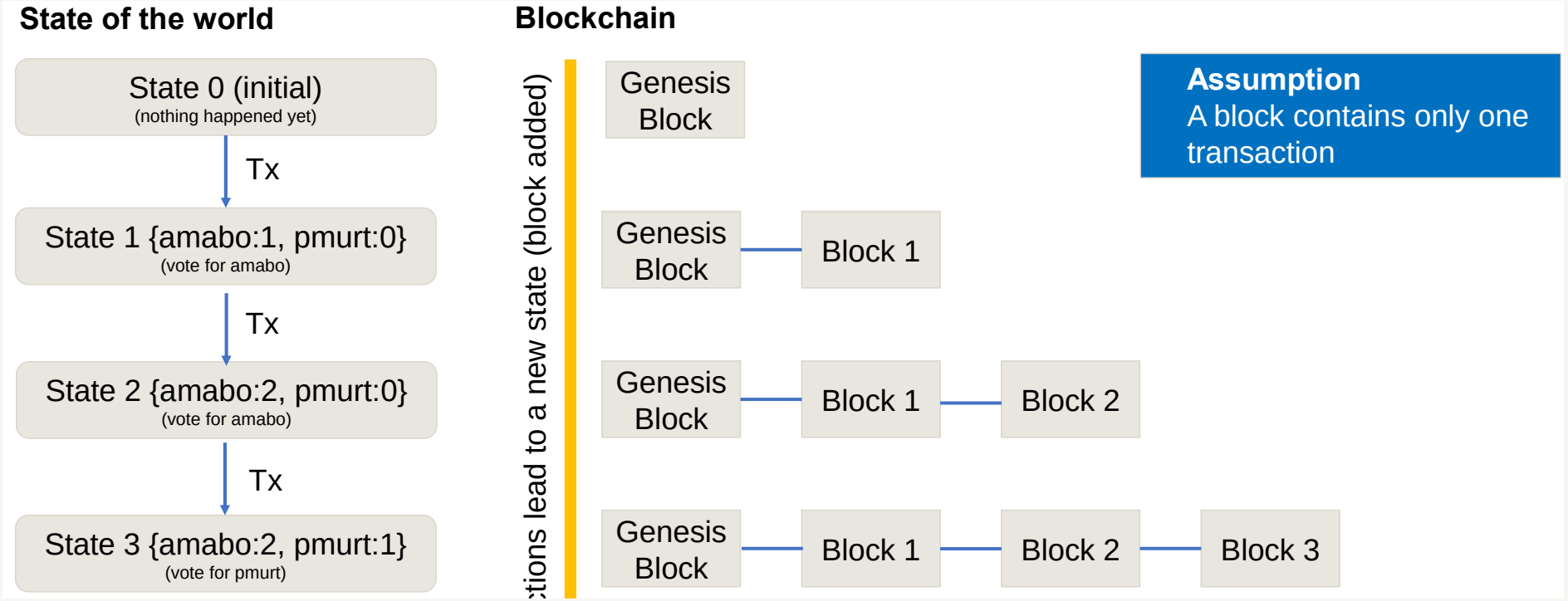
Ethereum visualization: election example



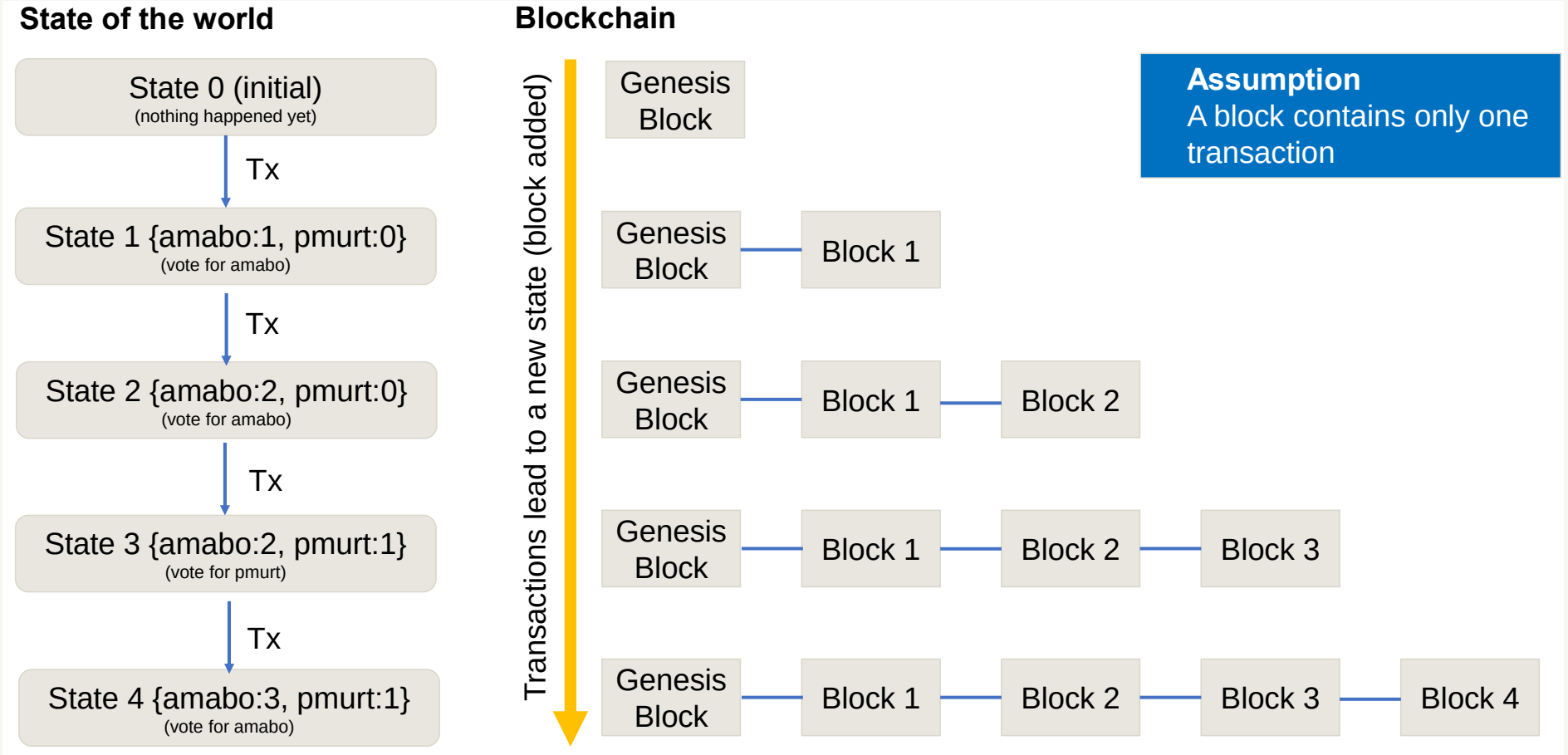
Ethereum visualization: election example



Ethereum visualization: election example



Ethereum visualization: election example



Question?