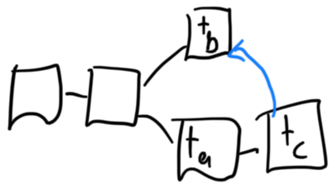
More uncles, bitcoinNG, and sharding

More uncles



Idea: Execute transactions from uncles.

$$t_a \rightsquigarrow \underset{\substack{\uparrow \\ \text{only execute } t_b \\ \text{if not conflicting}}}{t_b} \rightsquigarrow t_c$$

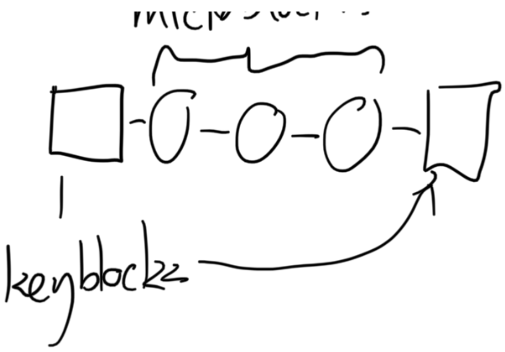
not done in Eth.

Problem

Mostly uncles will include the same transactions

Bitcoin NG

microblocks



safety

Keyblocks

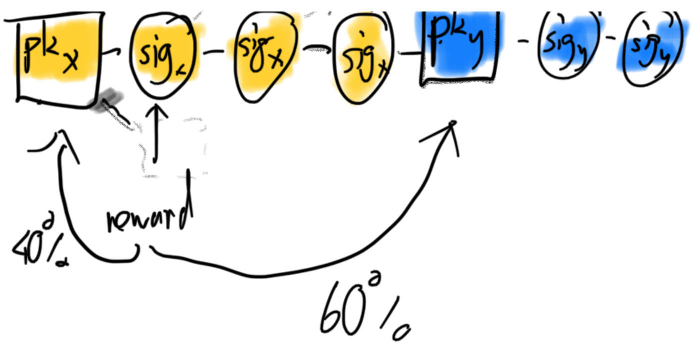
- No transactions ⇝ small ⇝ small network delay δ
- PoW
- seldom ⇝ difficult PoW ⇝ small p ⇝ few forks
- Pk of the new "leader"

trougput

Microblocks

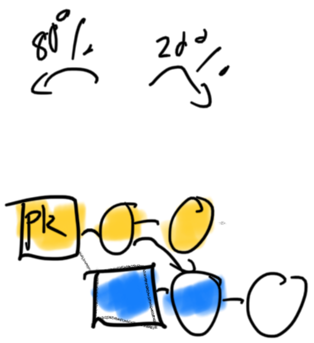
- No PoW
- Signed by the last "leader"
- Transactions
- frequent
- give reward

safety Use longest chain rule on keyblocks

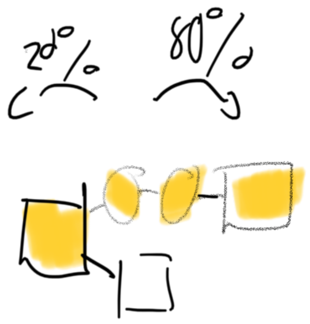


Possible attacks

1)



2)



Problem: Leader is a target for attack.
e.g. DDOS

Sharding

Shard: Subsystem with fraction of the state.

Potential for (linearly scaling

Potentially linear in scaling

Bank

Accounts: A–M

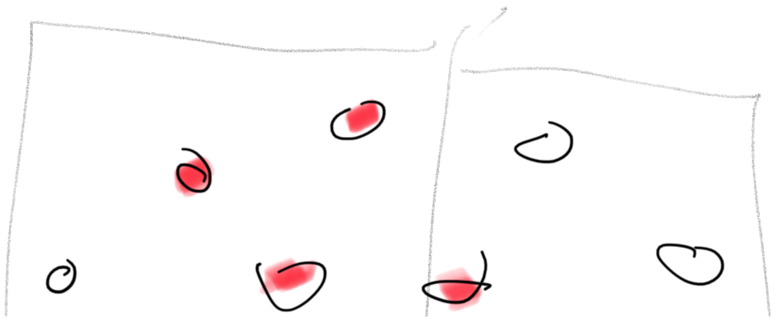
Account: N-Z

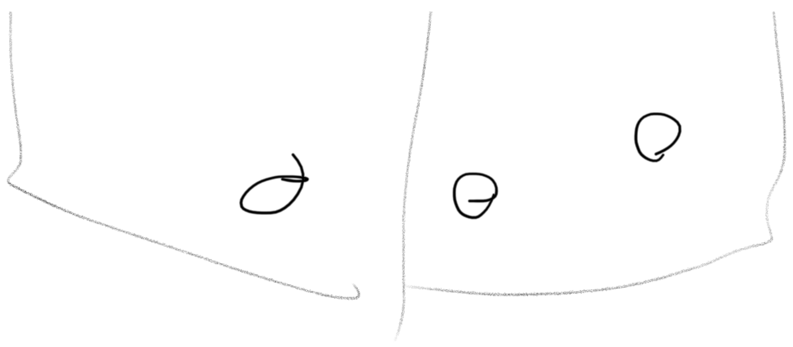
Problems

A: How to divide state?

B: Cross shard transactions?

C: Avoid mining power dillusion

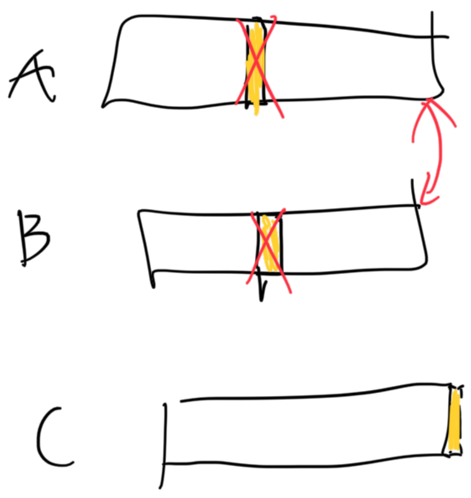




A solution:

Hashing ranges

B: Cross shard protocol



C: